

# 从守着 AI 干活， 到验收 AI 的活

VibeCoding vs AI Native Git : 研发效率的范式跃迁

# 你一定干过这件事

1. 打开 IDE，新开聊天窗口
2. `Ctrl+A` 全选代码 → `Ctrl+C`
3. 切到 AI → `Ctrl+V`
4. 等回复.....发现少了上下文
5. 再切回去复制另一个文件
6. AI 说：“请提供更多上下文”
7. 心态崩了 💣



禁止关机

不是因为怕丢数据  
是 IDE 关了 AI 就断了

全行业的日常：人肉 Copy-Paste 工程师 🦴

# VibeCoding：听起来很美，用起来很累

理念

用自然语言指挥 AI 写代码  
不用写代码了 🤖

现实

你得守在屏幕前  
next next next 😞

**痛点 1** 喂上下文：项目结构、技术栈、编码规范.....  
每次新对话重来一遍

**痛点 2** 看输出：AI 改错地方了？你得帮它改

**痛点 3** 做 Git 操作：建分支、提 PR、解冲突，还是  
你的活

**痛点 4** 看 CI 日志：挂了？你来看日志定位修复

「AI 写了代码，但提交的是你。  
你是 AI 和 Git 之间的肉身 API 网关。」

# 根本瓶颈——AI 没有记忆，人机必须同步



AI 每次对话都是空白的

关掉窗口，一切归零。不知道项目结构、历史决策、代码风格

全行业的方案 → 给失忆症贴创可贴

超长上下文

账单爆炸 📉

RAG

检索到废弃代码

记忆框架

换个项目又重来

瓶颈不在 AI 不够聪明  
在于 AI 没有住进代码仓库  
——你的项目上下文全在 Git 里，但它看不见。

# AI Native Git : 让 AI 住进仓库

核心思想 : Git 仓库是 AI 最完美的外部记忆



Issue

= 需求记忆

做什么、为什么做



PR

= 决策记忆

怎么做、Review 了什么



CI

= 质量记忆

跑没跑通、哪里挂了



代码库

= 完整记忆

# 核心差异——同步 vs 异步

维度	 VibeCoding	 AI Native Git
你要做什么	守在 IDE 前，一步步引导 AI	说一句话，去干别的
AI 上下文	你手动喂，每次从头来	Git 仓库自动提供
人机关系	人机绑定，你走它就停	人机解耦，各干各的
Git / CI	你来	NPC 自动
你的角色	监督执行 + 亲自动手	关键决策 + 最终验收
AI 是什么	帮你写代码的工具	放大你决策力的杠杆
并行能力	1 件	N 件——@ 多个 NPC 并行

# 场景——修 Bug

## VibeCoding

1. 开 IDE
2. 贴 Bug 描述 + 相关代码 + 项目结构
3. AI 写修复
4. 手动建分支 → 提交 → 推送
5. 建 PR → CI 挂了
6. 看日志修 → 再推.....

30-60min

## AI Native Git

1. Issue 里 @NPC : 这个 Bug 帮修一下
2. NPC 自动读库、定位、修复
3. 建分支、提 PR、CI 通过
4. 你 Review → Merge

< 2min

 时间  
↓ 95%+

 上下文  
零成本

 流程  
全自动

 价值  
描述+验收

# 场景——新功能开发

## VibeCoding

1. 喂需求文档 + 技术设计 + 代码规范 + 现有代码
2. 一段段让 AI 写
3. 自己拼装
4. 手动提 PR
5. Review 意见手动改.....

1-3h

## AI Native Git

1. @NPC : 参考这个设计文档，实现注册功能
2. NPC 自动读库、理解架构、完整实现
3. 提 PR → 你 Review → NPC 自动迭代 ✓

< 5min



时间

↓ 97%+



架构  
自动理解



迭代  
零摩擦



并行  
一个人当一支团队

# AI 作为放大器

异步释放的不是时间，是决策力。

## 🎯 同时指挥多个 NPC 并行作战

- 前端一个 NPC
- 后端一个 NPC
- 测试一个 NPC
- DevOps 一个 NPC

省下来的精力做 AI 做不了的事

做关键决策 — 方向、架构、取舍

攻克 Key Point — 人判断方向，AI 快速实现迭代

AI 不是替你干活的工具，  
是放大你决策力的杠杆。

你决策的质量，决定了团队产出的天花板。

# 新流程

1. 开 Issue 描述需求



2. @NPC 自动执行



3. 收到通知



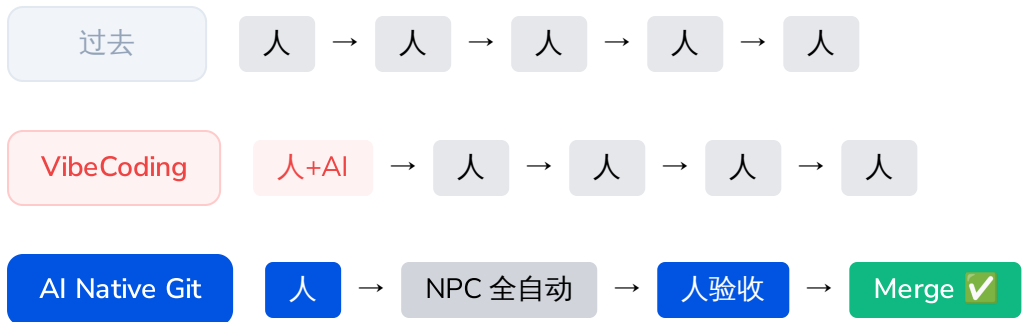
4. Review



5. 完事 ✓

指标	VibeCoding	AI Native Git	改进
Bug 修复人工时间	30-60min	< 2min	↓ 95%+
功能开发人工时间	1-3h	< 5min	↓ 97%+
并行任务数	1	N	N 倍
精力分配	80%执行+20%决策	10%验收+90%决策	质变

# 团队协作的跃迁



AI 帮了忙，但流程没变

流程本身变了

不再需要全员盯着屏幕等 AI 输出——  
每个人只在自己负责的环节介入

VibeCoding : 你守着 AI 干活。

AI Native Git : 你验收 AI 的活。

AI 不是替你干活的工具，是放大你决策力的杠杆。

AI Native Git , 启动。🚀

# Q & A

来吧，尽管问 