

# 投资的门槛换了

零代码 · AI 原生投资者指南 (第 1 章)

# 瓶颈换了位置，没有消失

## 旧时代 (1995-2023)



## AI 原生时代 (2026)



# 范式转移：旧约束 vs. 新约束

	旧范式	AI 原生范式
前置条件 (Prerequisite)	会写代码 / 彭博终端	Agent 访问权 / API Key
分析耗时 (Time to Analyze)	8 小时 (手工清洗/建模)	< 1 分钟
核心门槛 (Core Bottleneck)	交付机制 (搭管道)	过滤机制 (判断对错)
边缘优势来源 (Edge Source)	触觉熟悉度 (Tactile grind 偶然发现)	刻意对抗 (Red-teaming / Pushback)
典型死法 (Typical Failure)	疲劳、确认偏差、跳过脚注	陷入 Echo Chamber, 确信地犯共识错误

## 你 (Portfolio Manager)

[保留判断]  
[不读完整 10-K]  
[决定信任谁的工作]

### Agent: 分析师

全天候处理数据, 产出  
研究 Memo

### Agent: 红队

寻找论点漏洞, 挑战假  
设

### Agent: 执行者

跑稳定 Workflow, 监  
控信号

真正的杠杆: Delegation without Abdication (委托但不放弃判断)

# 对冲基金类比：你与你的单人舰队

## 传统对冲基金

角色定位

**PM（决策） + Junior（执行）**

产出速度

**几天/几周**

信任建立

**多年校准，知道谁激进谁保守**

## 你的单人舰队

角色定位

**PM（你） + Agents（执行）**

产出速度

**几秒/几分钟（受限于 API 成本）**

信任建立

**永远在和没有持续记忆的新团队合作**

## ⚠️ 纠错机制（The Gap）

优秀分析师会带着数据 Push Back

Agent 默认妥协，不主动反驳，极易形成信息茧房。

# 程序员的习惯迁移：并非无缝对接

## Debugging 本能

Metadata  
structural

### ✓ Zone A

优势：隔离变量，改假设看结果。

### ⚠ Zone B

限制：金融 Bug 没有 Error Message。偏了 40% 的 DCF 只要叙事好，看起来也完全合理。

## Version Control (版本控制)

Metadata  
structural

### ✓ Zone A

优势：记录 Thesis 演变，防止记忆篡改。

### ⚠ Zone B

限制：如果你不知道 Thesis 应该包含什么元素，Diff 是毫无意义的。

## Workflow 设计

Metadata  
structure

### ✓ Zone A

优势：把模糊任务拆解为带验证节点的 Pipeline。

### ⚠ Zone B

限制：市场有反身性反馈环，执行本身（买入）会改变前提条件。

## 抽象的舒适度

Metadata  
strucluze

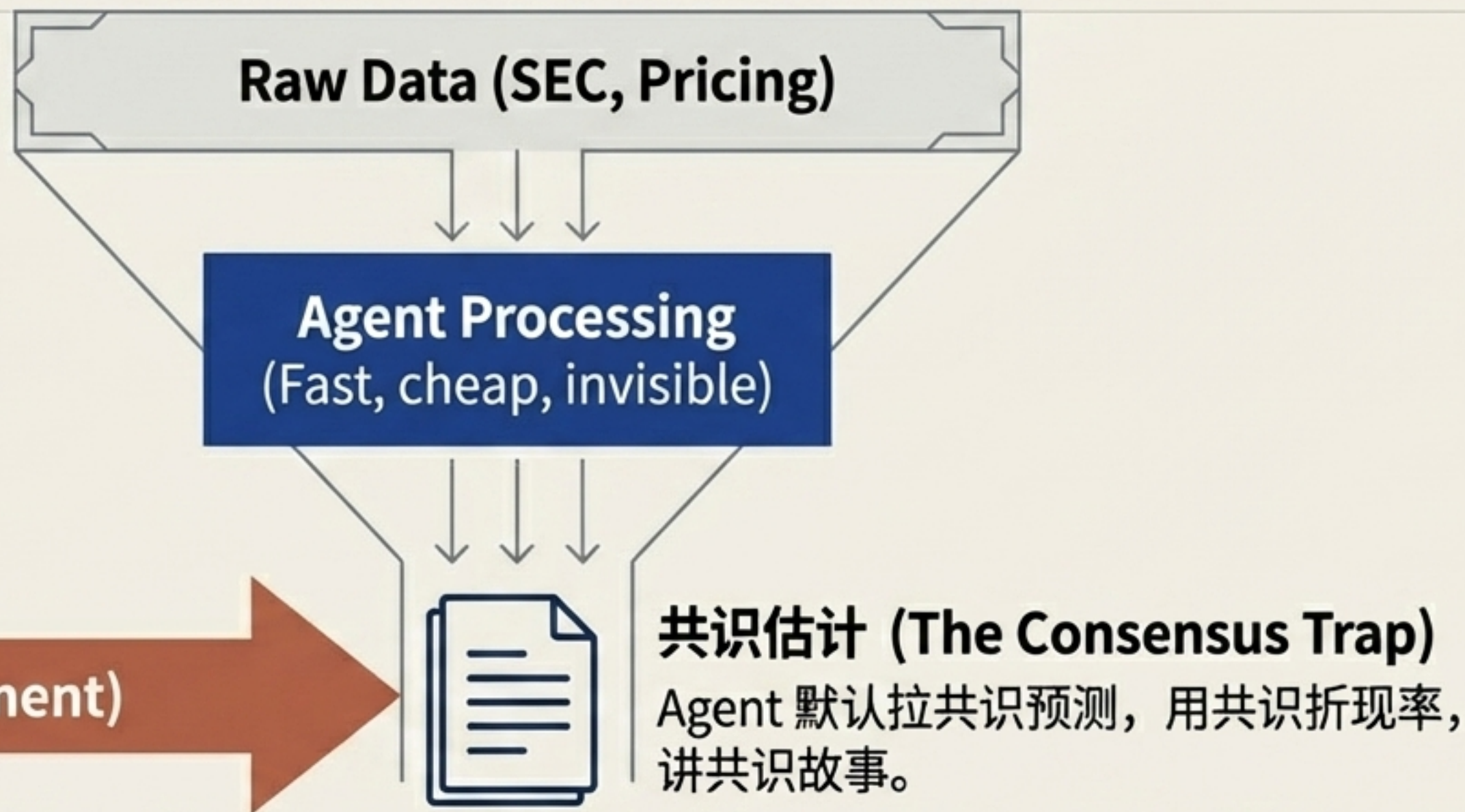
### ✓ Zone A

优势：习惯与简化现实的模型打交道。

### ⚠ Zone B

限制：在软件里你控制抽象；在市场里，系统会用真金白银惩罚过度简化 (Oversimplification)。

# Flow of Edge



如果每个人都能在十分钟内让 Agent 产出完美的分析，分析本身就不再是 Edge。  
你的 Edge = 你选择去对抗的那些假设。

# 课程核心：两条轨道，交织进行

## Track A: 投资基本功

几十年来 load-bearing 的思维模型（DCF、风险、第一性原理）。

If learned alone: 拥有基础，但在这个时代毫无执行速度。



两件事必须同时学。

## Track B: Agent 舰队指挥

2026 原生技能（角色分配、Fleet 架构、Workflow）。

If learned alone:  
高效地产出关于你不理解的公司分析。高效的 Garbage 看起来像 Signal。比没有 Fleet 更糟。

# Track A · 投资基本架构



## Lens 1 (第 3/4 章) : 第一性原理 & 报表模式

- 股票到底是什么？三种回报来源。
- 将三张报表视为透镜（业务结构、脆弱度、真假），学习模式识别而非纯会计。

案例：Enron, Costco vs Walmart



## Lens 2 (第 5 章) : 估值即叙事 (Damodaran 框架)

- DCF 里的每个数字都是一个故事选择。
- 寻找估值偷偷依赖的假设。



## Lens 3 (第 6/7 章) : 风险与 Thesis 纪律

- 风险不是波动率 (LTCM 失败机制)。
- 防弹 Thesis 的六个要素与 Pre-mortem 技术。

# Track B · Agent 舰队指挥学

解决 'Agent 变强前不存在' 的新问题。

## 第 2 章



### 1. 分析师

默认模式，产出研究



### 2. 红队

在市场攻击你之前，先攻击你的 Thesis



### 3. 老师

帮你评估输出，填补概念盲区



### 4. 执行者

在 Scale 上跑稳定 Workflow

## Fleet Architecture - 第 8/9 章

### 5 个核心设计模式：

- Delegation
- Verification
- Iteration
- Composition
- Failure Detection

**⚠️ 核心原则：**如果舰队达成高度一致同意，这是一个警告信号。

# 协议规则： 如何阅读本课程

■	[!] 这不是博客系列	高密度，无法 Skim。10 章，总耗时预期 30-60 小时。
■	[!] 把 Agent 当老师用 (Meta 指令)	遇到不懂的概念 -> 用自己的话复述给 Claude Code -> 让 Agent 找 Gap -> 迭代直到无死角。
■	[!] 重读是预期中的	特别是第 4-6 章（公司、估值、风险）。第二遍才是框架结晶（Crystallize）的地方。
■	[!] Workshop 是强制的	没做 Workshop 过掉一章，等于读了编程书没敲代码。



# Workshop · 你框架的第一像素

最后，你拿走的不是 Stock Pick，不是模型。  
是一个你能 Defend 的框架，和运行舰队的能力。

任务说明 | 时间：30-45 分钟 | 工具：Claude Code, Codex 或 Cursor

目标：选一只你真正在意或持有的股票（拒绝热点），建立后续课程需要嫁接的基石。

# 实操手册：搭建第一像素

The First Pixel.md

Agent 生成 3 段 Bull Case (看涨) & 3 段 Bear Case (看跌)。

## 纯手工标记 (不查资料)

[✓] 标记同意的 Claim

[✗] 标记不同意的 Claim

[?] 标记无法判断的 Claim

### 框架之锚

我最强烈不同意的 claim 是 [具体论点]，我这么相信是因为 [具体原因，例：最大客户正在自建替代方案]。

NOTE: 保存文档 (Ticker+Date)。第 7 章重新打开，与未来的自己 argue。

# 真正的稀缺品

把模糊的问题精炼成精确的问题，是程序员的特长。  
这种能力可以迁移，但需要刻意努力 (Deliberate Effort)。



最稀缺的投资技能从来不是找到答案。  
是知道该问什么。