

# Agentic RAG Delivery OS

## 企业级多智能体 AI 工程交付系统

Engineering Constitution v1.1 / Final Frozen Edition (最终冻结版)

本文档定义系统的最终工程形态、产品化交付协议、治理主权、职责边界、失败语义与事故处置机制。

任何代码实现、产品形态或对外交付，必须与本文档保持一致；任何偏离，必须先修订本文档并完成治理流程。

---

## 0. 系统边界声明 (System Boundary & Non-Goals)

### 0.1 本系统解决什么

本系统用于将自然语言需求工程化、产品化交付为：

- 可上线 (Deployable)
- 可回滚 (Rollbackable)
- 可审计 (Auditable)

并覆盖以下完整交付闭环：

需求澄清 → 规格采集 → 数据接入 → 解析结构化 → 切分索引 → 检索配置 → 证据校验 → 评测验收  
→ 上线 → 变更 → 回滚

---

### 0.2 本系统不解决什么 (Non-Goals)

- ❌ 不追求端到端黑盒式“自动写代码工程师”
- ❌ 不承担模型训练，仅负责推理、路由、评测与交付
- ❌ 不提供领域知识本身，仅负责工程化与可信交付
- ❌ 不绕过人工责任主体（关键节点必须支持制度化人工接管）

---

## 1. 项目定位 (One-Sentence Positioning)

一套以制度化多智能体编排 + 产品化交付入口为核心的 AI 工程交付系统，用于将“我想做一个 RAG”的模糊需求，稳定交付为可运行、可回滚、可审计的企业级系统，而非一次性 Demo。

---

## 2. 系统总体架构（End-to-End, No Skipping）



核心原则：  
多智能体不是“并发聊天”，而是按制度、按顺序、按失败条件执行的工程交付流水线。

## 3. 核心 Agent 角色与职责（交付导向，最终版）

Agent	唯一职责
Product	面向非技术用户的需求澄清、Spec UX、交付目标冻结
Orchestrator	执行顺序、状态迁移、失败治理与人工接管
Data	数据接入、解析、结构化与数据治理
Execution	工程执行器：生成配置、构建索引、运行评测、产出交付物
Evaluation	质量评测、上线门槛、失败归因
Cost	成本监控、预算约束、路径剪枝

能力映射说明：- Retrieval / Audit 为 Execution 阶段内子能力，不作为对外 Agent - Cost 为全局约束信号，独立参与调度

## 4. Delivery Spec 产品化交付入口协议 (Delivery Entry Protocol)

### 4.1 向导式规格采集 (Wizard) ——强制交互约束

Delivery Spec 采集 **必须** 通过向导完成，而非自由表单：

- Wizard **至少包含  $N \geq 4$  个步骤**：
- 场景与目标确认
- 数据源与范围确认
- 质量 / 成本 / 风险偏好
- 上线方式与回滚策略

任何跳过步骤的行为 → 禁止 Spec 冻结。

---

### 4.2 默认模板 (Scenario Templates)

系统必须内置以下模板，并作为 Wizard 的默认起点： - FAQ / 客服知识库 - 条款 / 政策解读 - 售后 / 工单支持 - 投研 / 内部分析 - 企业 Wiki / 内训资料

模板字段不可删除，仅允许显式修改。

---

### 4.3 Assisted Spec (“我不知道怎么选”模式)

当用户无法明确选择时： - 系统可自动代填推荐值 - 每一项代填**必须给出可解释理由** - 所有字段在冻结前允许人工修改

---

### 4.4 Spec 冻结前强制 Review 页面

在 Delivery Spec 冻结前，系统 **必须展示 Review 页面**，明确列出： - 关键决策项 - 风险与成本提示 - 上线与回滚摘要

未经过 Review 页面确认 → 禁止 Spec 冻结。

---

### 4.5 Delivery Spec 最小字段集 (MVP)

任何可冻结的 Spec **至少包含**： - 目标用户 (Audience) - 回答风格 (Strict / Balanced / Exploratory) - 必须引用规则 - 数据源类型 - 上线渠道 (API / Web / Internal) - SLO 预算 (Latency / Cost / Quality)

字段缺失 → Spec 不可冻结。

---

## 4.6 变更 Diff 的强制展示规则

在任何二次上线或变更前： - Diff 页面 **必须默认展示前三大变化项** (Spec / Data / Index) - 用户需显式确认变更影响

未确认 Diff → 禁止执行 Execution Agent。

---

## 4.2 默认模板 (Scenario Templates)

系统必须内置以下场景模板： - FAQ / 客服知识库 - 条款 / 政策解读 - 售后 / 工单支持 - 投研 / 内部分析 - 企业 Wiki / 内训资料

模板提供默认字段、推荐策略与风险提示。

---

## 4.3 Assisted Spec (“我不知道怎么选”模式)

当用户无法明确配置时： - 系统允许自动代填推荐值 - 每个自动决策必须提供**可解释理由** - 所有字段在冻结前可人工修改

---

## 4.4 Delivery Spec 最小字段集 (MVP)

任何可冻结的 Spec **至少包含**： - 目标用户 (Audience) - 回答风格 (Strict / Balanced / Exploratory) - 必须引用规则 - 数据源类型 - 上线渠道 (API / Web / Internal) - SLO 预算 (Latency / Cost / Quality)

字段缺失 → Spec 不可冻结。

---

# 5. 多智能体编排与治理 (System Orchestrator)

- 显式 DAG (LangGraph / 等价)
- 禁止隐式循环、隐式重试
- 阶段失败 → 中止 / 降级 / 人工接管

## 5.1 人工接管的制度化触发条件 (Governance)

必须触发人工接管的情况包括但不限于： - 同一阶段连续失败  $\geq N$  次 - 合规 / 引用冲突 - SLO (Latency / Cost / Grounded Rate) 破线 - 灾难级事故信号 (见第 12 章)

人工接管后系统进入： - **Paused / Read-only / Rollback** 之一

---

## 6. 数据接入、解析与 Data Manifest

### 6.1 Source Connector 规则

- Connector 仅负责获取数据
- 不得执行解析、Embedding 或 Index 行为

### 6.2 Data Manifest（强制）

每批数据必须生成 Manifest： - source - hash - version - license - PII level

---

## 7. Execution Agent（工程落地执行器）

Execution Agent 是系统中唯一允许产生工程级交付物的 Agent，其职责不仅是“执行任务”，而是对最低可运行工程结果负责。

### 7.1 Execution Agent 的最小可运行 Contract（Non-Negotiable）

任何一次成功执行，Execution Agent 至少必须保证产出以下三项之一组完整工件：

1. 可运行工程工件（Runnable Artifact）
2. 一个可运行的 Repo、服务包或等价工程工件
3. 明确运行入口与依赖边界
4. 一键部署入口（One-Click Deploy Entry）
5. 明确的启动方式（如 `make deploy` / `docker compose up` / 平台化部署按钮）
6. 不依赖隐式人工步骤
7. 可复现评测命令（Reproducible Evaluation Command）
8. 明确评测入口（如 `make eval` / CI Job / Pipeline Step）
9. 在相同 Spec / Data / Index 条件下结果可复现

未同时满足以上三项 → 本次执行视为失败，不得进入交付状态。

---

## 8. 非技术用户的失败恢复与可视化（Delivery UX）

### 8.1 失败原因分类（可读）

- 数据不可用
- 解析失败

- 索引失败
- 评测未达标
- 预算超限

## 8.2 一键修复建议

- 更换解析器
- 调整切分策略
- 放宽 / 收紧引用规则
- 调整预算

## 8.3 变更 Diff 可视化

展示与上一版本的 Spec / Data / Index 差异。

---

## 9. 成本治理（Cost-Aware Routing）

- Fast / Standard / Premium 分层
  - Token 配额 + 熔断
  - 超限 → 降级或阻断
- 

## 10. Platform Profiles（产品定位去风险）

- **Default Profile :**
  - 成本、回滚、可观测
  - **Enterprise Profile（可选） :**
  - 多租户隔离
  - 外部审计接口
  - 合规 workflow
- 

## 11. Constitution 修改与治理流程（Platform Sovereignty）

- Constitution 为最高治理文件
  - 修改需走 RFC → 双评审 → 审计留痕
  - 紧急修改允许临时生效，但必须事后补审
-

## 12. 灾难级事故处置流程（Post-Incident Protocol）

适用事件：- Data Leak - 严重 Hallucination

流程：1. 自动冻结相关交付 2. 强制人工接管 3. 证据保全（Spec / Data / Trace） 4. RCA 分析 5. 修复后重新评测与上线

---

### 最终声明

本系统不是“如何生成答案”，  
而是“当 AI 系统长期运行、失败、被追责时，组织如何仍然可控地交付价值”。