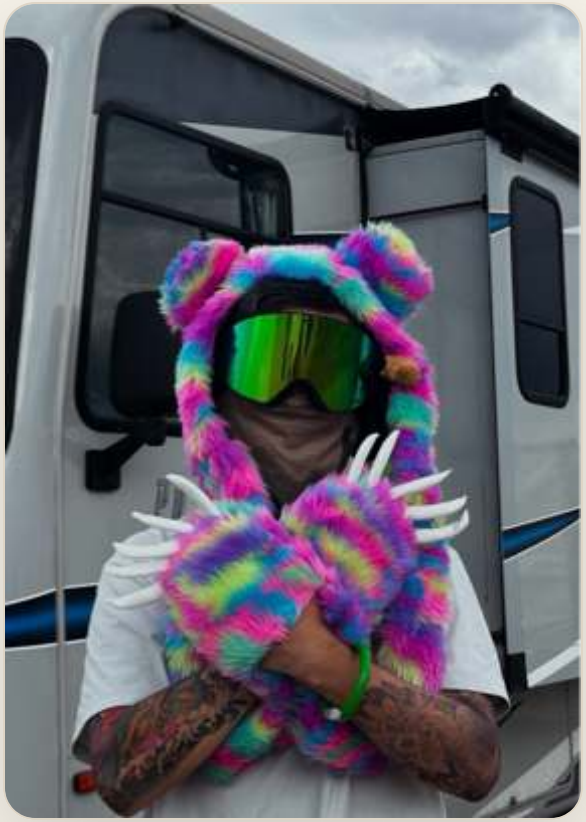
和合谷 | Syfo

AI AGENT WORKSHOP · PART A

AI Agent 带来的组织变革

执行能力快速扩张之后，组织设计、管理责任与竞争优势如何变化。

60 分钟 · 讲解 35′ · Demo 10′ · Q&A 15′



Tony Ren

长期从事 AI 研究、数据产品与全球化产品管理。

- **香港科技大学** · 计算机科学本科、硕士、博士
- **研究员经历** · 腾讯、华为诺亚方舟 AI 实验室、日本国立情报学研究所
- **一面数据 · 创始人** 2019 年被 Ascential plc 收购后并入 Omnicom；2026 年被 NIQ 再次收购。
- **前 Ascential / Flywheel Omnicom** · 亚太区首席产品官
- **反曲科技 · 创始人**

这 60 分钟的路径

从个人使用 AI，走到组织能够稳定调用执行能力。

01 · 诊断

个人提效为什么没有自动形成组织能力

看清上下文零散、效率红利外流、权责失衡三类组织损耗。

02 · 设计

组织如何编排 Agent、信息与责任

把频道、任务、Thread、Review 和权限组成可追踪的执行系统。

03 · 迁移

人的工作如何转向组织、判断与闸口

管理者重做目标、分工、Context、权利和绩效，员工学习组织执行能力。

KEY TAKEAWAY

AI 价值进入组织，要同时解决能力吸收、执行编排和人的角色迁移。

执行建议

会后选一个稳定业务闭环，明确结果、Owner、Context 和 Review 闸口。

个人提效，
不等于组织提效。

员工使用 AI 的能力越强，组织能力甚至可能越弱：个人完成更多，
组织却更难理解过程、接手工作和复制经验。

KEY TAKEAWAY	个人速度上升，只说明工具有效；可接力、可追责、可复用才说明组织能力增强。
执行建议	检查关键员工离开后工作能否继续，同类任务再来时能否直接复用上次积累。

三个原因

个人变快，组织能力却可能持续流失。

01 上下文零散化

Claude Code、Codex、WorkBuddy 的会话、判断和产物散落在个人环境，交接依赖口头复述。

02 效率红利被截留

目标、激励与任务供给不变，多出来的时间未必投入组织目标，产能扩张停留在个人层面。

03 权责不对等

组织要求更多结果，却没有同步授权、收益与风险保护，员工会回避 AI 带来的业务责任。

KEY TAKEAWAY

AI 使用越分散，组织越容易形成流程黑箱、知识私有和责任空档。

执行建议

建立公共工作现场，同步调整授权、收益分配和风险保护。

两套完全不同的能力

组织能力要求成果能够跨人、跨时间、跨任务复用。

个人 AI 能力

我能把这件事做出来

上下文	保存在个人会话、文件和记忆里
过程	只有操作者清楚关键判断
失败	靠个人经验临场补救

→

组织 AI 能力

团队能稳定重复这类结果

上下文	进入公共频道、任务、知识与标准
过程	状态、交接、Review 和证据可见
失败	有升级路径、权限边界与责任人

KEY TAKEAWAY	组织能力的最小标准，是成果能够跨人、跨时间、跨任务稳定复现。
执行建议	把关键 Context、过程判断、异常处理和验收证据纳入公共记录。

从工具到组织

价值沿着任务连续性与组织连接逐层增加。

LEVEL 1

单次问答

人问一次，AI 答一次；价值停留在局部信息处理。

→

LEVEL 2

任务委托

Agent 跨工具执行并交付；开始替代完整动作。

→

LEVEL 3

工作流

任务有状态、交接、验收与异常；结果开始稳定。

→

LEVEL 4

组织系统

角色、信息流、权利、责任和绩效一起调整。

KEY TAKEAWAY	规模化价值来自任务连续性和组织连接，单次问答只能带来局部效率。
执行建议	先把一条工作流做到状态可见、交接清楚、失败可恢复，再扩大 Agent 数量。

过去企业优化
“每个人做得更快”。

Agent 时代优化
“一个人能组织多少并行执行能力”。

KEY TAKEAWAY	执行供给扩张后，目标定义、判断、授权和验收会成为新的组织瓶颈。
执行建议	管理层开始衡量每位 Owner 能组织的并行能力，以及 Review 带宽和决策速度。

一个业务结果，配置一套可运行的执行系统。

结果

清晰业务结果

定义完成条件、时限、质量与成本。

责任

人类 Owner

设定优先级，处理权衡，承担最终承诺。

执行

Agent 组合

按研究、制作、检查、跟进等角色并行工作。

控制

协作与闸口

频道承载 Context，任务承载状态，Review 控制风险。

KEY TAKEAWAY

一个业务结果需要结果、责任、执行和控制四个要素共同成立。

执行建议

先写完成条件和责任边界，再配置 Agent、工具、频道与闸口。

真实研发例子

一次报障如何进入可追踪的研发闭环。

#syfo-doctor
统一接收现象、影响、复现线索和期望结果。

→

Han Solo 分诊
补信息、判断归属、识别紧急程度与风险。

→

研发频道承接
携带结构化 Context 转发，减少重复解释。

→

创建并认领任务
Owner、状态、依赖和下一步进入公共记录。

→

交付与回告
修改、测试、Review、部署证据和结论闭环。

入口标准化
减少“找谁”和“重新讲一遍”的等待。

过程可观察
管理者看到卡点、风险与责任，不靠追问进度。

经验回流
一次修复沉淀成规则、测试和下一次分诊 Context。

KEY TAKEAWAY

共同入口和结构化路由减少等待；公共任务和证据让过程可观察、经验可回流。

执行建议

为高频问题建立统一入口、分诊角色、承接频道和闭环回告。

CONTROL PLANE

Agent 规模扩大后，需要一套组织控制面。

频道 / Channel

共同上下文与协作边界

谁参与、看见什么、围绕哪个业务主题工作。

任务 / Task

责任、状态与依赖

谁承接、做到哪一步、被什么条件阻塞。

线程 / Thread

过程、判断与异常

把执行讨论留在任务现场，避免信息散落。

交付 / Artifact

结果、证据与版本

让 Review 面向具体产物，保留来源和修改历史。

KEY TAKEAWAY	频道、任务、Thread 和 Artifact 共同承载 Context、状态、过程与结果。
执行建议	要求每项跨人机工作都能回答：谁负责、做到哪、卡在哪里、交付是什么。

先分工作，再分权利

用可逆性 × 判断复杂度决定执行方式。

	判断复杂度低	判断复杂度高
容易回退	**Agent 自动执行** 整理、格式化、例行检查、信息同步；保留日志与抽检。	**Agent 先做，人抽检** 研究、初稿、分类、方案比较；人关注例外与偏差。
难以回退	**Agent 准备，人批准** 批量发布、客户触达、预算调整；执行前设置闸口。	**人主导，Agent 辅助** 重大承诺、法律风险、品牌危机、关键人事与资本决策。

KEY TAKEAWAY

可逆性和判断复杂度，比任务看起来是否简单更能决定人机分工。

执行建议

高风险动作设置人工批准；低风险高频动作允许 Agent 自动执行并保留日志。

DELEGATION NEEDS RIGHTS

任务委托必须同步配置四种权利。

信息权

能看什么

数据、客户资料、历史记录与知识范围。

行动权

能做什么

调用工具、修改内容、发送消息与触发系统。

资源权

能花什么

预算、算力、额度、外部供应商与人力。

升级权

何时停下来

遇到冲突、低置信度和高风险时找谁决策。

KEY TAKEAWAY	没有信息权、行动权、资源权和升级权，任务委托就无法形成真实责任。
执行建议	按最小必要原则授权，明确低置信度、冲突和高风险场景的升级对象。

ROLE REDESIGN

未来公司里，纯执行岗位会持续减少。

传统执行者

接任务，亲自完成动作

价值集中在熟练度、速度、信息记忆和个人经验。

→

Agent 时代执行者

设计执行系统并对结果负责

价值集中在拆解、编排、补充 Context、Review 和异常处理。

KEY TAKEAWAY	人的价值从亲自完成动作，转向设计执行系统并对结果负责。
执行建议	人才标准加入任务拆解、Agent 编排、Context 建设、Review 和异常处理。

MANAGEMENT BECOMES A GENERAL SKILL

完成一件事之前，先设计谁与谁怎样协作。

01

建几个 Agent

按能力差异和并行价值分角色，避免一个万能 Agent 承担所有判断。

02

拉几个工作现场

频道承载长期 Context，任务承载结果，Thread 承载执行过程。

03

怎样分工

明确输入、输出、完成标准、依赖与可调用资源。

04

信息怎样传导

结构化交接，减少摘要失真、重复说明与上下文丢失。

05

怎样 Review

按风险设置抽检、双人复核、自动测试和人工批准。

06

怎样管过程

看状态、阻塞、例外和证据，减少追问式管理。

KEY TAKEAWAY 组织管理会成为通用技能，资历不再决定谁能调度复杂执行能力。

执行建议 训练员工先设计 Agent、工作现场、分工、信息流和 Review，再开始执行。

MANAGER'S NEW WORK

管理者要重做五项基础工作。

定义结果	把模糊要求转成完成条件、质量、时限、成本和不可触碰的边界。
设计分工	决定哪些工作由人、Agent 或组合承担，安排交接与并行关系。
建设 Context	把流程、材料、案例、标准和判断依据变成 Agent 可使用的组织资产。
配置权利	设置数据、行动、资源和升级权限，让委托与责任保持一致。
管理例外	把精力放在冲突、风险、低置信度和跨部门权衡，而非逐项催进度。

KEY TAKEAWAY	管理者的工作重心会从催进度转向结果、分工、Context、权利和例外。
执行建议	把管理时间从逐项跟进移到目标澄清、授权设计和高风险决策。

PERFORMANCE SYSTEM

绩效要衡量结果、杠杆与组织沉淀。

业务结果

交付了什么

结果质量、客户价值、收入影响、时效与成本。

组织杠杆

组织了多少能力

并行度、自动化覆盖、Review 效率与异常处理能力。

能力沉淀

留下了什么

可复用 Context、流程、测试、模板、知识和改进规则。

KEY TAKEAWAY	只看产出数量会奖励忙碌和隐藏自动化，无法鼓励组织能力沉淀。
执行建议	同时衡量业务结果、组织杠杆和可复用 Context，并共享效率红利。

CONTEXT IS INFRASTRUCTURE

组织 Context 决定 Agent 能否进入真实业务。

流程

工作如何流动

触发条件、步骤、角色、依赖和升级路径。

物料

基于什么工作

数据、模板、案例、客户资料和历史记录。

交付物

产出长什么样

格式、颗粒度、版本、证据和接收方式。

评判标准

如何判断好坏

质量、品牌、合规、绩效和业务取舍标准。

KEY TAKEAWAY	流程、物料、交付物和评判标准，是 Agent 进入真实业务的基础设施。
执行建议	从高频稳定闭环开始，每次失败都补一条 Context、规则或测试。

MATURITY PATH

Agent 对组织 Context 的使用，从理解走向行动。

知悉

知道组织术语、角色、业务与边界。

信息提取

从资料中找到事实、案例与历史判断。

主动建议

结合当前任务提出下一步和风险提示。

方案制定

拆任务、调资源、形成计划和备选方案。

自动化执行

在权限边界内持续行动、记录状态并升级异常。

KEY TAKEAWAY	Agent 每提高一级自治程度，都要求更完整的 Context、评估、权限和恢复机制。
执行建议	按知悉、提取、建议、方案、执行逐级放权，不跨级追求全自动。

HUMAN ROLE SHIFT

人的工作重心沿着执行 → 组织 → 决策迁移。

过去

亲自执行

搜索、整理、制作、搬运、跟进；时间消耗在动作本身。

→

现在

组织执行

配置 Agent、频道、任务、信息流和 Review；管理并行工作。

→

未来

决策岗与闸口

定义目标、补 Context、处理例外，对高风险动作和最终结果负责。

KEY TAKEAWAY	人类的稀缺性集中在价值判断、关系、责任和复杂语境，Agent 提供速度与并行。
执行建议	把人安排在目标、例外和高风险闸口，让 Agent 承担高密度执行。

10 分钟真实 DEMO

看 #syfo-doctor 如何把一次报障转成可追踪研发闭环。

链路

入口 → 分诊 → 路由 → 任务

Han Solo 补信息、判断归属，把问题连同 Context 交给正确研发频道和负责人。

执行

Thread → 修改 → 测试 → Review

过程公开、依赖显式、风险升级，生产发布由有权限的人批准。

交付

commit → 测试 → 部署证据

交付物和验证证据挂回任务，状态可以独立验收。

学习

回告 → 规则 → 记忆

结论回到报障者，一次修正成为下一次任务的组织 Context。

KEY TAKEAWAY	完整闭环要同时看见责任、信息、状态、Review、批准和经验回流。
执行建议	Demo 后用六个问题审视和合谷的一条真实流程，找出最先缺失的控制点。

15 分钟高管深聊

用四个问题定位和合谷的组织起点。

01 · 哪类门店—区域—总部工作，最受等待和信息重述拖累?

02 · 如果执行能力增加十倍，运营、营销、供应链或人才管理哪个瓶颈会先暴露?

03 · 涉及食品安全、价格、客户承诺和资金的哪些决定必须由人保留?

04 · 老团队最可能抵触哪种职责、权利或评价方式变化?

KEY TAKEAWAY	组织转型应从一个高频、稳定、可衡量的闭环启动，而非从岗位裁撤启动。
执行建议	现场选定一个候选闭环，确定 Owner、当前痛点、成功指标与下一步验证。

和合谷 | Syfo

个人 AI 能力进入
共同的工作系统，
才能形成组织能力。

把执行组织起来，把过程留在现场，把经验变成 Context，把决策与责任放在清晰的闸口。

- 下一步：选择一个稳定业务闭环，验证新的组织方式

现场问题
逐题回答

六类问题：场景选择、生产化准入、Workflow、人机介入、责任治理、人才与组织能力。

每题包含：核心判断 · 判断标准 · 执行建议

01 · 场景选择与 Agent 化判断

QUESTION 01

判断一个业务是否值得 Agent 化，通常看哪些指标?

核心判断 用“价值潜力－实施与风险成本”评估。价值看频率、人工时长、流程等待和规模；可行性看输入是否数字化、结果能否验证、错误能否发现；成本看集成、返工、风险暴露与维护。

执行建议 优先选择高频、高耗时、结果可验证、错误可回退的场景。先建立当前人工基线，再比较 Agent 后的周期、质量、成本与异常率。

QUESTION 02

变化快或早期业务中，如何平衡 AI Native 投入与业务验证进度?

核心判断 早期业务需要保留探索速度。此时应稳定记录、评估和接口，不急于固化完整流程；业务假设尚未验证时，重平台投入容易把错误流程自动化。

执行建议 采用人主导、Agent 辅助的轻量闭环，完整记录真实案例。连续出现相同输入、判断和交付标准后，再把重复部分产品化。

01 · 场景选择与 Agent 化判断

QUESTION 01

自动化和 AI Native 的边界在哪里?

核心判断 规则稳定、输入结构化、异常少的工作用传统自动化更便宜可靠；输入非结构化、需要上下文判断、要动态选择工具或处理例外时，Agent 更合适。成熟方案通常是确定性流程骨架加 Agent 判断节点。

执行建议 先用规则覆盖确定部分，只把模糊分类、信息理解、方案生成和动态路由交给 Agent；避免用大模型替代一个简单判断表。

QUESTION 02

AI 员工适合独立承担什么任务？哪些必须保留人工判断?

核心判断 独立执行适合低风险、可逆、可观测、完成标准明确的任务。外部承诺、资金权限、品牌危机、法律合规、关键人事和不可逆操作必须保留人工决策。

执行建议 按影响范围、可逆性、判断复杂度和置信度设置权限；高风险动作由 Agent 准备材料，人类批准后执行。

02 · AI Native 成熟度与生产化准入

QUESTION 01

Agentic AI 项目从 POC 到生产最常见的断层在哪里?

核心判断 断层通常不在模型效果，而在缺少真实评估集、数据与权限治理、异常 Owner、监控回滚和成本边界。POC 展示平均能力，生产系统必须管理长尾失败。

执行建议 上线前明确业务基线、验收指标、错误预算、灰度范围、人工接管、日志审计和责任 Owner；这些条件缺一项，就仍是试验。

QUESTION 02

从辅助分析到自动决策，可分哪些阶段？每阶段如何放权?

核心判断 可分为只读分析、建议、人批准后执行、边界内自动执行四级。放权依据是风险、可逆性、置信度校准、异常可发现性和历史稳定度。

执行建议 每一级先运行影子模式，对比人与 Agent 结果；达到错误预算并具备监控和回退后，才扩大动作权与覆盖范围。

02 · AI Native 成熟度与生产化准入

QUESTION 01

企业的数据、权限、指标口径和治理达到什么状态，才足以支撑生产？

核心判断 无需等到所有数据完美，但目标流程必须有可信数据源、稳定主键、统一指标定义、最小必要权限、访问日志和明确的数据 Owner。Agent 无法弥补来源冲突与责任空缺。

执行建议 先为目标闭环建立小型数据契约：来源、字段、口径、刷新频率、权限、质量阈值和异常联系人，再接入 Agent。

QUESTION 02

离线、跨系统流程需要哪些前期准备？

核心判断 先看关键事件能否被数字化、系统之间是否有统一对象 ID 和状态、动作是否能通过 API 或受控 RPA 执行。线下判断与口头交接必须先外化。

执行建议 画出现状流程和系统地图，标记数据源、人工判断、权限、异常与交接；优先打通最短闭环，不要一次性集成所有系统。

03 · Workflow、任务拆解与跨系统协作

QUESTION 01

业务模式、流程和岗位分工都不清晰时，如何设计 Agent Workflow?

核心判断 先做流程发现，不从画理想流程开始。选取真实案例，追踪触发、信息、判断、动作、等待和返工，找到事实上的工作流；初期由人担任总协调者。

执行建议 连续复盘 10–20 个真实任务，形成事件与决策地图。先固定输入、输出和升级点，再逐步拆出 Agent 节点。

QUESTION 02

引入多个 Agent 后，如何判断任务拆解方案是好的?

核心判断 好的拆解让每个角色输入清楚、输出可验、依赖最少、可并行、失败可隔离、上下文不过载。Agent 数量越多不代表设计越先进。

执行建议 用五项检查拆解：接口清晰、结果可验、并行有价值、失败不扩散、交接信息最小充分。达不到就合并角色。

03 · Workflow、任务拆解与跨系统协作

QUESTION 01

从人拼接多系统信息到 Agent 协作，核心缺口是什么?

核心判断 Prompt、知识组织和工具接口都重要，但常见瓶颈先后是：对象与状态不统一、工具动作不可靠、Context 无结构，最后才是 Prompt。没有数据与工具契约，模型能力无法稳定落地。

执行建议 先定义对象 ID、状态机、字段口径和工具输入输出，再建设检索 Context 与评估集；Prompt 放在这套基础设施之上迭代。

QUESTION 02

自主性高的 Agent 能进入核心流程吗?

核心判断 可以，但自主性应被限定在清晰边界内。生产系统往往收敛为确定性 Workflow 骨架、Agent 判断节点和人工风险闸口的组合。

执行建议 为每个 Agent 标明目标、工具白名单、预算、超时、停止条件和升级对象；核心流程先小范围灰度，再扩大自治范围。

04 · 人机判断、验收与介入机制

QUESTION 01

审美、人物理解和内容创作等主观任务，如何定义完成标准？

核心判断 主观不等于无法评估。可以用评价维度、参考样本、成对比较、硬性否决项和接受区间，把隐性偏好转成可讨论的评审协议。

执行建议 建立小型黄金样本集，定义 4–6 个评分维度和必须避免的错误；先让多人校准评分，再评估 Agent。

QUESTION 02

人的判断放得太早或太晚，如何动态决定介入节点？

核心判断 比较预期损失与审核成本：错误概率 × 影响 × 返工代价高于审核成本时，人应提前介入。置信度必须经过校准，不能直接相信模型自报。

执行建议 在高分支、高承诺和不可逆动作前设检查点；低风险步骤采用抽检，持续用真实错误更新阈值。

04 · 人机判断、验收与介入机制

QUESTION 01

准确率很高但每单仍需人工确认，怎样才算真正跑通?

核心判断 如果人工只是重复确认，系统没有释放产能。成熟机制应按风险与置信度分流：高置信低风险自动通过，中间区抽检或快速确认，高风险全量审核。

执行建议 记录每次人工修改的原因，按漏检率、审核时长、自动通过率和下游缺陷共同调阈值，而非只看准确率。

QUESTION 02

人的选择如何沉淀为 Agent 的长期能力?

核心判断 把选择结果、理由、上下文和最终业务反馈结构化，形成版本化样本、规则与评估集；原始聊天记录本身不会自动成为可靠能力。

执行建议 建立反馈、标注、评估、更新、回归测试闭环，分别更新 Context、Prompt、Policy、工具或模型，并比较更新前后表现。

05 · 责任、风险与容错治理

QUESTION 01

AI 或 Agent 上岗后犯错，责任应该如何界定?

核心判断 责任跟随决策权和控制权。业务 Owner 对结果负责，系统 Owner 对控制机制负责，操作人按制度执行，高风险批准人承担批准责任；模型或供应商不能替代企业责任主体。

执行建议 为每条 Workflow 建立 RACI，明确结果 Owner、系统 Owner、执行者、审核者和升级负责人，并与权限配置保持一致。

QUESTION 02

不同任务的犯错代价差别很大，容错边界如何分级?

核心判断 按影响范围、可逆性、客户暴露、合规风险和恢复时间分级。零容错事项要求预防性闸口；可恢复事项可以设置错误预算和监控。

执行建议 建立 L0 禁止自动执行、L1 人工批准、L2 自动执行加抽检、L3 自动执行加监控四级，并按事故数据调整。

05 · 责任、风险与容错治理

QUESTION 01

问题被下游发现时，责任在 Workflow、审核环节还是使用者?

核心判断 不应只追最后触点。应判断哪个控制点本可合理发现或阻止问题：设计缺陷、数据缺陷、执行错误、审核失效、权限配置或违规使用分别归责。

执行建议 统一事故分类和根因分析，区分触发点、未拦截点和影响扩大点；整改系统控制，避免把系统问题转化为个人背锅。

QUESTION 02

Agent 出错时，系统如何支持定位、纠正和人工接管?

核心判断 需要随时停止、保留现场、重放链路、隔离版本、切换人工和验证修复。没有可观察性与接管机制，准确率再高也不具备生产资格。

执行建议 配置 kill switch、人工接管队列、输入输出与版本追踪、回归测试；修复后再调整权限和覆盖范围。

05 · 责任、风险与容错治理

QUESTION 01

长链路任务如何设计日志、状态、审计、回滚和介入点?

核心判断 每个任务需要唯一关联 ID，每一步记录输入、输出、模型、Prompt、工具版本、状态、耗时、费用和操作者；关键动作要幂等，无法回滚时准备补偿动作。

执行建议 采用任务级状态机和事件日志，在外部承诺、资金、发布与高风险分支前设置检查点；超时、重复失败和低置信度自动升级人工。

06 · 人才、组织与可复用能力

QUESTION 01

如何识别一个人只是会用 AI，还是具备 AI Native 工作方式?

核心判断 会用工具关注一次输出；AI Native 工作方式会外化 Context、拆解任务、配置角色、设计验收、管理例外，并留下可复用资产和完整证据。

执行建议 用真实工作样本评估：让候选人组织一项任务，说明分工、风险、验证和沉淀；不要只考 Prompt 技巧。

QUESTION 02

年轻员工跳过底层肌肉记忆时，如何重构面试与硬性准入?

核心判断 允许使用 AI 后，更应测试问题定义、事实核验、口头答辩和对错误的敏感度。关键岗位仍需保留无工具基础测试，确认候选人理解底层原理。

执行建议 采用有 AI 工作样本、现场答辩、关键基础无 AI 测试三段式；硬门槛按岗位风险设置，不做统一禁用。

06 · 人才、组织与可复用能力

QUESTION 01

如何重设计年轻员工成长路径，避免丢失底层能力?

核心判断 成长路径应同时积累专业判断与组织杠杆。先亲自经历典型案例和失败，再学习自动化与编排；不能只会调用，也不能长期停留在手工执行。

执行建议 采用手工理解、Agent 辅助、组织 Agent、负责闸口的阶梯，配合轮岗、复盘、师徒 Review 和周期性基础能力校验。

QUESTION 02

业务理解 AI、技术理解业务都困难，如何建立共建机制?

核心判断 单向需求交付会造成翻译损失。需要小型跨职能 POD：业务 Owner、领域专家、AI 或产品工程师、平台与风险角色共同对同一业务指标负责。

执行建议 用真实任务共创，每周看失败案例和指标；业务负责结果与规则，技术负责系统与控制，双方共同维护评估集。

06 · 人才、组织与可复用能力

QUESTION 01

如何把评估、知识治理和 Workflow 模板沉淀为可复用能力?

核心判断 复用对象不只是 Prompt，还包括评估集、数据契约、权限策略、连接器、状态机、审核规则和事故模式。需要版本、Owner、适用范围和使用反馈。

执行建议 建立能力注册表与平台负责人，项目交付时强制提交可复用资产；用 80% 通用组件加 20% 场景配置，避免每次从零定制。