

AI领导力：像管理团队一样管理AI

揭光发 (Jeff)

腾讯云架构师同盟社群管理主席；腾讯技术专家



目录

- 01 Vibe Coding —— "心流"在编码中重现
- 02 Vibe Working —— "心流"延伸至所有工作
- 03 AI 领导力 —— 新范式下的核心四要素
- 04 行业影响 —— 软件世界的"板块漂移"
- 05 超级个体的崛起
- 06 总结与展望

📍 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 AIOps
- 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
- LMOps
- 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

Vibe Coding — "01" 心流"在编码中重现

AI驱动的编程新范式，让开发者重获创造心流



从Vibe Coding讲起

💡 从Vibe Coding讲起

个体开发者通过与 **AI对话**，专注于 **"构想"**，而将 **"实现"** 交给AI，从而在编码中重获心流状态



自然语言交互

用口语化描述需求，
无需关注语法细节



专注创意设计

将精力集中在高层次
问题与解决方案



开发效率提升

数小时完成过去数周
的工作量



迭代反馈循环

持续优化，快速验证
与调整

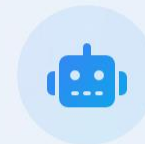
🔧 Vibe Coding 工作流程



构思需求



自然语言描述



AI实现代码

★ 核心价值

- ✓ 重获创造心流，提升专注度
- ✓ 降低技术门槛，扩大创新群体
- ✓ 加速产品迭代，缩短上市时间



我的 Vibe Coding 之路



在管理之余的碎片时间得到有效利用，主导多个重要技术基建和产品开发落地



团队 AI Coding 实践成果



早期布局优势

在我的影响下，团队很早就开始了AI Coding进程，抢占先机，建立了技术优势和开发文化



效率大幅提升

团队代码AI含量平均达到50%以上，开发效率显著提升，释放更多时间专注于架构设计和业务创新



技术基建突破

利用碎片时间主导多个重要技术基建和产品开发落地，从管理者重新回归技术一线的实践者

团队AI含量



让曾经资深的架构师
重新焕发第二春 🌸



案例一：跨语言逻辑编排平台



多语言统一编排

支持Python、Go、Node.js等多种语言服务的统一编排和调度管理

Python

Go

Node.js



智能流程引擎

基于规则引擎和工作流的智能业务逻辑编排，支持复杂的条件分支和并行处理

规则引擎

工作流

分布式调度



可视化配置

提供直观的拖拽式界面，业务人员可通过可视化方式配置复杂的业务流程

React

Canvas

TypeScript

可视化配置层

拖拽式流程设计、规则配置界面



编排引擎层

流程解析、规则执行、任务调度



服务适配层

多语言服务统一接口适配



执行服务层

Python、Go、Node.js服务

90%+

AI 代码贡献度

案例二：AI Agent 低代码搭建平台



可视化Agent设计器

可视化构建AI Agent，支持多轮对话、意图识别、路由能力、知识库集成等核心能力

React

Canvas

Langchain



知识库集成

支持文档、FAQ、API接口等多种知识源，智能检索和语义匹配

Vector DB

Embedding

RAG



先进的Agent工具管理

支持多种Agent工具管理，支持工具链的开发和集成

MCP

HTTP

前端工具



全链路监控与 trace

全链路监控与 trace，支持全生命周期日志追踪和性能分析

全生命周期追踪

性能分析

可视化

日志分析

平台架构

核心组件

Flow 流程编排

Agent 智能体

LLM Completion



配置管理

可视化配置

版本控制

一键发布



多渠道部署

Chat UI

REST API

企微集成

SDK调用

95%+

AI 代码贡献

百万级

企业级调度量

案例三：Angular 到 React 大型重构



AI 驱动的重构流程

1

原组件分析与单测生成

AI深度分析Angular组件结构，自动生成覆盖各种场景的单元测试

2

React组件自动转换

基于原组件逻辑和API，AI生成对应的React Hooks组件

3

新组件单测生成

参考原单测覆盖场景，为新React组件生成完整的测试用例

4

自动化验证与优化

CI/CD集成，自动化测试验证，性能对比分析与优化建议

重构成果

94%

AI 代码贡献

80+

组件迁移

2000+

测试用例

6周

完成周期

传统需要6个月的重构工作
AI协助下6周完成



创新亮点

通过"先测试后重构"的策略，确保迁移过程中业务逻辑的完整性和正确性。AI不仅生成代码，更重要的是生成了全面的测试覆盖，为重构质量提供了强有力的保障。

我们为何失去了"创造心流"?



认知负荷过载

技术细节分散 **创造力注意力**，大脑被琐碎实现占据，无法专注于高价值思考



反馈循环过长

从想法到实现的周期太长，**创意热情**在漫长等待中消耗殆尽



协作摩擦内耗

沟通成本高、信息不对称、决策链长，持续消耗 **创造能量**



价值认同缺失

重复性工作削弱 **成就感**，工作意义感缺失，内在驱动力不足

心流缺失的影响

- ▶ 创造力下降68%，创新产出显著减少
- ▶ 工作满意度降低，倦怠风险增加
- ▶ 学习效率下降，技能成长缓慢



心流如水流

心流状态就像顺畅流动的水流，而现代工作中的各种阻碍就像石头和堤坝，不断打断这股创造力的清流

"在无干扰的深度工作中，人类的创造力才能达到巅峰状态"

Vibe Coding与Software 3.0不得不说的关系



**要理解Vibe Coding
就不得不引入一个概念**

Software 3.0 — 基于Agentic System的全新生产力范式



让我们先来了解这个支撑Vibe Coding的技术基础



Software 3.0 范式

🕒 软件开发范式的演进

1.0

传统代码

人类编写指令

2.0

神经网络权重

数据驱动优化

3.0

自然语言接口

提示词即程序

</> Software 1.0

显式编程 · 人类编写精确指令 · 逻辑明确 · 语法严格



Software 2.0

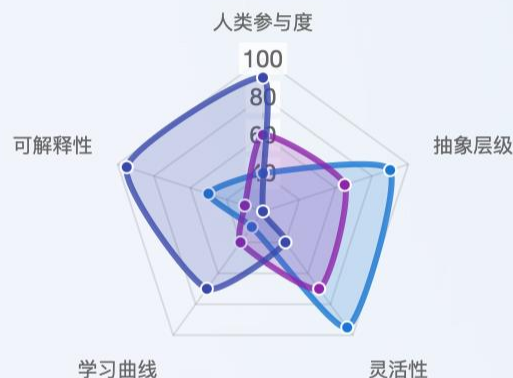
权重优化 · 调整数据集 · 训练神经网络 · 自动生成参数



Software 3.0

自然语言编程 · 提示词即代码 · 英语成为编程语言 · LLM执行任务

三种范式特性对比



Software 1.0 Software 2.0 Software 3.0



范式转变的本质

从 "如何实现" 到 "想要什么" · 自然语言成为新接口 · 人机协作新形态



三种范式实现情感倾向判定功能的对比



Software 1.0

规则驱动

⚙️ 实现方式

```
if word in ['excellent',  
'amazing']:  
    return "褒义"  
if word in ['terrible',  
'awful']:  
    return "贬义"  
if word in ['the', 'a',  
'is']:  
    return "中性"
```

✓ 显式规则 · 人类编写



Software 2.0

数据驱动

⚙️ 实现方式

```
model = NeuralNetwork()  
model.train(sentiment_corpus)  
result = model.predict(word)  
return result.sentiment
```

✓ 神经网络 · 数据训练



Software 3.0

意图驱动

⚙️ 实现方式

"请判断这段文本中每个单词的情感倾向"

[LLM直接返回结果:](#)

"excellent (褒义) terrible
(贬义) good (褒义) bad (贬义)
the (中性)..."

✓ 自然语言 · LLM执行

Software 3.0 的本质: Agentic System

什么是 Agentic System?

Software 3.0 的本质是 **智能体系统**，以自然语言为接口，让 AI 能够理解人类意图并自主执行任务

Agentic System 工作流程



💡 核心转变

从 **"编写代码"** 到 **"描述需求"**，人类角色从实现者转变为意图引导者

Vibe Working —— "心流" 02 延伸至所有工作

将AI协作理念从编码扩展到所有知识工作领域

从 Coding 到 Working 的跃迁

↑ Vibe Working 的核心理念

将 Vibe Coding 的核心理念（人主导意图，AI负责执行）从编码场景扩展到所有知识工作的全新工作范式

▲ 跨领域应用

✍ 写作创作

专注创意构思，AI辅助表达与润色

📊 数据分析

提出问题与假设，AI执行复杂计算

✂ 设计创意

定义美学方向，AI生成多样化方案

🔍 研究探索

聚焦核心问题，AI处理文献与数据





案例：Vibe Sliding - AI辅助演示文稿制作

▣ 传统PPT制作的痛点

以前做PPT最头疼的是：有了 **核心内容** 后，还要花大量时间思考 **排版布局**、寻找 **合适图片**、调整 **视觉效果**。这些工作固然重要，但却是最费时、最枯燥无味的部分，常常导致拖延症发作。

✂️ Vibe Sliding 工作流程

1 内容构思

专注核心观点和逻辑架构，用自然语言描述想要表达的内容

2 AI对话协作

通过持续的自然语言交流，让AI理解需求并逐步完善PPT结构

3 Web版输出

AI生成基于HTML/CSS的网页版PPT，支持响应式设计和现代交互

4 迭代优化

继续对话调整细节，实时看到修改效果，直到满意为止

🏆 核心优势

- ✓ 告别排版焦虑，专注内容创作
- ✓ 实时预览修改，即时反馈优化
- ✓ 现代化设计，自动响应式布局
- ✓ 自然语言交互，零设计门槛
- ✓ Web原生支持，跨平台兼容
- ✓ 大幅提升制作效率，减少拖延

<> 技术实现方式

这正是现阶段大多数AI PPT产品的核心实现方式：通过 **AI Coding** 技术，将用户的自然语言需求转化为 **HTML/CSS/JavaScript** 代码，生成具有现代化视觉效果的Web版演示文稿。这种方式兼具了 **灵活性** 和 **可维护性**。

案例：Vibe Writing - AI写作助手

✍️ AI写作的核心价值

传统写作往往在 **结构梳理**、**语言表达**、**格式规范** 等方面耗费大量精力。Vibe Writing让创作者专注于 **核心观点** 和 **创意构思**，AI负责文本优化和格式完善。

🏠 应用场景

📄 技术文档

项目文档、API接口文档、系统设计文档等技术写作

📖 用户手册

产品使用指南、操作手册、FAQ文档等用户向内容

📋 营销内容

公众号文章、产品介绍、品牌故事等营销传播文案

📊 商务写作

商业计划书、项目提案、工作报告等正式文档

✦ 核心优势

- ✓ 快速构建文档结构
- ✓ 自动格式规范化
- ✓ 实时协作式修改
- ✓ 大幅提升写作效率
- ✓ 智能语言润色优化
- ✓ 多风格适应调整
- ✓ 跨领域专业表达
- ✓ 降低写作门槛

↔️ 协作流程



案例：Vibe Reading - AI阅读助手

📖 AI阅读的核心价值

面对 **海量信息**、**复杂文档**、**专业论文** 时，传统阅读效率低下。Vibe Reading让用户专注于 **核心理解** 和 **深度思考**，AI负责内容解析和知识提取。

📍 应用场景

📄 学术论文

快速理解研究方法、核心观点、实验结果等学术内容

📖 万字长文

深度报告、调研文章、行业分析等长篇内容解读

🎥 视频内容

视频号、课程视频等多媒体内容的智能解析

📁 批量文档

PDF、Word、Excel等多格式文档的统一处理

✦ 核心优势

- ✓ 快速提取关键信息
- ✓ 跨语言内容理解
- ✓ 交互式问答解读
- ✓ 大幅提升阅读效率
- ✓ 智能总结核心观点
- ✓ 多格式文档支持
- ✓ 知识图谱构建
- ✓ 深化理解层次

元 腾讯元宝

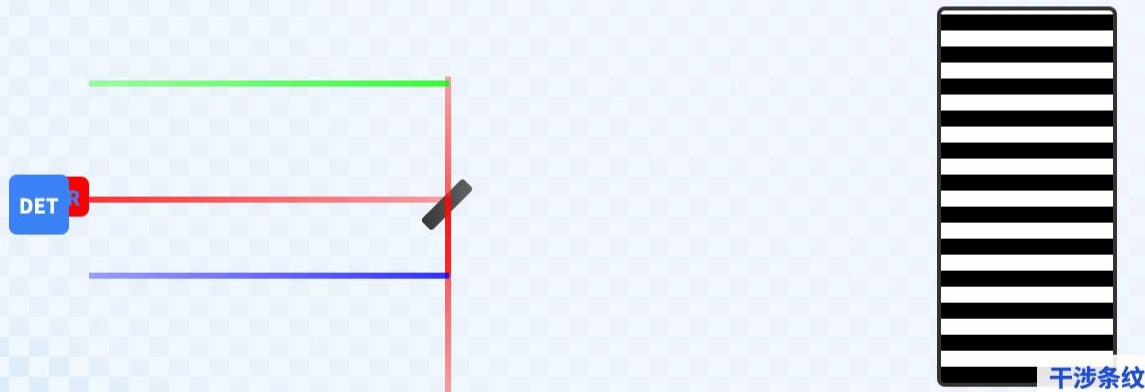
- ◆ 单文档最长1000万字
- ◆ 支持PDF、DOCX、TXT
- ◆ 智能问答式解读
- ◆ 多语言文档处理
- ◆ 一次解析最多50个文件
- ◆ 支持XLSX、PPTX格式
- ◆ 视频号内容解析
- ◆ 实时协作式阅读

案例：Vibe Learning - AI学习助手

⚙️ AI学习的核心价值

传统学习中 **抽象概念** 难以理解，**复杂原理** 缺乏直观展示。Vibe Learning通过AI生成 **交互动画** 和 **可视化演示**，让学习者直观理解复杂知识，提升学习效果。

📌 迈克尔逊干涉仪演示动画



🔄 自动演示进行中

✦ 核心优势

- ✓ AI生成交互动画
- ✓ 复杂概念可视化
- ✓ 实时参数调节
- ✓ 即时反馈机制

▶ 自动演示状态

- 正在改变光波长...

观察干涉条纹的自动变化过程，体验光波干涉的物理原理



案例：Vibe Scheduling - AI个人秘书

🕒 AI驱动的个人效率革命

传统的日程管理依赖手动记录、容易遗忘、效率低下。通过 **Gemini CLI** 作为智能个人秘书，实现 **自动化工作流程**：智能记录待办事项、自动创建工作目录、实时日志记录，甚至自动提交工作总结到Git仓库。

✨ 核心功能场景



智能工作目录

上班时自动创建当天的工作目录结构，包含TODO、日志、文档等子目录



智能待办管理

通过自然语言交互记录、跟踪、优先级排序待办事项，智能提醒



头脑风暴助手

实时记录思维碎片，AI辅助整理思路，形成结构化创意文档



自动归档总结

下班时自动生成当日工作总结，提交到Git仓库，形成工作档案

```
> 我还有哪些 todo?

+ 好的，这是你当前的待办和进行中任务：

  ✓ ReadFile TODO.md

+ 任务清单

待办 (To Do)
* [ ] 优化腾讯架构师联盟演讲PPT (创建于：2025-08-11, 优先级：高)

进行中 (In Progress)
* [ ] 撰写：[ ] (创建于：2025-07-30, 优先级：高)
  * 进度：
    * 2025-07-31: 因其他任务，推迟到明后天。
    * 2025-08-06: 再次推迟。

---

以上是您目前待办和进行中的任务。

Using 2 GEMINI.md files and 3 MCP servers (ctrl+t to view)

> [ ] Type your message or @path/to/file

~/projects/working-logs (main*)
```



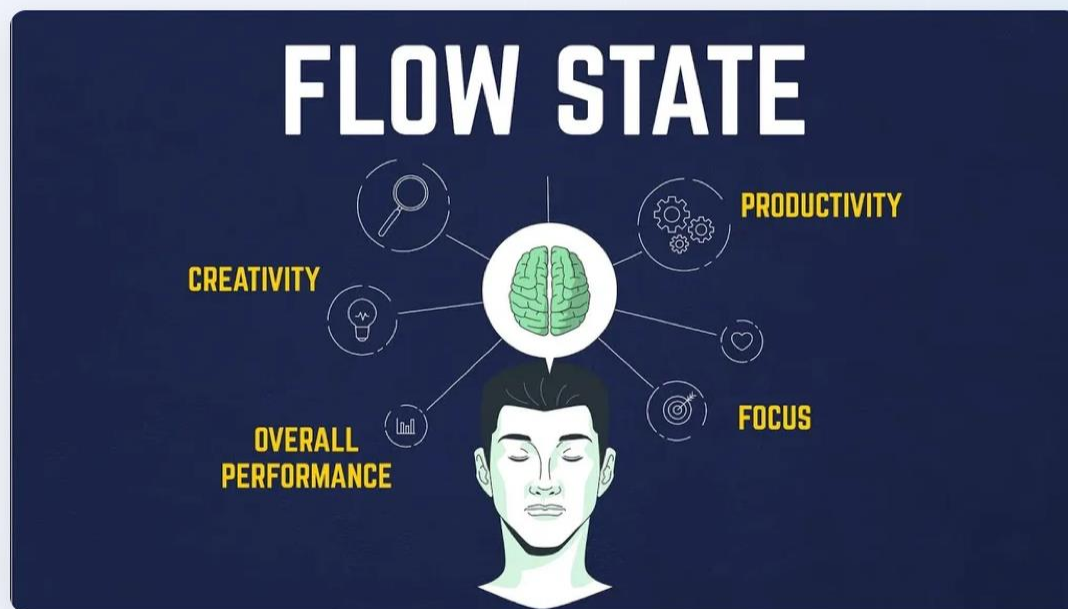
Vibe Working 的核心

🔧 人机协同的"Vibe"状态

Vibe Working的本质是一种 **人机协同** 的工作状态，让个体能将精力聚焦于 **最高价值的创造性思考** 上

→ 工作模式对比

维度	传统工作方式	Vibe Working
焦点	执行细节	创意构思
角色	执行者	意图引导者
价值	完成任务	创造突破



AI 领导力 —— 新范式下的核心四要素

高效驱动AI、将个人构想转化为卓越成果的核心方法论

定义"AI领导力"及四个关键要素

💡 什么是AI领导力?

不是 **管理人** 的能力，而是 **高效驱动AI**、将个人构想转化为卓越成果的核心方法论



知人善任

Know Your AI

深刻理解不同AI模型和工具的优势、劣势与适用边界，为正确的任务选择正确的AI

模型特点

能力边界

工具选择



目标设定

Set the Goal

将模糊的想法，转化为清晰、明确、可执行的目标和指令，为AI提供精准的"靶心"

清晰表达

具体指令

精准目标



过程管理

Direct the Process

在AI执行过程中，通过设定关键节点、提供约束条件、追问和引导，确保过程不偏航

关键节点

约束条件

过程引导



结果验收

Judge the Outcome

基于自身的专业知识、经验和品味，对AI的产出进行最终的审核、筛选和裁决

专业判断

质量把控

成果筛选



知人善任(Know Your AI)

🔍 核心理念

深刻理解不同AI **模型和工具的优势、劣势与适用边界**，为正确的任务选择正确的AI

💡 核心洞察

"知人善任"是AI领导力的基础环节，正确的AI选择是成功的一半，深入了解每种AI的特点和边界，才能发挥其最大价值

🗺️ AI选择决策框架

1

明确任务类型

识别是创意生成、逻辑推理、数据分析还是技术实现

2

评估复杂程度

判断任务的复杂度、专业性要求和质量标准

3

匹配AI能力

选择最符合任务需求的AI工具和模型

4

验证适用性

通过小规模测试验证选择的正确性



知人善任(Know Your AI)



大语言模型 (LLM)

擅长文本生成、理解、翻译和对话，适合内容创作、分析和知识问答

文本生成

逻辑推理

多语言

对话交互



图像生成模型

专门用于创造视觉内容，从艺术创作到产品设计，提供无限创意可能

艺术创作

产品设计

风格转换

概念可视化



代码助手

专业的编程辅助工具，提供代码生成、调试、优化和重构支持

代码生成

错误检测

性能优化

文档生成



音视频生成

专门用于音频和视频内容创作，包括语音合成、音乐生成和视频编辑

语音合成

音乐创作

视频生成

多模态编辑



AI编程工具选型



Claude Code

编程绝对好手

上下文管理大手大脚，代码质量极高，但容易触发token上限，需要精打细算使用。

✓ 优势

- ✓ 代码质量极高
- ✓ 理解能力超强
- ✓ 架构设计优秀
- ✓ 重构能力强

✗ 限制

- ✗ 容易触发token上限
- ✗ 成本相对较高
- ✗ 需要精细化管理
- ✗ 响应速度较慢



Cursor

IDE集成利器

使用Claude4时相对靠谱，但每次只有25次工具调用限制，难以一次性完成复杂功能。

✓ 优势

- ✓ IDE深度集成
- ✓ 实时代码提示
- ✓ 上下文感知强
- ✓ 开发体验流畅

✗ 限制

- ✗ 25次调用限制
- ✗ 复杂任务受限
- ✗ 需要分步完成
- ✗ 功能相对有限





AI编程工具选型



CodeBuddy IDE

AI驱动的全栈开发平台

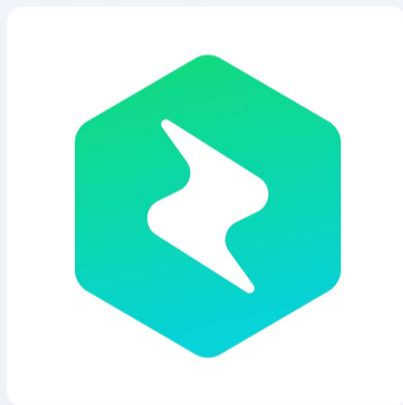
集成Figma的全栈开发IDE，支持从设计稿直接生成代码，打通设计→开发→部署完整链路，让设计师和开发者无缝协作。

✓ 优势

- ✓ Figma深度集成
- ✓ 设计稿自动转代码
- ✓ 设计→开发→部署全链路
- ✓ 用户完全拥有代码

✗ 限制

- ✗ 目前处于内测阶段
- ✗ 需要邀请码才能使用
- ✗ 功能仍在持续完善中
- ✗ 社区生态待建设



v0.dev

AI驱动的Web生成器

Vercel出品的Web应用生成器，从自然语言或设计稿直接生成React/Next.js代码，支持实时预览和迭代。

✓ 优势

- ✓ 零配置即用
- ✓ 设计稿转代码
- ✓ 实时预览效果
- ✓ 响应式设计自动化

✗ 限制

- ✗ 主要限于前端开发
- ✗ 定制化程度有限
- ✗ 依赖Vercel生态
- ✗ 收费门槛较高





AI模型选型实例



Gemini CLI

经适模型，皮糙肉厚

综合能力平衡，几乎免费使用，是性价比极高的经适模型，用来做秘书一流。

✓ 优势

- ✓ 几乎免费使用
- ✓ 综合能力平衡
- ✓ 稳定可靠
- ✓ 响应速度快

✗ 限制

- ✗ 精度不是最高
- ✗ 创新能力有限
- ✗ 复杂推理较弱
- ✗ 专业深度不足



GLM4.6

国产代码新王者

代码能力对齐Claude Sonnet 4，200K超长上下文，开源免费，是国产AI的技术新标杆。

✓ 优势

- ✓ 代码能力国内最强 (较4.5提升27%)
- ✓ 200K超长上下文 (128K→200K)
- ✓ 开源免费使用
- ✓ 完全国产化自主可控

✗ 限制

- ✗ 推理部署需要优化
- ✗ 复杂推理仍需提升
- ✗ 英文能力略弱
- ✗ 生态体系在发展中





新生代AI模型力量对比



Kimi K2

2025

月之暗面旗舰，海外专家盛赞

月之暗面发布的万亿参数智能体模型，专为AI Agent workflow设计，在海外获得高度评价。

✓ 优势

- ✓ 1万亿总参数，320亿激活参数
- ✓ 专为AI Agent能力设计
- ✓ 256K超长上下文窗口
- ✓ 开源模型，社区活跃

✗ 限制

- ✗ 生态建设仍需时间
- ✗ 复杂推理能力待提升
- ✗ 部署资源要求较高
- ✗ 中文场景优化需加强



DeepSeek V3.2 Exp

2025

性价比之王，价格革命

实验版模型，采用DSA机制实现推理成本直接折半，Agent能力实测表现出色。

✓ 优势

- ✓ DSA机制，推理成本降低50%+
- ✓ Agent能力实测非常能打
- ✓ API价格大幅下调75%
- ✓ 推理效率大幅提升

✗ 限制

- ✗ 实验版本，稳定性待验证
- ✗ 某些复杂任务能力有限
- ✗ 文档和工具生态待完善
- ✗ 长文本处理仍需优化



目标设定：从提示词技巧到需求表达

◎ 核心理念演进

AI Agent时代，目标设定重点从 **提示词技巧** 转向 **需求表达能力**



Chatbot时代

通用对话，提示词技巧导向



角色设定

精心设计角色prompt激活AI特定能力



指令优化

调整用词、语序、语气获得更好输出



提示词工程

使用CoT、Few-shot等专业技巧



Agent时代

场景化协作，需求表达导向



需求深度理解

清晰理解自己的真实需求和问题



场景精准描述

详细描述使用场景、约束条件和标准



目标拆解能力

将复杂目标分解为Agent可执行任务



需求表达的核心能力

🔊 从"我会说"到"AI能懂"

AI Agent时代，**需求表达能力**比提示词技巧更重要。清晰的场景描述、准确的约束定义、明确的期望管理，是高效AI协作的基础。

🔍 场景深度理解

深入理解真实需求和使用场景，避免表面化表达。

- ✓ 区分"想要"和"需要"
- ✓ 具体使用场景描述
- ✓ 充足背景信息提供
- ✓ 上下文管理能力

🌾 约束条件设定

明确定义边界条件、技术限制和质量标准，为AI提供清晰框架。

- ✓ 技术栈限制明确
- ✓ 质量标准设定
- ✓ 禁止行为定义
- ✓ 边界红线设定

🎯 期望精准管理

清晰表达预期结果和成功标准，确保AI理解与期望一致。

- ✓ 可量化成功指标
- ✓ 输出格式要求
- ✓ 优先级明确
- ✓ 权衡取舍标准

🔧 任务分拆能力

将复杂目标分解为Agent可执行的独立任务，设计合理执行顺序。

- ✓ 核心路径识别
- ✓ 任务正交性确保
- ✓ 独立性验证
- ✓ 验收节点设计



设计vs使用：两种不同的目标设定思维



设计AI Agent

提示词工程导向



角色设定与身份构建

通过精心设计的角色prompt，激活AI的特定知识库和能力模式。专业身份定义和工作风格设定。



指令优化与格式定义

使用专业的提示词技巧，优化AI的响应质量和输出格式。CoT、Few-shot等技巧应用。



约束与边界设定

预先定义AI的行为边界和安全限制，确保输出的稳定性和可控性。禁止行为和红线设置。



使用AI Agent

需求表达导向



场景描述与问题定义

清晰描述具体的使用场景、业务背景和要解决的核心问题。真实痛点和根本问题分析。



目标明确与期望管理

明确表达想要的结果、成功标准和质量要求，确保理解一致。可量化指标定义。



任务分拆与依赖管理

将复杂目标分解为可管理的子任务，设计合理的执行路径。正交化任务分拆原则。



AI Agent时代的目标设定技巧



目标清晰化升级

从模糊想法到可执行目标，建立多层次目标体系，确保Agent理解准确。

✓ 场景化描述

用具体场景替代抽象概念

✓ 可量化指标

设定具体数字、时间、标准

✓ 验收条件

明确定义"完成"的标准



正交化任务分拆

将复杂目标分解为独立正交的子任务，最大化并发效率，减少冲突。

✓ 功能独立性

确保任务间无实现细节依赖

✓ 数据隔离

设计独立的数据和状态管理

✓ 接口标准化

定义清晰的通信协议



里程碑驱动设计

设定阶段性目标和检查点，建立可预期的反馈循环，及时调整方向。

✓ 关键节点设定

识别决策点和验证时机

✓ 渐进式交付

从MVP到完整功能的实现

✓ 风险识别

预先识别风险和备选方案



迭代优化机制

建立基于反馈的持续改进循环，从每次协作中学习和优化目标设定。

✓ 反馈收集

记录AI执行效果和问题

✓ 模式识别

总结成功失败的模式规律

✓ 知识沉淀

建立团队协作知识库



目标设定案例：市场营销场景

× 模糊的指令 (普通用户)

"帮我写几篇关于我们AI笔记应用的宣传文案。"

😞 AI输出结果

- 产出通用化文案，千篇一律
- 没有明确的受众定位
- 缺乏具体的营销策略
- 无法衡量和优化效果
- 风格单一，缺乏创意亮点
- 无法形成系统化内容规划

结果：产出非常通用、缺乏亮点的文案，营销效果有限

📁 案例场景

📱 AI笔记应用推广

目标：社交媒体引流获得5000+下载

✓ 清晰的目标设定 (AI领导力)

1. 设定目标与受众

我们的目标是吸引20-30岁的大学生和职场新人，他们对效率工具和AI新科技非常敏感。本轮营销的目标是在 **一个月内，通过社交媒体引流，带来5000次应用下载**。

2. 定义核心卖点与风格

文案需要突出我们的核心卖点："**一键将课堂录音和会议记录转化成结构化笔记**"。风格要年轻、有趣，多用emoji，避免使用生硬的营销术语。

3. 拆解具体任务

请为我生成一个为期一周的内容日历，发布平台是小红书和B站动态：

- 3篇小红书图文笔记：学习场景、工作场景、生活场景的应用
- 2篇B站动态：内容要更口语化，可以带一个互动性的话题
- 每篇需要一个吸引人的标题、5张配图的描述、相关的#标签

4. 设定衡量标准

所有文案需要包含应用的下载链接，并追踪每条内容的互动率（点赞、评论、转发）。

😊 优质输出结果

- | | |
|-----------------|----------------|
| ✓ 精准的受众定位和传播策略 | ✓ 完整的内容日历和执行计划 |
| ✓ 差异化的平台内容适配 | ✓ 可量化的营销目标和追踪 |
| ✓ 符合目标用户喜好的创意风格 | ✓ 系统化的营销流程设计 |

结果：获得完整、可执行、有针对性的营销方案



过程管理核心挑战



粒度管理

设定一个目标让 AI **一杆子插到底** 一口气做完，还是进行更细粒度的分拆？
如何判断任务的复杂度和AI的执行能力边界？



执行策略

当大目标可以分拆为多个子目标时，通过 **串行**，还是 **并行** 的方式驱动任务？

依赖关系如何分析？资源利用率如何优化？



工作树管理

并行任务推进中，工作树如何 **隔离避免污染**，成果合并时冲突如何解决？
版本控制策略和冲突解决机制如何设计？



验收瓶颈

当并行任务较多时，**验收工作占满人类心力带宽** 时，如何突破瓶颈？
如何设计自动化验收机制和优先级管理？



Kiro的规格驱动AI编程

Kiro核心 workflow



requirements.md

定义用户故事和验收标准
使用EARS格式规范



design.md

描述技术架构
数据模型和接口设计
结构化设计文档



tasks.md

生成具体的
可跟踪开发任务列表
任务分解与优先级



AI实现

Agent根据规格
自动生成代码
质量可控的产出



减少AI幻觉

结构化规格大幅降低AI理解偏差
确保输出符合预期



工程化保障

强制性的文档流程
为复杂项目提供纪律保障



BMAD-METHOD框架

💡 核心理念

人类作为指挥官，与AI Agent团队协同工作，实现覆盖项目全生命周期的 **智能化软件开发流程**



规划工作流 (Planning)

与AI产品团队协同，生成高质量规格文档：

PRD (产品需求文档)

架构设计图

用户故事切片

产品经理 AI

架构师 AI

产品负责人 AI



开发工作流 (Development)

基于规格进行迭代开发：

按用户故事开发

自动化测试验证

持续集成交付

敏捷教练 AI

开发代理 AI

测试代理 AI



工作区隔离策略



Docker 容器提供操作系统级别的虚拟化，为每个 AI Agent 任务创建完全独立的运行环境，包括文件系统、进程空间、网络 and 依赖包。

+ 优势

- ✓ 完全隔离
- ✓ 环境一致性
- ✓ 资源控制
- ✓ 安全性高

- 劣势

- ✗ 资源消耗大
- ✗ 启动较慢
- ✗ 管理复杂
- ✗ 存储开销



Git Worktree 允许在同一仓库中同时检出多个分支到不同目录，为每个 AI Agent 任务创建独立的代码工作区，共享 Git 历史但隔离工作文件。

+ 优势

- ✓ 轻量高效
- ✓ 快速切换
- ✓ 共享历史
- ✓ 简单易用

- 劣势

- ✗ 隔离有限
- ✗ 依赖冲突
- ✗ 进程共享
- ✗ 安全风险



结果验收的两种方式



核心原则

优先设计可自动化验收的任务,降低人工验收成本,提高验收效率和一致性



自动化验收

Automated Verification

基于明确的规则、逻辑和可计算标准进行验收,适用于有确定性答案的任务



逻辑验证: 数学计算、逻辑推理等可验证性任务



程序测试: 单元测试、集成测试、功能测试



格式检查: 代码规范、文档格式、数据结构



人工验收

Manual Verification

依赖专业判断、行业经验和审美品味进行验收,适用于需要主观评价的任务



专业判断: 方案设计、技术选型、架构决策



审美评价: 界面设计、文案创作、品牌形象



用户体验: 交互流程、产品易用性、情感共鸣



自动化验收:提升效率与一致性



自动化验收通过**明确的规则**和**可量化的标准**来评估AI输出,消除主观性,提高验收速度和一致性



逻辑与计算验证

验证数学计算、逻辑推理等有确定答案的任务

案例 数学题求解:验证计算结果的正确性

案例 SQL查询:执行查询验证返回数据



程序性测试

通过测试用例验证代码功能的正确性和健壮性

案例 单元测试:函数级别的功能验证

案例 集成测试:模块间协作验证



格式与规范检查

检查输出是否符合预定义的格式和编码规范

案例 代码风格:ESLint/Prettier 自动检查

案例 文档格式:Markdown/JSON 结构验证



数据完整性验证

验证数据的完整性、一致性和符合业务规则

案例 数据迁移:记录数完整性对比

案例 批量处理:字段有效性规则检查



人工验收:专业能力的体现



人工验收依赖验收者的**专业判断力**、**行业经验**和**审美品味**,是无法被算法完全替代的核心能力



行业认知

深入理解行业规则、业务逻辑和发展趋势,能够判断方案的可行性和前瞻性



专业能力

扎实的专业基础和实践经验,能够识别技术缺陷和潜在风险



品味与审美

对设计、体验和表达的高标准要求,追求卓越而非仅仅满足基本需求



验收方法

系统的评估框架和检查清单,确保验收过程全面且有条理



需要专业判断的场景

- 技术选型:评估不同技术方案的适用性和长期维护性
- 架构设计:判断系统架构的扩展性和可靠性



需要审美评价的场景

- UI设计:界面布局、配色方案、视觉层次的协调性
- 文案创作:品牌调性、情感共鸣、语言表达的精准度



需要用户视角的场景

- 易用性评估:从新手到专家的学习曲线是否合理
- 场景适配:是否真正解决用户在真实场景中的痛点



验收策略:自动化优先,人工保底

1 任务设计阶段考虑验收

在设计AI任务时,优先考虑可自动化验收的任务分解方式,明确可量化的验收标准

2 建立分层验收体系

自动化验收负责基础质量把关(格式、逻辑、测试),人工验收负责高层次评估(创新性、适配性、审美)

3 持续优化验收规则

将人工验收中发现的通用性问题逐步转化为自动化检查规则,扩大自动化验收的覆盖范围

4 提升人工验收能力

通过持续学习和经验积累,提升验收者的专业判断力和审美品味,建立系统化的验收方法论



结果验收案例 - MCP多节点Transport实现



AI知识盲区的挑战

在Agent平台做MCP接入时，需要AI编写支持多节点的MCP Transport。尽管让AI搜索了MCP相关文档和代码，但死活写不对。

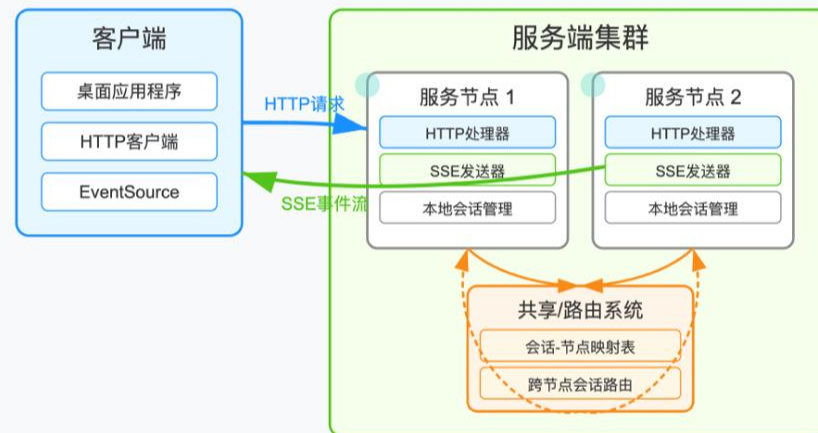
1 初始尝试：AI基于搜索结果编写代码，但理解不准确

2 反复调试：AI多次修改仍无法满足多节点架构需求

3 深度指导：提供具体的实现思路和架构设计理念

4 最终成功：在细致指导下，AI最终完成正确实现

关键经验：对AI产出需认真review设计实现及运行效果，遇到知识盲区时需要耐心的教导和具体指导



通信流程说明:

1. 客户端通过HTTP请求连接到任一服务节点（图例中为节点1）
2. 服务节点将会话信息注册到共享/路由系统
3. 如果SSE连接在另一节点上（图例中为节点2），通过路由系统查询会话
4. 最终，正确的节点通过SSE连接向客户端推送结果

MCP (Model Context Protocol) 多节点架构示意图



认知鸿沟：为何结果千差万别？

AI：能力的"放大器"而非"拉平器"

AI不是消除差距的工具，而是 **放大个体能力差异** 的催化剂

个人能力



AI放大



成果差异

！ 拉开差距的关键因素

🧠 认知深度

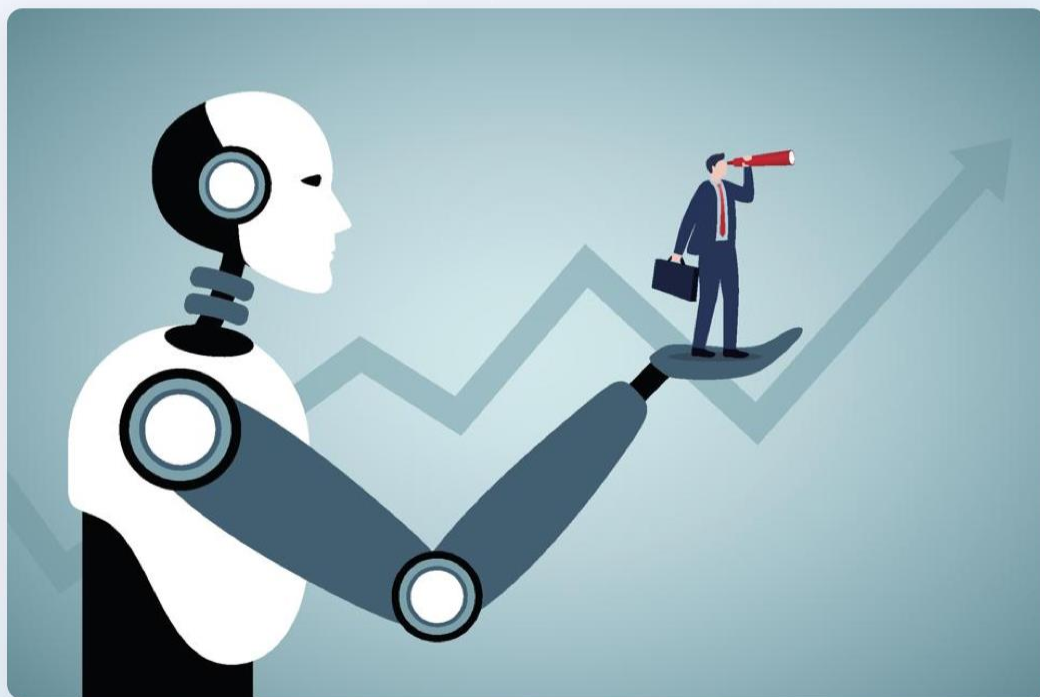
对问题本质的理解程度，决定提问质量与引导方向

💡 专业品味

辨别优质输出的能力，影响最终成果筛选与优化

📁 经验积累

领域知识沉淀，提升问题拆解与方案评估能力



提示词差异示例：待办管理应用

小白用户 非产品技术背景

典型提示词

帮我做个待办事项的APP

- 表达模糊，缺乏具体需求
- 没有功能边界定义
- 依赖AI自行猜测和补充
- 结果随机性大，难以满足预期

产品经理 产品思维导向

典型提示词

设计一个个人待办管理应用，需要支持：1) 任务的增删改查 2) 优先级设置（高中低） 3) 截止日期提醒 4) 任务分类标签 5) 进度跟踪 6) 数据导出功能。目标用户是职场人士，追求简洁高效的操作体验。请给出产品功能规划和用户体验设计建议。

- 功能清单详细明确
- 考虑用户场景和体验
- 有明确的目标用户定位
- 注重产品完整性和商业价值

技术人员 技术实现导向

典型提示词

用React+TypeScript构建一个Todo应用，要求：1) 使用Context API做状态管理 2) LocalStorage持久化数据 3) 支持拖拽排序（react-beautiful-dnd） 4) 响应式设计（Tailwind CSS） 5) 添加单元测试（Jest+RTL） 6) 使用自定义Hooks封装逻辑 7) 支持主题切换。请提供完整的项目结构、核心组件实现和测试用例。

- 技术栈选择明确具体
- 关注代码架构和工程化
- 考虑性能、测试、可维护性
- 实现细节描述详尽精准



雕刻的艺术：从AI中解放作品

🧠 经典雕刻理念

米开朗基罗相信，**雕像本身就藏在大理石中**，雕刻家的任务不是创造，而是通过精准的凿击，将已经存在的完美作品从石头中解放出来。

"每一块石头里都沉睡着一座雕像，雕刻家的任务就是去发现它。"

🔧 Vibe Coding/Working

同样地，**完美的解决方案已经存在于AI的知识海洋中**，我们的任务是通过精确的提示词"凿击"，将最适合的方案从AI中"雕刻"出来。

🏔️ 传统雕刻

原料：大理石
工具：凿子、锤子
技艺：精准的力度与角度

🔗 Vibe Coding

原料：AI知识库
工具：提示词、对话
技艺：精准的表达与引导



米开朗基罗《大卫》- 从石头中解放的完美艺术

行业影响 —— 04 软件世界的"板块漂移"

AI驱动下软件行业格局的深度重塑与变革



AI驱动的市场增量与需求释放

生产力革命的双面性

AI显著提升开发效率，释放了大量 **原本被压抑的需求**，市场规模快速扩张，但 **需求满足方式** 发生根本性变化

市场悖论

市场有增量，需求在爆发
职位可能减少，传统角色在消失
增量市场由"用户+AI"直接消化

市场变化三阶段

AI前时代

开发成本高、周期长
大量需求被压抑，市场供给不足

AI赋能期

开发效率大幅提升
压抑需求开始释放，市场快速扩张

新均衡态

用户+AI直接满足需求
中小需求去中介化，自助解决



行业格局的重新洗牌



传统行业结构

- ★ 高级架构师/技术专家
- 👥 中级开发工程师
- 👤 初级程序员
- 🎓 实习生/培训生



AI时代新结构

- 👑 顶级专家 + AI
- ✗ 中级岗位被压缩
- ✗ 初级岗位被替代
- 🧑‍🎓 非专业人士 + AI

新时代人才结构



少量顶部专业人士 + AI

复杂架构设计、技术决策、创新突破



大量非专业人士 + AI

直接使用AI工具满足中小规模需求



个人与组织的适应性变革



个人能力转型



从：垂直专业能力

深度钻研单一技术栈

↓ 转型 ↓



到：综合能力 + AI领导力

跨领域思维 + AI协同能力

🔑 核心能力要求

T型人才：广泛的知识基础 + AI协同的深度实践



组织结构演进



从：传统层级结构

多层管理，信息传递链条长

↓ 演进 ↓



到：扁平化 + AI Native

减少中间层，AI直接参与决策

🏢 AI Native组织

AI不仅是工具，更是 **组织的智能成员**，参与从决策到执行的全流程

05 超级个体的崛起

AI时代个人与组织的新机遇与挑战

📍 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 AIOps
- 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
- LMOps
- 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

