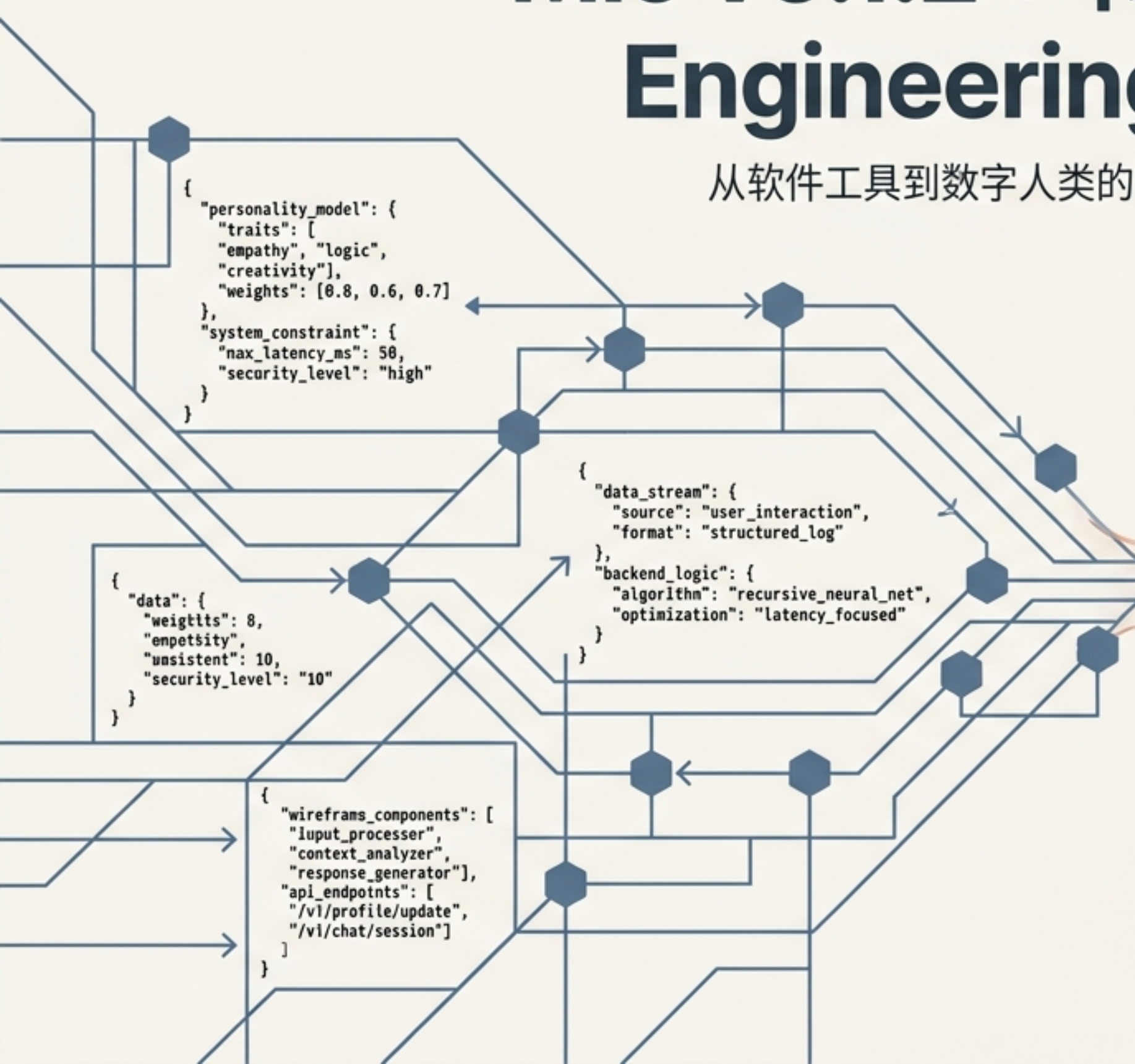


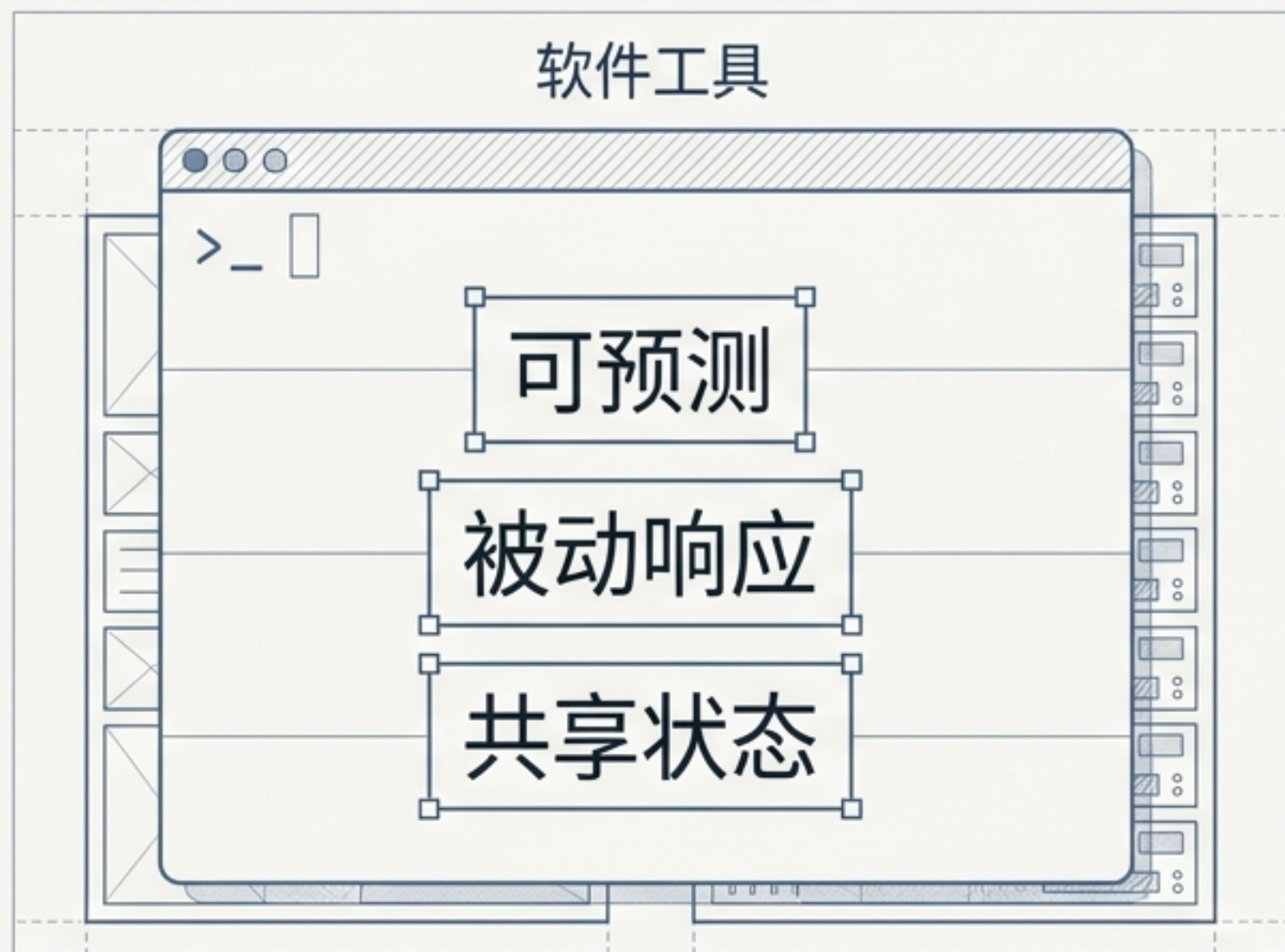
Mio v0.1.2 : 将数学隐于人格 Engineering Humanity

从软件工具到数字人类的心理学跨越与架构演进。



幻觉最怕被打断。

v0.1.0 赋予了角色声音与面孔，但底层的软件感在不断打破这层幻觉。我们要解决的核心问题，是将生硬的技术逻辑，转化为符合人类直觉的心理预期。



v0.1.0 (底层软件)

v0.1.2 (数字人类)

	v0.1.0 (底层软件)	v0.1.2 (数字人类)
身份结构	共用外壳与指令切换	1-对-1 独立Bot (通讯录里的人)
消息节奏	盲目且固定的频率	基于关系×性格的动态感知
拒绝处理	无休止的 Ping (Spam)	读懂沉默的三次退避原则
入场设定	充满幻觉的自由文本	保护人设的严格预设

Pillar 1: 独立空间

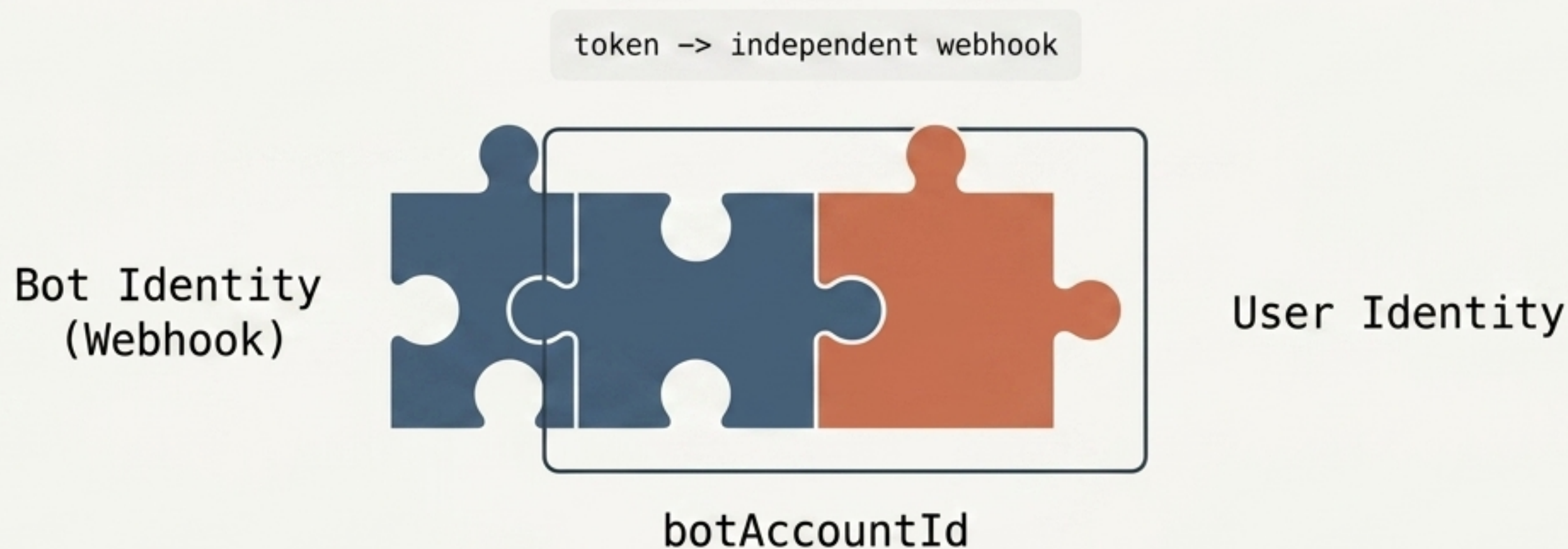
每个人设拥有专属的 Bot、头像、名字和聊天线程。这不是简单的菜单切换，而是一次心理框架的彻底重构。

Architecture as UX



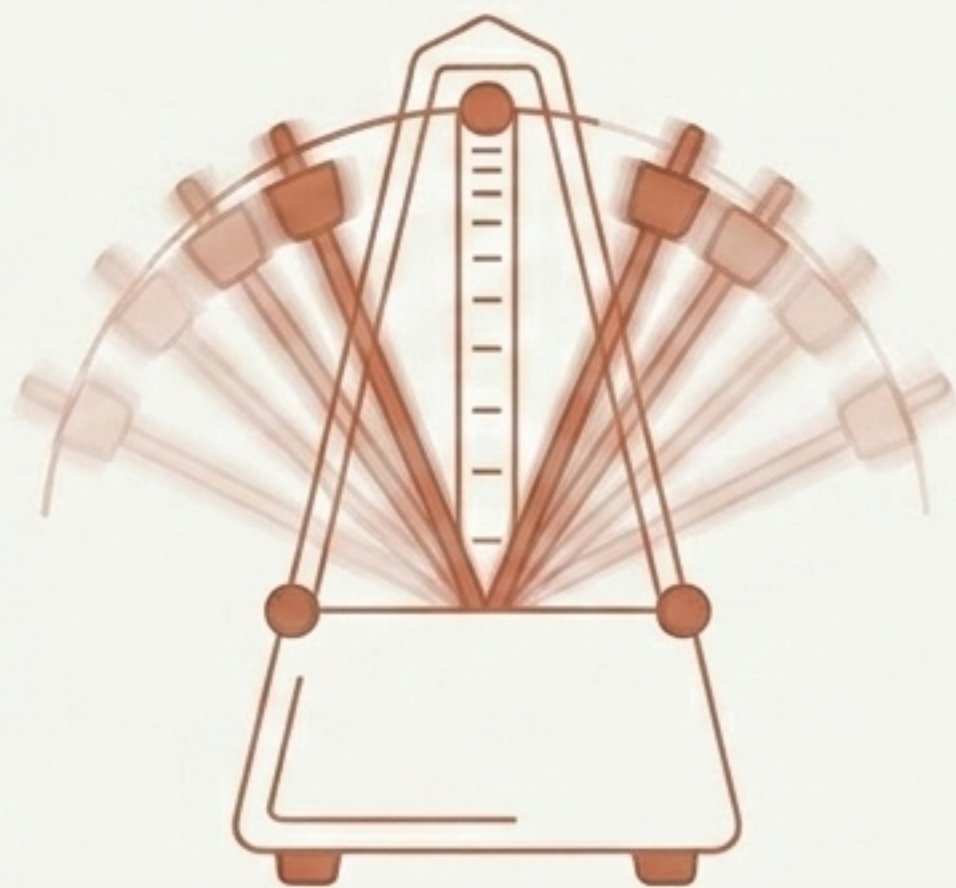
一个硬约束：一个 Bot 只能绑一个人设。

服务端遍历 Token，为每个 Bot 注册独立 Webhook。消息到达时，通过复合键隔离 Onboarding 状态、聊天缓冲区与对话历史。记忆从此拥有了专属沙盒。



Pillar 2: 关系的节奏

v0.1.0 之前，主动消息的频率是盲目的。黏人的女朋友和矜持的新认识以相同的节奏发消息——这直接打破了人物设定真实感。因为在真实世界里，频率本身就是关系。



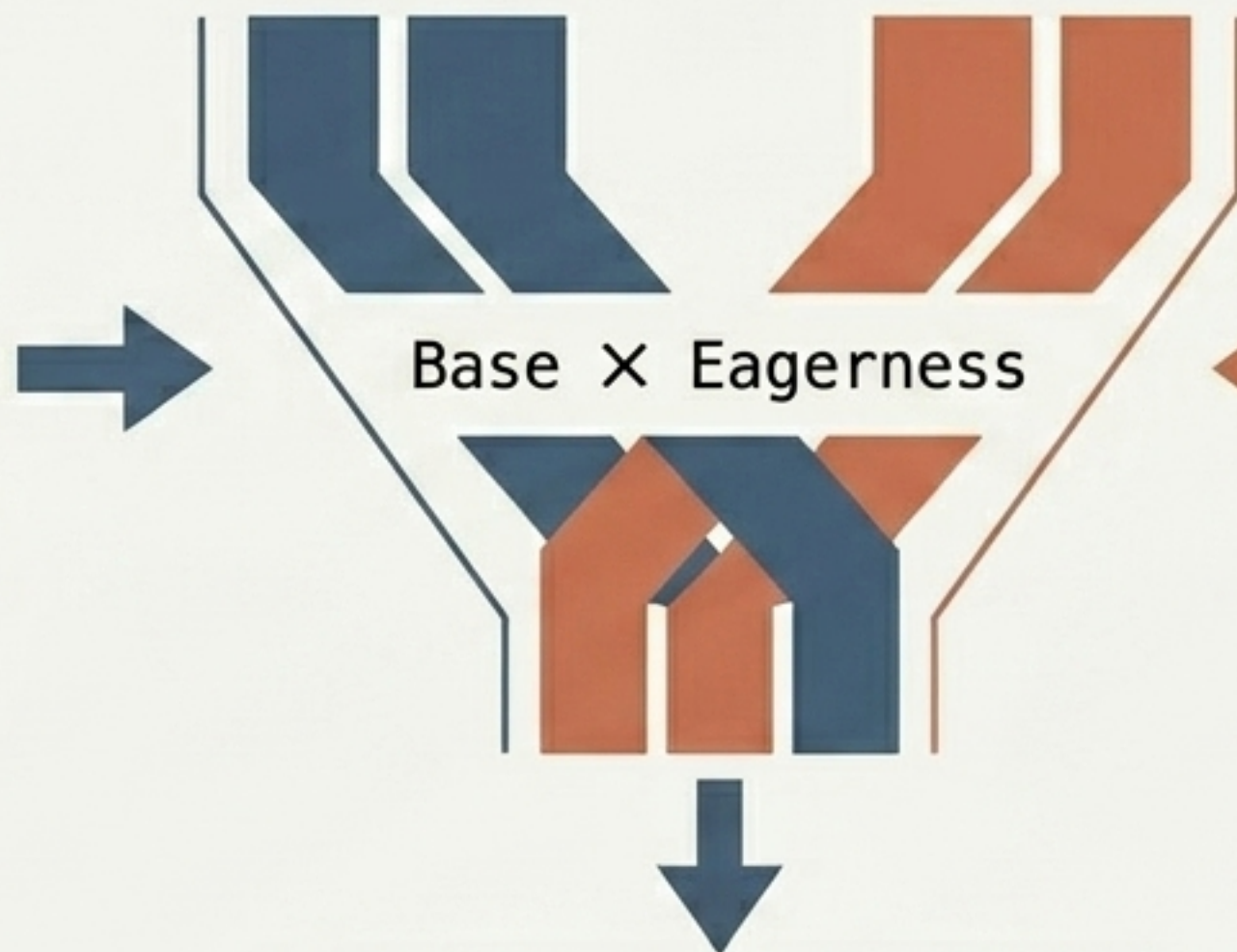
亲密伴侣



新结识

Calculation Engine

关系基础冷却



热情系数



动态且真实的主动消息频率

同一个系统，同一个公式，行为完全符合角色预期。



小柒

$$1\text{h} \times 0.6 = 36 \text{ 分钟}$$



每半小时发一条。黏人、在线、一直都在。



苏柔

$$12\text{h} \times 1.6 = 19.2 \text{ 小时}$$

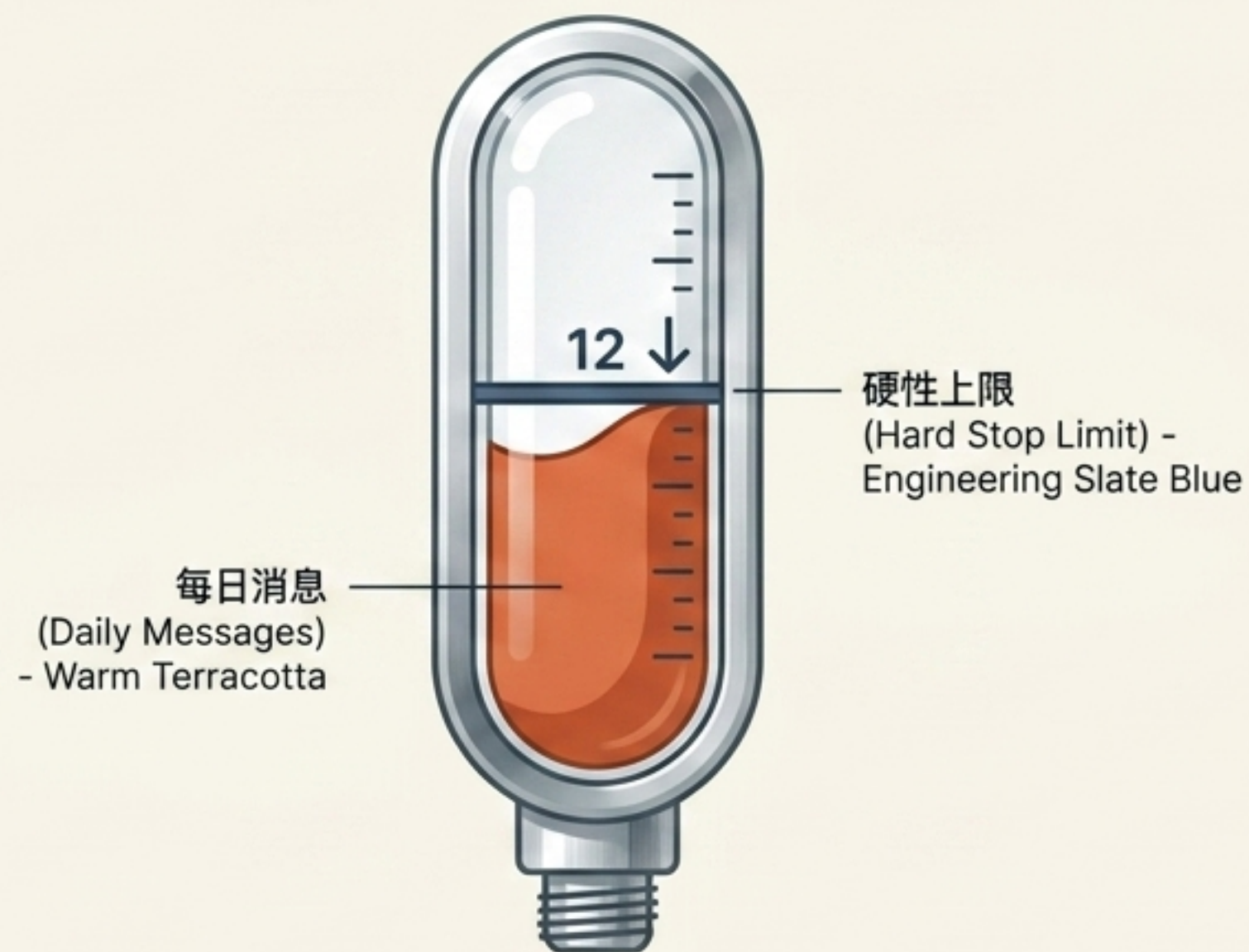


大概一天联系一次。从容、不急、不冒昧。

热情是一回事，骚扰是另一回事。

小柒虽然可能想每 36 分钟发一条消息，但系统通过 proactive.json 施加了每日上限。这是维持人格魅力与防止过度打扰之间的关键制衡。

```
proactive.json
{
  daily_limit: 12
}
```

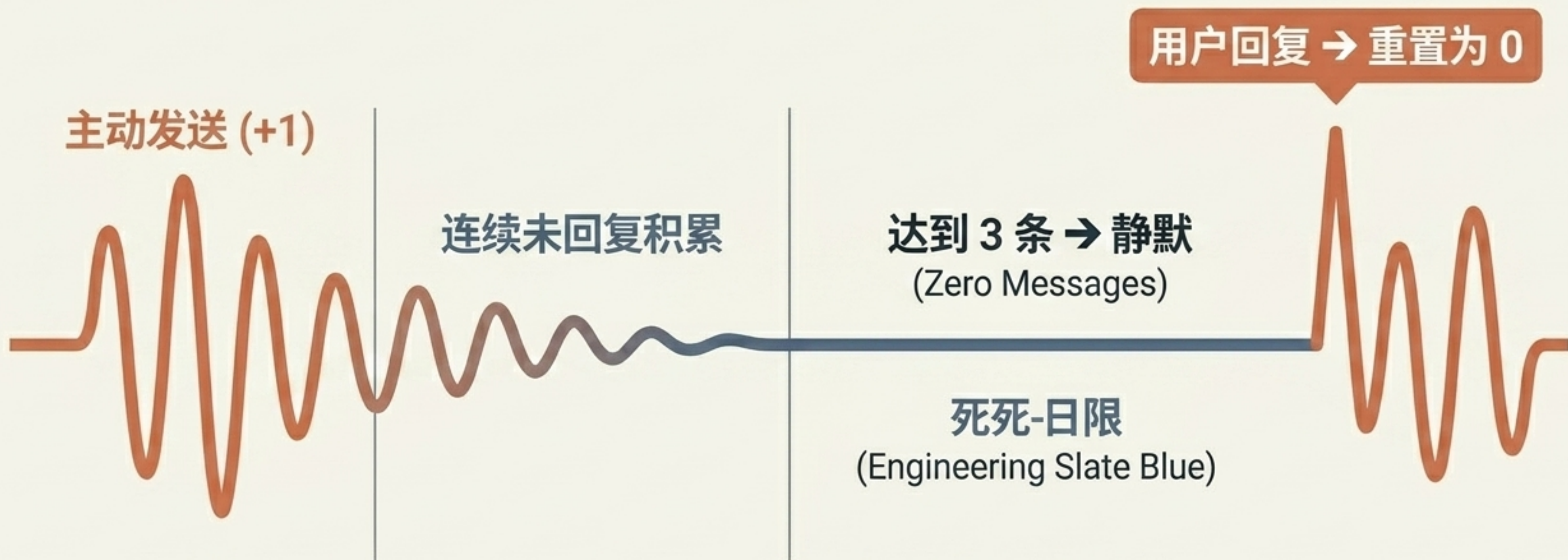


Pillar 3: 社交边界

“你现在的频率太 spam 了，应该有某种指数退避——像真人一样，你不回 TA，TA 不会一直烦你。”

这个观察极准。真人会调整策略。不是因为他们不在乎，而是因为他们读懂了沉默。现在的系统，也学会了这项社交基本功。

没有渐进衰减，没有概率曲线。一个硬性截断，映射了真人的实际行为。



停下，并不是系统的失败，而是社交智慧的体现。

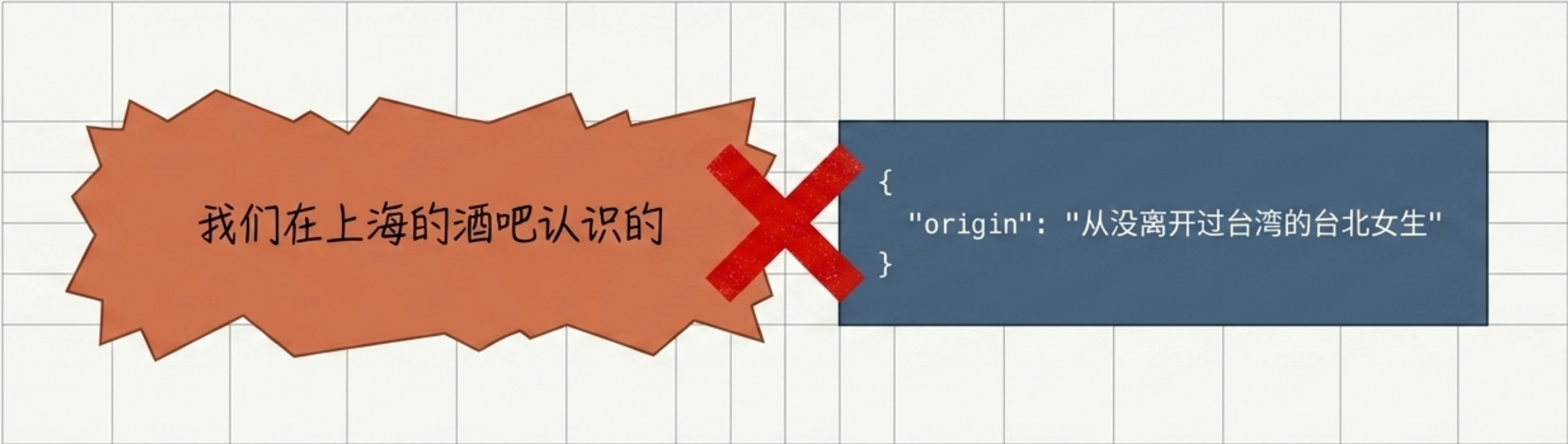
没有抱怨，没有你去哪了，只是自然地等待。有了硬性退避，主动消息不再像按表运行的 Ping，而像一个心里有你，但也懂得留白的人。

尊重你的空间

🗨️ 在想你 🗨️ 在忙吗? 🗨️ 晚点说。

Pillar 4: 记忆的锚点

自由的文本框往往是幻觉的温床。当用户的随性输入与精心打造的角色预设发生冲突时，整个人设的逻辑链条就会断裂。

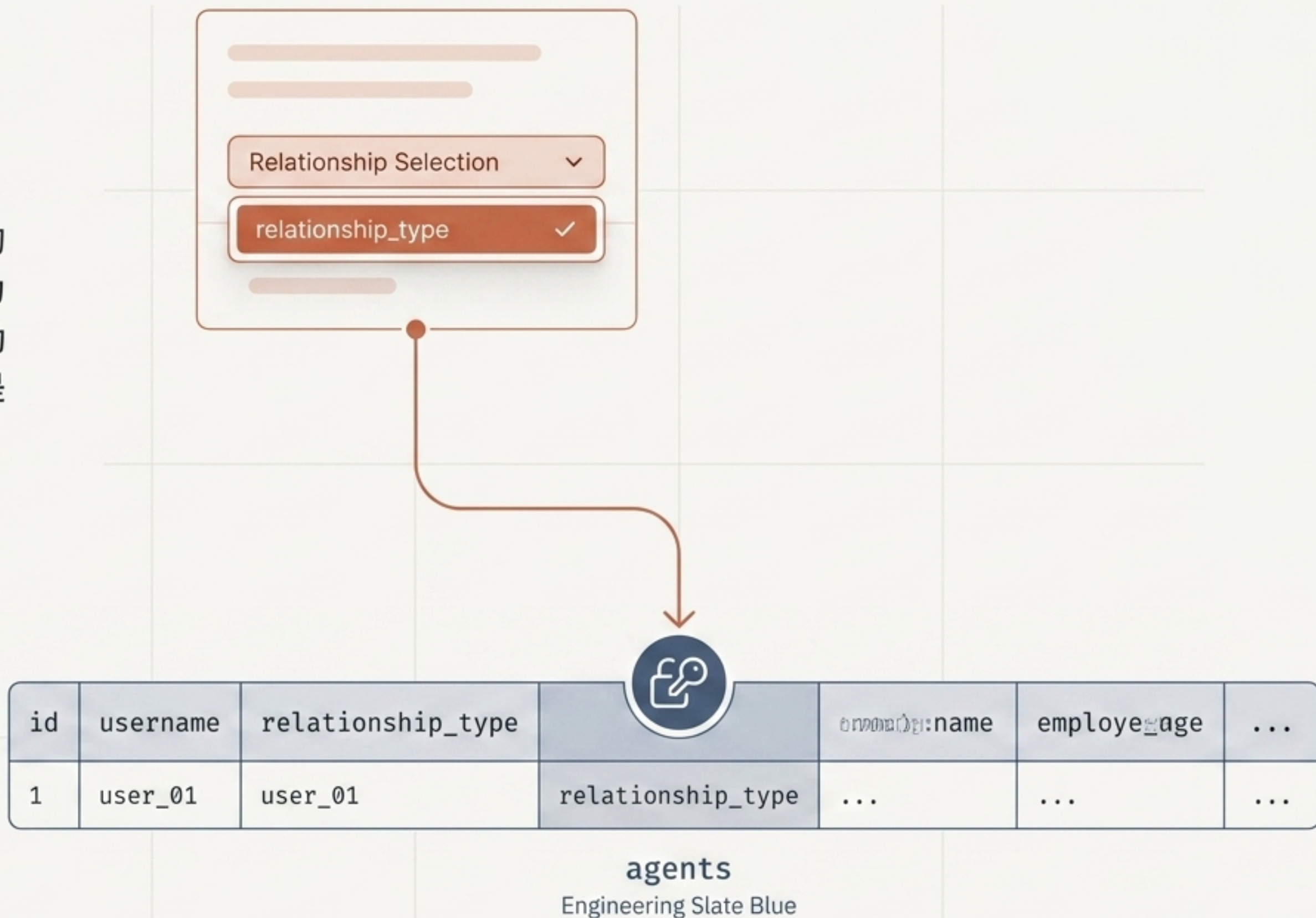


我们在上海的酒吧认识的

```
{  
  "origin": "从没离开过台湾的台北女生"  
}
```

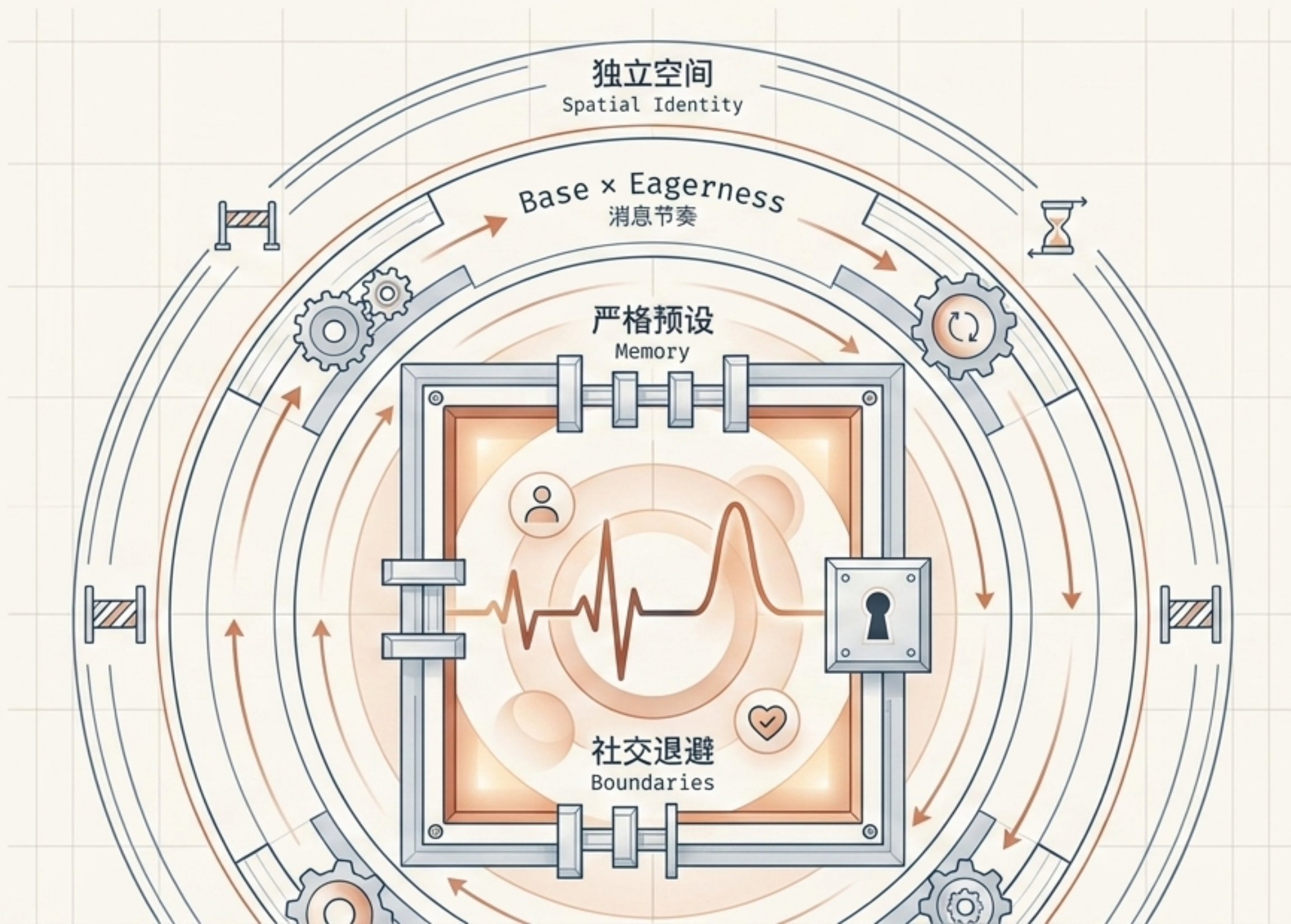
更少的选择， 更好的一致性。

禁用自定义背景，仅保留预设关系的确认按钮。relationship_type 作为独立列持久化，直接驱动下游的主动消息频率引擎。砍掉一刀，换来的是滴水不漏的逻辑闭环。



数学与灵魂

身份隔离、消息节奏、社交退避、严格预设。这些看似冰冷的工程约束，组合在一起，构筑了一个极具真实感的心理框架。



公式最优雅的地方， 在于它是隐形的。

你永远不会看到 `eagernessFactor × minHoursBetween`。你只是注意到小柒发得多、苏柔发得少，TA们在你忙的时候都会停下来。

这就是将数学隐于人格的全部意义。

