```
// Initialize narrative engine
const narrativeState = new NarrativeEngine.State({
    protagonist: 'User',
    worldType: 'Fantasy',
    coreConflict: null,
    plotPoints
});

function gene
    // Logic
    return Pr
}

const initialPrompt = generateScenePrompt(narrativeState);
console.log(initialPrompt);
```

帮不会写故事的人写出一个游戏剧本

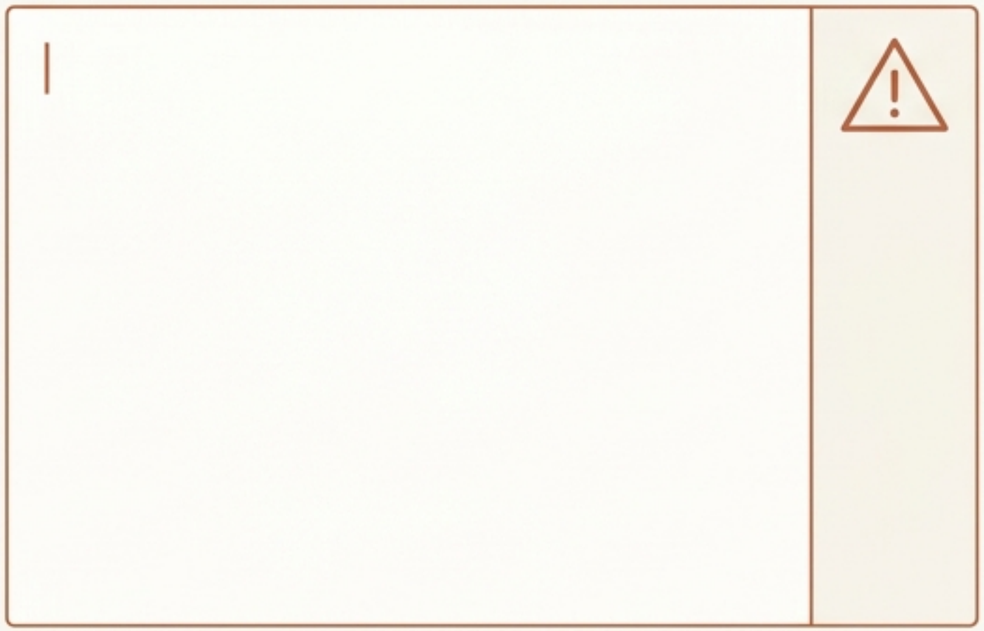
AI 作为「陪伴式」编剧的从零构建指南

独立开发者的两极困境：能搭建复杂的战斗系统，却在「主角是谁」前彻底卡壳

代码的困境

拥有明确的报错机制，Bug 可以被精准定位和修复。

故事的困境



缺乏切入点，思维一旦卡住，只能在空白面前无谓消耗。

为什么依靠传统教程无法解决游戏剧本的创作阻碍

	传统剧本教程	AI 创作伙伴
最终目标	试图培养你成为一名专业编剧。	**产出能撑住当前游戏的剧本——「够用就行」。**
反馈机制	单向输出，死板教条。无论写什么，只提供固定内容。	**实时动态审查**，指出前后矛**盾与平淡处**，提示伏笔。
适用起点	默认一套固定的学习大纲。	**结合草案**进行**实时辅导**，**因人而**异。

核心设计一：拒绝吓退新手的「动态校准」机制

AI 提问：你能说出游戏的核心冲突或世界观规则吗？

NO

降低辅导门槛，从最感性的问题开始：你想玩一个什么样的游戏？

YES

跳过基础发散，直接从「剧本结构」切入。

不预设专业门槛，根据创作者当前的水平，动态调节辅导力度。

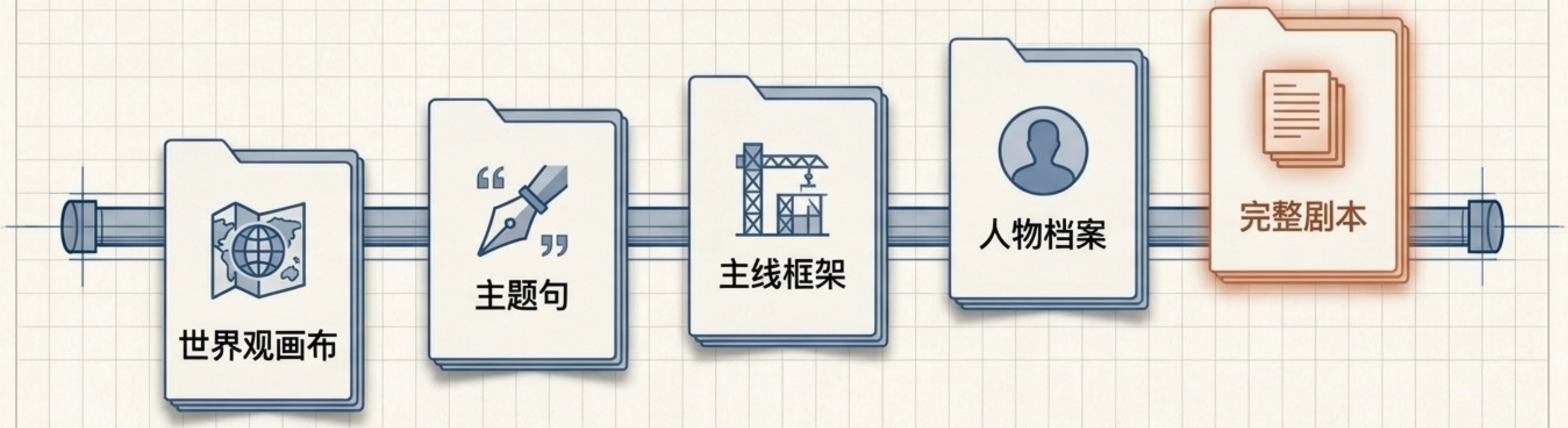
核心设计二：游戏特有的「世界观先行」原则

空间的本质：游戏不是背景板，而是玩家要真正置身其中的空间。

沉浸感来源：世界观的密度直接决定了游戏的沉浸感。

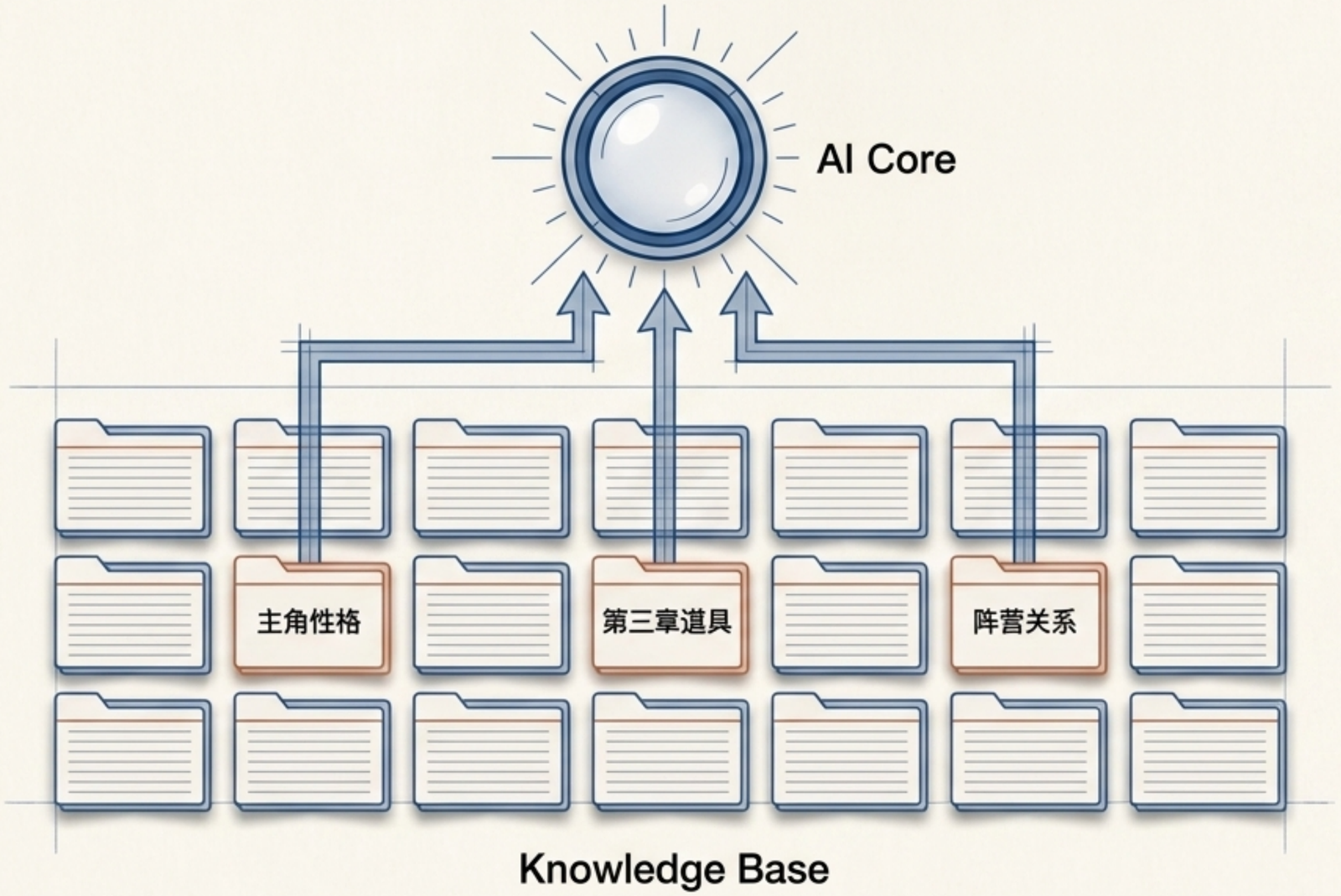
自然生长：当物理与社会法则定型后，把人物放进去，故事自然就会长出来。

核心设计三：用「阶段性交付物」终结无效的空泛对话



拒绝毫无沉淀的闲聊。每一个阶段都必须产出具象化文档，
让创作者看着进度一路积攒，消除焦虑感。

技术攻坚一：解决长文档中致命的「设定一致性」问题



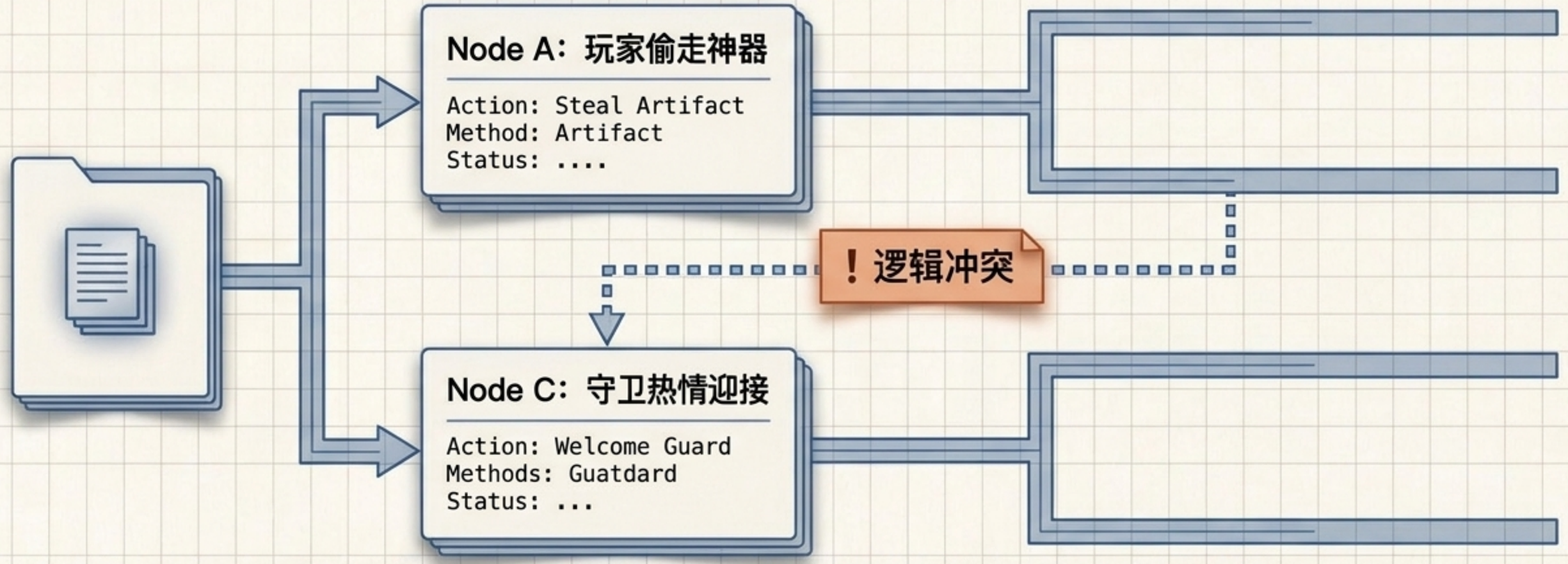
设定的失忆

剧本包含数百个要素。写到第十章时，极易忘记第三章随口提过的细节。

动态检索注入

AI 在后台维护独立知识库。动笔前精准检索背景并注入 Context，确保前后不矛盾。

技术攻坚二：驾驭交互叙事中复杂的「分支因果逻辑」

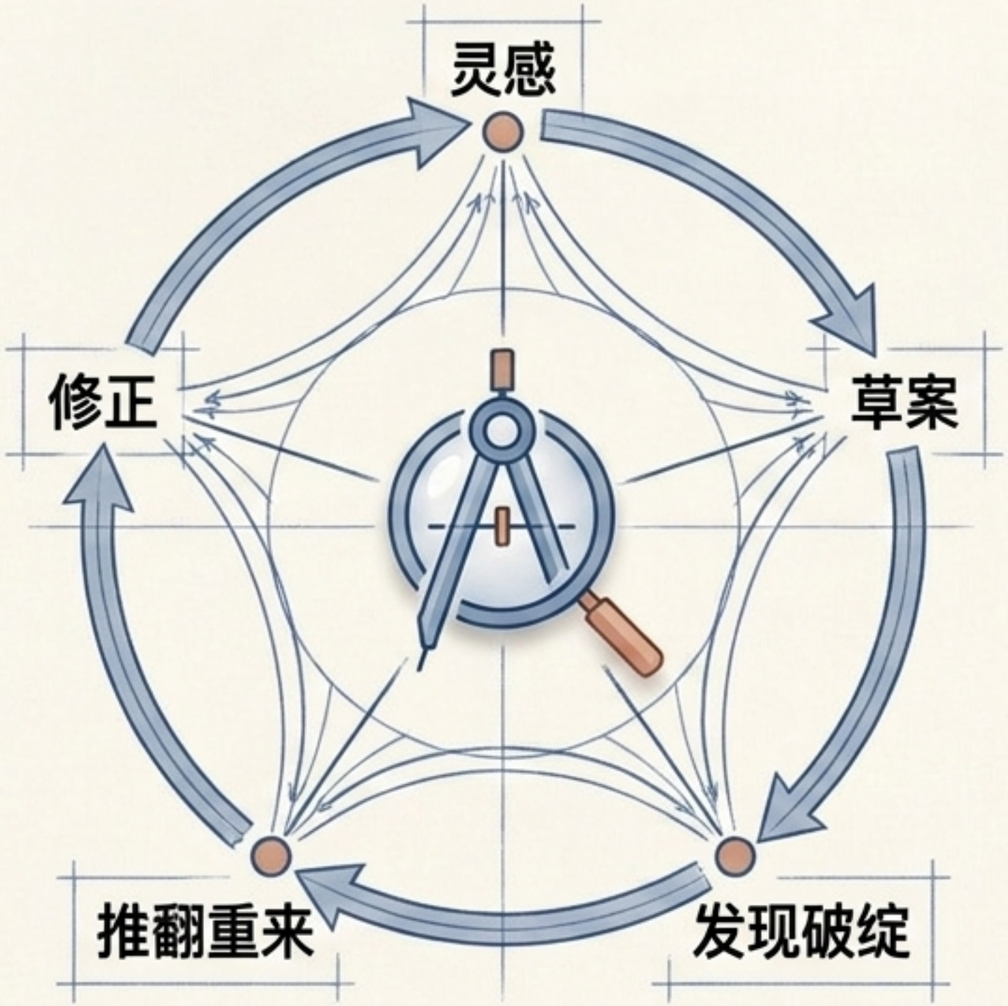


- 不止于多写几条线：分支剧情的核心难点在于跨线之间的因果关联与内部逻辑自洽。
- 逻辑校验引擎：AI 必须审查每个选择节点——这个选择合理吗？后续的剧情后果接得上吗？

从「内容生成器」到「创作伙伴」的范式转移

失效的指令模式

单向的「你提需求、AI直接生成」行不通，因为　创作者在一开始无法提清清所有不断变化的创作需求。



真正的陪伴式 AI

听懂话锋，接住话题。陷入死胡同时挑战你的错误方向；卡壳时提出正确问题，捋清未成形形的想法。

剧本还是你自己写，AI 负责给你搭脚手架。

释放开发者的技术潜力，用陪伴式 AI 跨越叙事鸿沟。