

面向智能体的知识工程： 从知识本体、知识构建到智能体应用

汇报人： 盛傢伟

2026年7月

CONTENTS

汇报提纲

- 01 研究背景
- 02 知识本体的发展
- 03 知识构建关键技术
- 04 知识增强的智能体应用

知识工程的基本内涵

- 知识工程是对知识进行系统化建模、获取、组织、验证、推理与应用的技术体系
- 它以本体提供语义规范，通过知识库构建形成可用资产，能够通过检索、推理与执行支撑智能应用

1 知识从哪里来？



专家经验：访谈、观察与知识梳理



文献数据：论文、标准与业务数据库



自动抽取：文本、图像及多模态信息

2 知识如何组织？



概念体系：定义领域术语与分类层级



关系规则：描述实体关联与逻辑约束



知识表示：构建本体、规则库和知识图谱

3 知识如何发挥作用？



知识检索：快速定位相关事实与证据

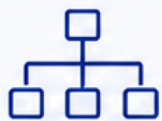


知识推理：发现隐含关系并辅助判断



智能应用：支持问答、推荐与决策

知识工程的主要环节



本体建模

定义概念、属性与关系



知识获取

采集、抽取领域知识



知识融合

对齐、消歧与去重



知识加工

校验、推理与更新



知识应用

检索、问答与决策

智能体时代的必要性

□ 大模型参数知识具有覆盖广、迁移快、交互自然等优势，但仍面临难以及时更新、准确溯源、稳定验证、安全执行的挑战。因此，智能系统需要将大模型的通用能力与工程化知识体系结合起来

模型参数中的知识



1

隐式、概率化

知识分散在参数权重中，以概率方式生成答案



2

难以直接修订

无法精准定位并更新某一条具体事实



3

来源不透明

回答与训练证据之间难以建立明确对应



4

擅长生成与泛化

能够综合语言模式，处理开放和模糊问题

适合：开放表达、归纳生成、模糊理解

工程化知识



1

显式、可查询

概念、事实、规则和关系都可以被定位



2

可持续更新

支持增量修改、版本管理、迁移与回滚



3

来源可追溯

每条知识关联证据、时间与责任主体



4

支持约束与验证

通过本体、规则和质量流程发现错误

适合：事实管理、规则校验、流程执行

大模型解决“语言与泛化”，知识工程解决“事实、结构、专业性与可信性”

本次报告内容

□ 面向智能体的知识工程，以本体定义业务世界，通过知识获取、融合与补全，将多源数据转化为可信、可持续更新的知识资产，智能体进一步利用知识，形成从认知、推理到可靠行动的闭环



CONTENTS

汇报提纲

- 01 研究背景
- 02 知识本体的发展
- 03 知识构建关键技术
- 04 知识增强的智能体应用

本体的发展

□ 本体历经漫长的发展历史，从哲学引入到计算机科学，再到大模型时代。面向大模型时代对“知识可信”的新需求，融合形式化本体、知识语义推理与大模型能力，逐渐成为新的智能范式



本体的概念

- ❑ 哲学本体论关注存在的类别及其关系，即“世界由什么构成、事物如何关联”
- ❑ 知识工程中的本体则是对领域概念体系的形式化、显式化和共享化描述

01

哲学中的本体

ONTOLOGY / METAPHYSICS

Q

词源

ontos = being (存在)
-logia = study (研究)

?

基本问题

研究“什么东西存在”，以及“它们之间有什么关系”。
世界由什么构成？如何分类？彼此如何依存？

M

《范畴篇》

亚里士多德提出十范畴，是现存西方哲学文献中最早且影响深远的系统分类之一。

亚里士多德十范畴

存在

实体

数量

性质

关系

地点

时间

姿态

状态

动作

承受

存在 · 范畴 · 关系

02

知识图谱中的本体

FORMAL ONTOLOGY

M

基本概念

对特定领域中的概念、关系、约束和公理进行明确、形式化的描述。

C

典型构成

概念 · 属性 · 关系 · 实例 · 公理与约束

A

主要作用

统一语义、组织与复用知识，并支持查询、一致性检查和逻辑推理。

E

例子 | 医疗知识

概念：患者、疾病、肺炎、病毒性肺炎
属性：患者（定义域：患者；值域：疾病）
类关系：病毒性肺炎 $\subset$ 肺炎 $\subset$ 疾病
类型断言：患者 A $\in$ 患者；病例 B $\in$ 病毒性肺炎
事实：患者 A $\rightarrow$ 患有 $\rightarrow$ 病例 B

医疗本体示例

患者 (概念)

疾病 (概念)

肺炎 (概念)

病毒性肺炎 (概念)

患者 A (实例)

病例 B (实例)

subClassOf

subClassOf

instance-of

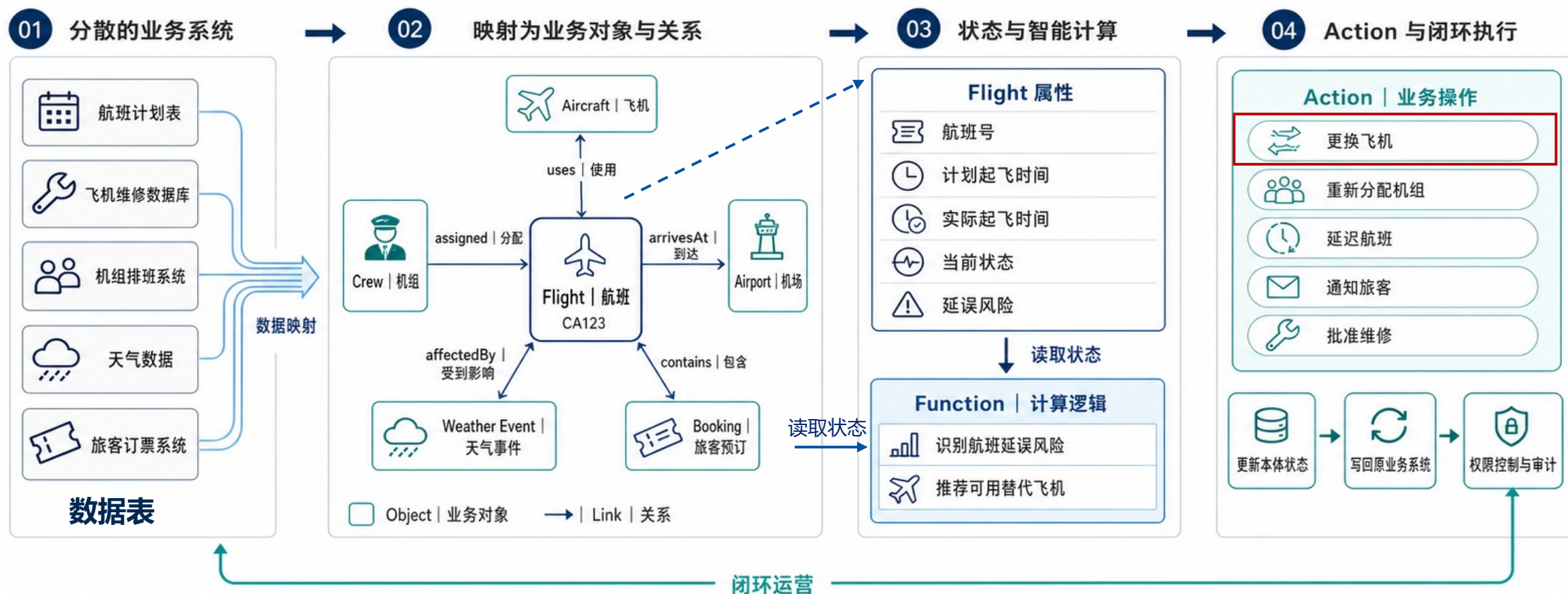
instance-of

患有

语义 · 约束 · 推理

Palantir 本体

- Palantir 本体将分散的数据映射为**业务对象、属性与关系**，并进一步**绑定函数、行动和权限**
- 人、应用与 AI 因而能够通过同一套业务语义读取状态、辅助决策并执行受控操作



Palantir 本体的组成

- Palantir 本体以**对象、属性、关系和接口**构建语义层，以 **Function 和 Action** 封装计算与受控变更
- **权限、安全策略、血缘和审计**构成治理底座，三者共同将数据转化为可理解、可计算、可执行的业务能力

01 语义层 | 描述业务世界



Object Types | 对象类型

定义企业现实世界中的业务“名词”

客户 · 订单 · 设备 · 航班

类型类似类，具体记录是对象实例



Properties | 属性

描述对象的特征与实时状态

金额 · 状态 · 位置 · 健康评分

可来自原始数据、传感器、模型或人工输入



Links | 链接

表达对象之间的业务关系

客户→订单 · 设备→工厂

沿业务关系查询，无需理解底层表连接



Interfaces | 接口

抽象不同对象类型的共同能力

Locatable · Maintainable

支持跨对象类型的统一处理

02 动态/行动层 | 计算并改变业务状态



Functions | 函数与业务逻辑

读取对象、遍历关系，执行查询、计算与推理



计算风险与指标



生成推荐与
派生属性



表达复杂对象
修改逻辑

如何计算

支撑复杂执行



Action Types | 动作类型

以业务意图为单位，正式改变对象、属性与关系



一次事务可修改
多个对象



可写回外部
业务系统



受参数校验、
权限与审计控制

可以执行什么

03 安全与治理 | 横贯所有组件的运行底座



对象与属性可见性



Action 执行权限



条件校验



审批流程



数据血缘



审计追踪

两类本体的关系

- 知识图谱本体由**概念、属性、关系、公理与约束**构成，侧重共享语义、知识表示和逻辑推理
- Palantir 本体在**语义模型**上进一步连接**数据、函数、行动与权限**，侧重支撑业务决策和执行闭环

知识图谱本体包含什么？

A 概念与关系定义



类 (Class) | 定义领域对象类型
对象属性 (Object Property) | 实体间关系 (Relation)
数据属性 (Data Property) | 实体属性 (Attribute)
注释属性 (Annotation Property) | 标签、定义、来源

B 形式语义与约束



公理 (Axiom)
类层次 · 等价与互斥 · domain / range
属性特征 · 类表达式 · 基数限制

C 实例与事实



个体 (Individual) | 知识图谱中的实体 (Entity)
断言 / RDF三元组 (Assertion / RDF Triple)
subject-predicate-object事实陈述

完整OWL本体可以包含 Individual 和 Assertion；
工程中也常将本体模式与实例数据图分开管理

概念对应关系

相近，但不完全等价

	知识图谱本体 / OWL	Palantir Ontology	对应程度
1	类 (Class)	对象类型 / 接口 (Object Type / Interface)	近似对应
2	对象属性 / 关系 (Object Property / Relation)	链接类型 (Link Type)	直接对应
3	数据属性 / 属性 (Data Property / Attribute)	属性 (Property)	直接对应
4	个体 / 实体 (Individual / Entity)	对象 (Object)	直接对应
5	断言 / RDF三元组 (Assertion / RDF Triple)	属性值 / 链接 (Property Value / Link)	近似对应
6	公理与约束 (Axiom / Constraint)	类型配置与 Logic	部分对应
7	无严格对应	动作类型 (Action Type)	Palantir扩展
8	无严格对应	Security	Palantir扩展

Palantir Ontology不是RDF/OWL标准栈的另一种实现

Palantir Ontology包含什么？

A 语义对象模型



对象类型 (Object Type)
属性 (Property)
链接类型 (Link Type)
接口 (Interface)
共享属性 (Shared Property)

B 运行对象



对象 (Object) | Object Type的具体实例
属性值 (Property Value) | 对象当前字段值
链接 (Link) | 具体Object之间的关联

C 运营能力

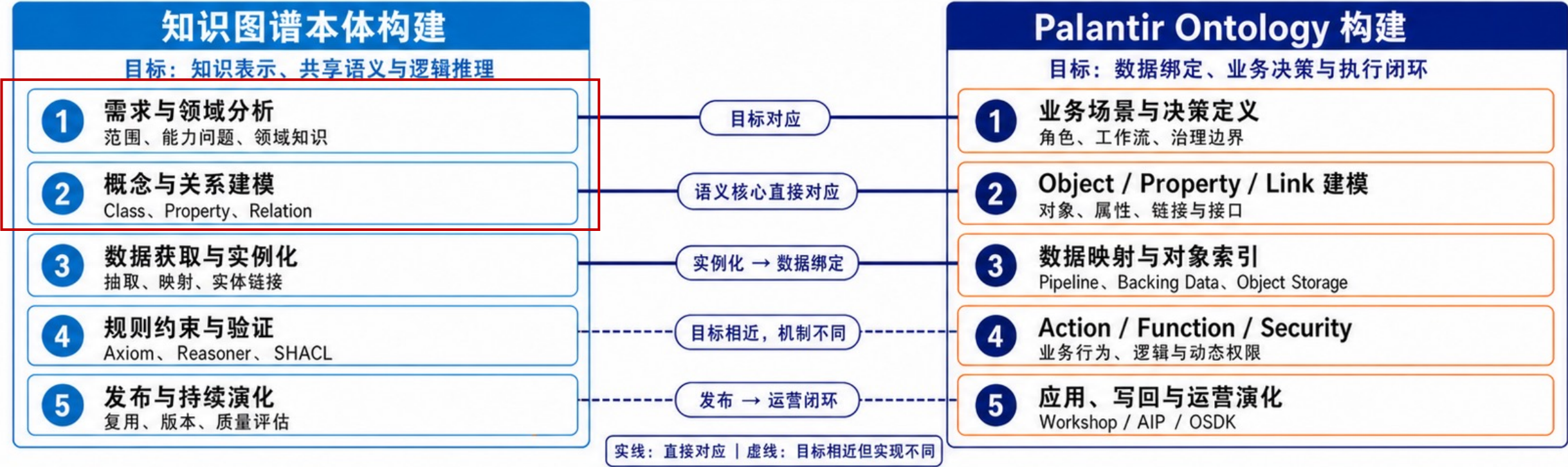


动作类型 (Action Type) | 业务输入、条件、逻辑与结果
Logic | 规则、Functions和模型调用
Security | 约束对象、逻辑和动作的访问与执行

官方Ontology resources还包括 Object Type Groups；
Logic与Security用于说明更广义的 Ontology system能力

构建过程对比

- 知识图谱本体从能力问题与知识范围出发，依次完成概念化、形式化、实例填充和逻辑验证
- Palantir 本体从业务决策与工作流出发，将数据映射为对象、属性和关系，配置接口、函数、Action 与权限



Palantir 的增量能力

真实数据绑定

Function 逻辑层

Action 行为层

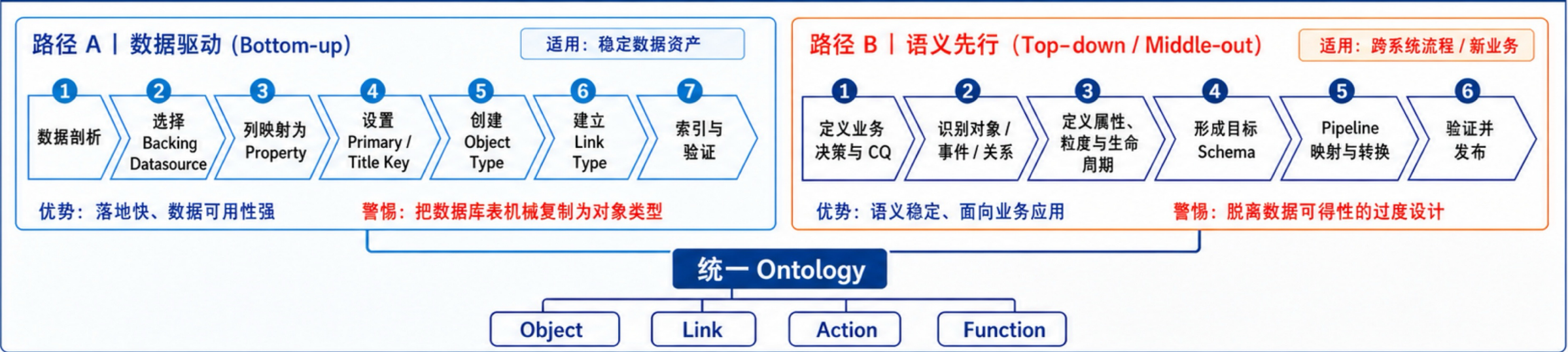
动态安全治理

应用与受控写回

Palantir Ontology = 共享语义骨架 + 数据绑定 + 逻辑与行为 + 安全治理 + 应用接口

本体构建实例

- 数据驱动路径自底向上，从数据库、文档与接口中识别实体和关系，经模式归纳形成本体
- 业务驱动路径自顶向下，从决策目标与工作流出发定义对象、关系、函数和 Action，再映射到底层数据



CONTENTS

汇报提纲

- 01 研究背景
- 02 知识本体的发展
- 03 知识构建关键技术
- 04 知识增强的智能体应用

知识获取任务

- 知识获取从文本、表格和多模态数据中抽取结构化信息，其目标结构由本体定义
- 典型任务包括：实体识别（类型）、三元组抽取（实体类型+关系类型）、事件抽取（事件类型+论元角色）

统一输入文本

2025 年 3 月 ， 甲公司 在 北京 发布 星辰大模型 。

TIME ORG LOC TRIGGER PRODUCT

1 实体识别

边界检测 + 类型分类

任务含义

定位文本中的实体边界，并判断实体类型。

输入

输入：原始文本 / Token 序列

结构化输出

[
 {"text": "2025年3月", "type": "时间"},
 {"text": "甲公司", "type": "组织"},
 {"text": "北京", "type": "地点"},
 {"text": "星辰大模型", "type": "产品"}
]

标注数据可视化

2025年3月 甲公司 北京 星辰大模型

B-TIME B-ORG B-LOC B-PRODUCT

结构抽象

Text → Span + Type
输出粒度：单个实体片段

2 三元组抽取

实体配对 + 关系分类

任务含义

识别实体之间的语义关系，形成知识图谱三元组。

输入

输入：文本 + 实体（显式或隐式）

结构化输出

(甲公司, 位于, 北京)
(甲公司, 发布, 星辰大模型)

标注数据可视化

甲公司 位于 北京

甲公司 发布 星辰大模型

结构抽象

Text → (Subject, Relation, Object)
输出粒度：实体对上的语义边

3 事件抽取

触发词识别 + 论元角色标注

任务含义

识别事件类型、触发词及不同角色的事件论元。

输入

输入：文本 + 事件 Schema

结构化输出

{
 "event_type": "产品发布",
 "trigger": "发布",
 "arguments": {
 "发布方": "甲公司",
 "产品": "星辰大模型",
 "地点": "北京",
 "时间": "2025年3月"
 }
}

标注数据可视化

发布方: 甲公司 产品: 星辰大模型

地点: 北京 时间: 2025年3月

触发词: 发布

结构抽象

Text → Event Type + Trigger + Arguments
输出粒度：多角色事件框架

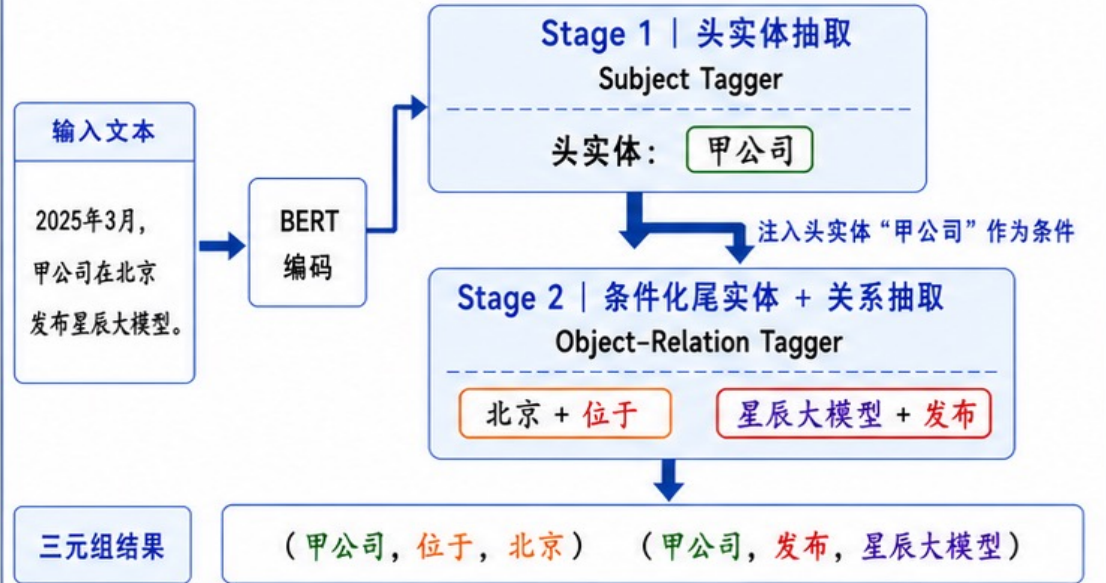
知识获取典型方法

- 任务专用模型分别对实体、关系和事件建模，统一生成模型则将不同任务转化为结构化生成
- 大模型抽取进一步增强开放场景适应能力，但仍需 Schema 约束与工具校验保证输出可控

统一案例：2025年3月，甲公司在北京发布星辰大模型。

01 传统方法：CasRel / BERT

专用三元组抽取 · 两阶段条件级联

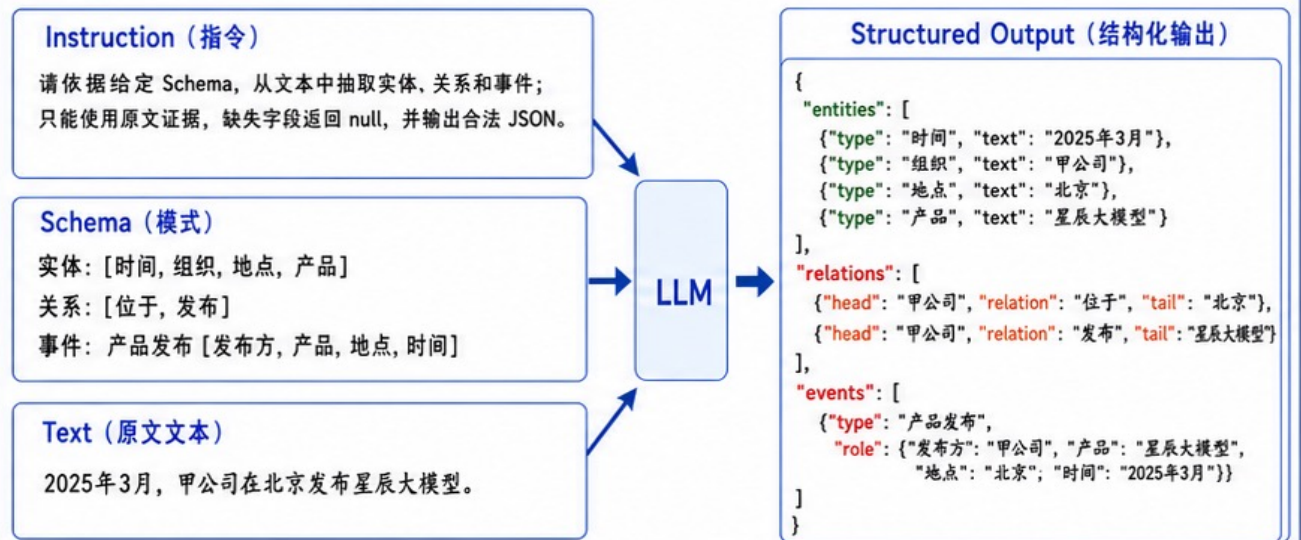


- 三元组结果
- (甲公司, 位于, 北京) (甲公司, 发布, 星辰大模型)
- 主要改进 / 提升
- ① 级联二元标注，避免枚举全部实体对
 - ② 以主体为条件识别客体和关系
 - ③ 支持共享主体或客体的重叠三元组
 - ④ 边界精确，固定关系下推理稳定

局限：关系 Schema 固定；新增关系通常需重新训练

02 大模型方法：统一信息抽取

Schema-conditioned LLM · 结构化生成



- 主要改进 / 提升
- ① 统一实体、关系、事件和槽位抽取
 - ② 支持零样本、少样本和动态字段
 - ③ 减少独立模型与流水线维护
 - ④ JSON Schema、约束解码、原文证据增强稳定性

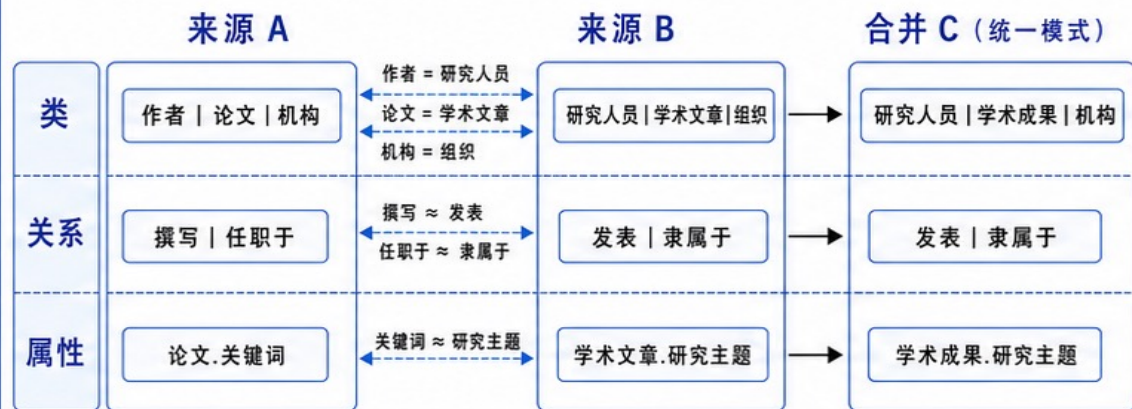
局限：存在遗漏、幻觉、边界漂移与推理成本

知识融合任务

- 知识融合解决**模式、实体和事实**等层面的不一致，将**异构、重复或冲突**的多源知识统一起来
- 典型任务包括面向概念层的本体对齐、面向实例层的实体对齐，以及指代消解等

01 本体对齐 | 概念层

对齐类、关系与属性，建立统一模式

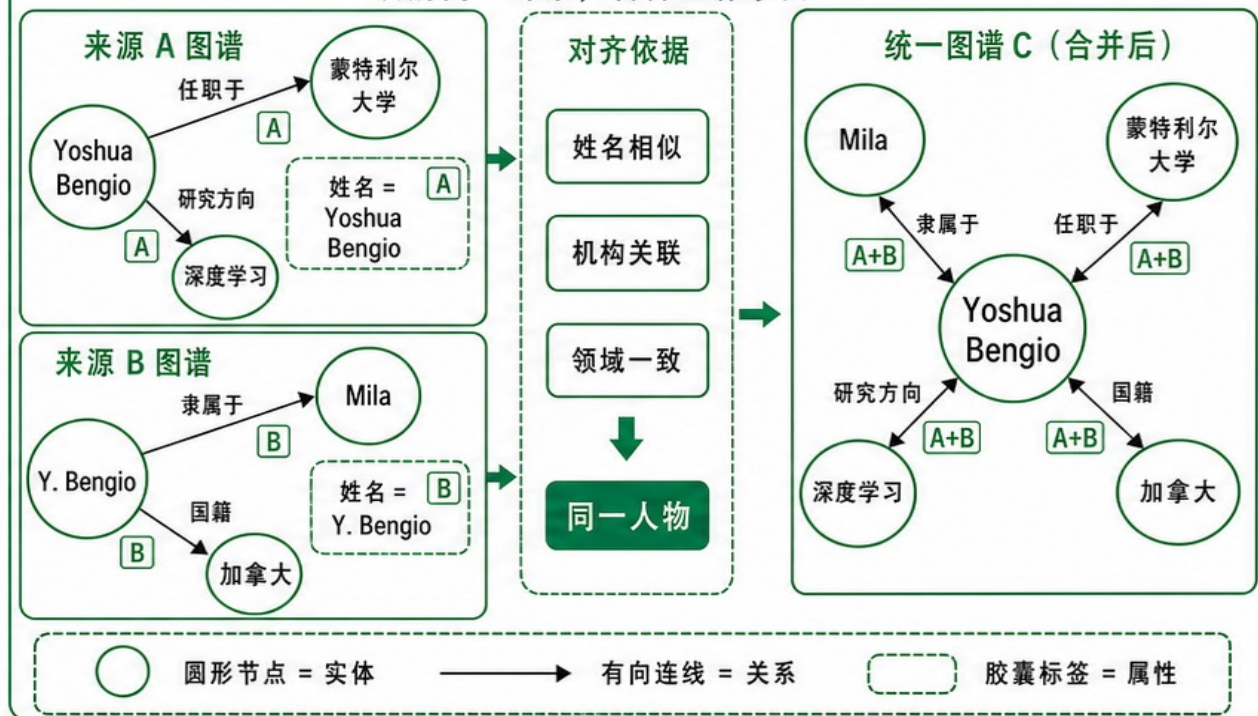


统一模式 (示例)



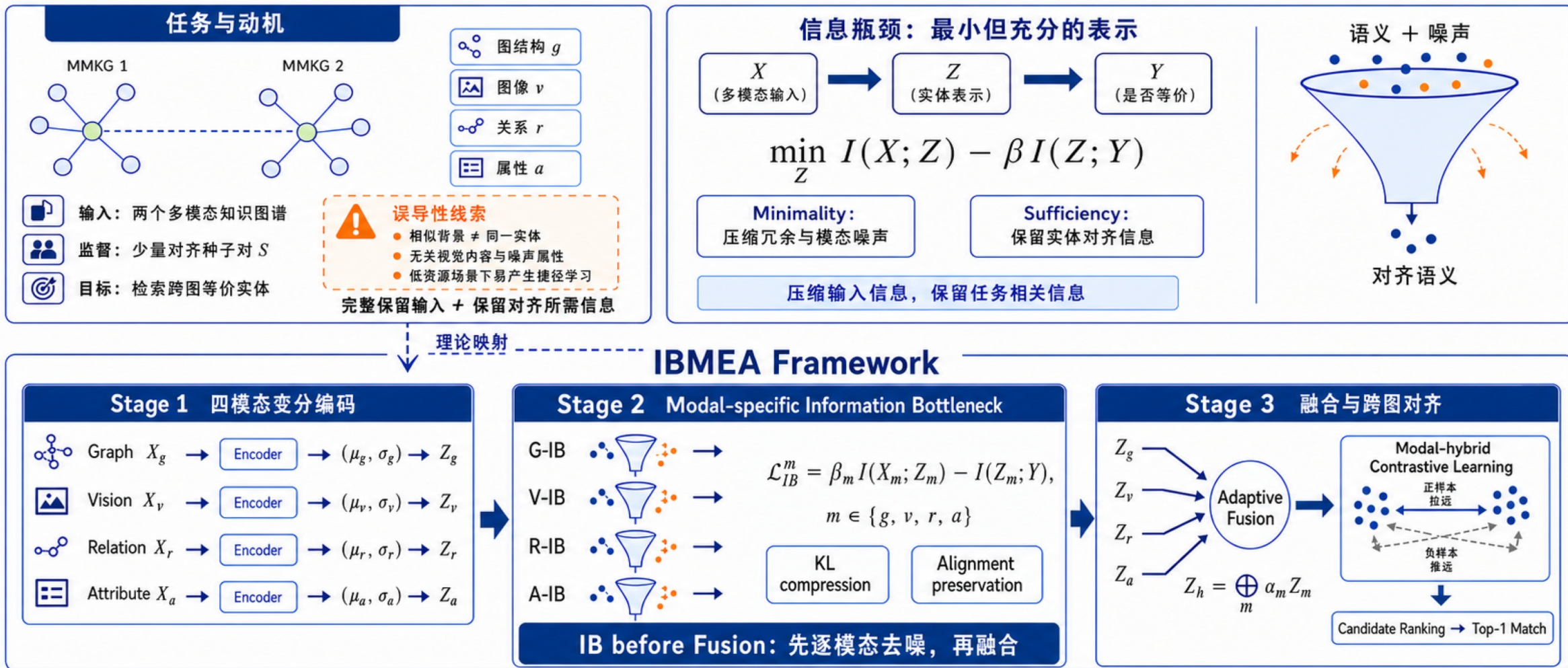
02 实体对齐 | 实例层

识别同一对象，合并互补事实



知识融合典型方法

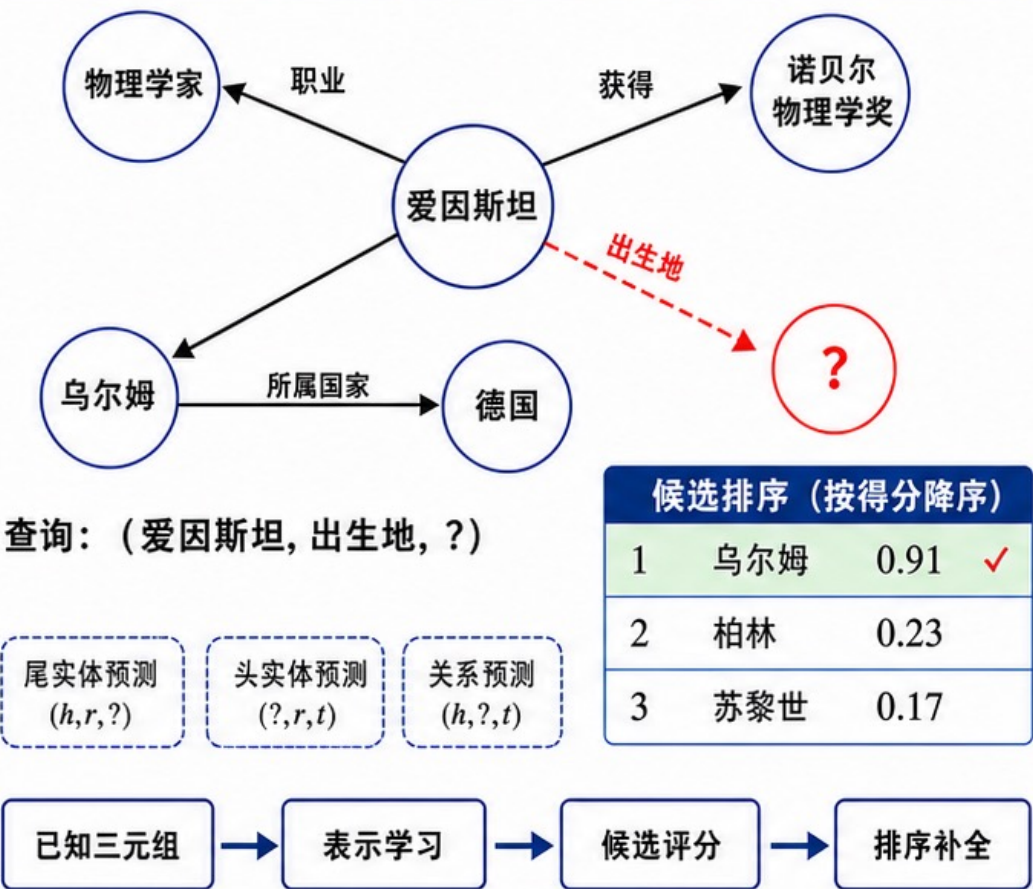
- 传统方法基于名称、属性和上下文相似度进行匹配，表示学习与迭代对齐进一步利用图结构和语义信息
- 在提取不同来源实体的特征时，可以通过约束噪声信息，提升相同实体的特征一致性，提升对齐质量



知识补全任务

- 知识图谱天然存在**缺失关系、缺失实体和属性不完整**等问题
- 知识补全**通过已有结构、语义与规则，对候选事实进行预测和排序**

任务案例 | 缺失链接预测



典型方法 | 知识图谱表示学习

1 TransE | 平移模型

关系是头实体到尾实体的平移

$$h \xrightarrow{r} t$$

$$h + r \approx t$$

$$f = -\|h + r - t\|_p$$

优势：简单、高效、易扩展

局限：复杂映射关系表达不足

2 DistMult | 双线性模型

关系对各维特征进行加权匹配

$$h \odot r \odot t =$$

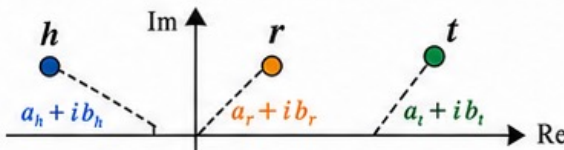
$$f = \langle h, r, t \rangle$$

优势：参数少、训练快

局限：难以建模非对称关系

3 ComplEx | 复数空间模型

复数相位区分头尾实体方向



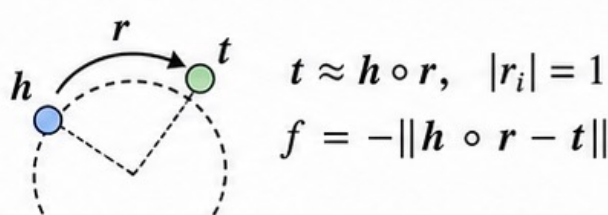
$$f = \text{Re} \langle h, r, \bar{t} \rangle$$

优势：兼顾对称与非对称关系

局限：几何解释不够直观

4 RotatE | 复平面旋转模型

关系表示复平面中的旋转



优势：支持逆关系与关系组合

局限：计算与调参成本较高

知识补全典型方法

□ 面向开放域知识图谱补全，通过智能体式“检索-反思”循环主动获取外部证据，并结合局部图谱上下文进行多轮综合推理。将验证后的候选事实规范化并写回知识图谱，从而补充新兴实体与知识截止点之后的事实

为什么需要 AgREE?



新兴实体

未进入既有 KG,
缺少邻居与实体表示



知识截止

LLM 参数知识元法
覆盖训练截止后的事实



单次检索不足

固定查询难以覆盖
一对多关系的完整答案



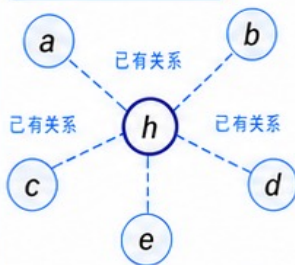
核心挑战:

动态、开放、证据不完备。

AgREE 如何完成一次开放域 KGC 查询?

1 A. KG 查询与局部上下文

输入:
待补全: $(h, r, ?)$
实体描述 $d(h)$
可选: 已有邻接三元组 $N(h)$



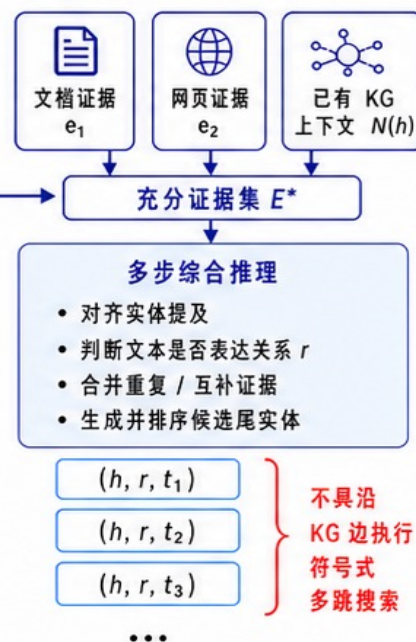
关系 r 定义需要寻找的事实类型;
已有 KG 只提供局部约束,
不保证包含答案。

2 B. Agentic 检索-反思循环

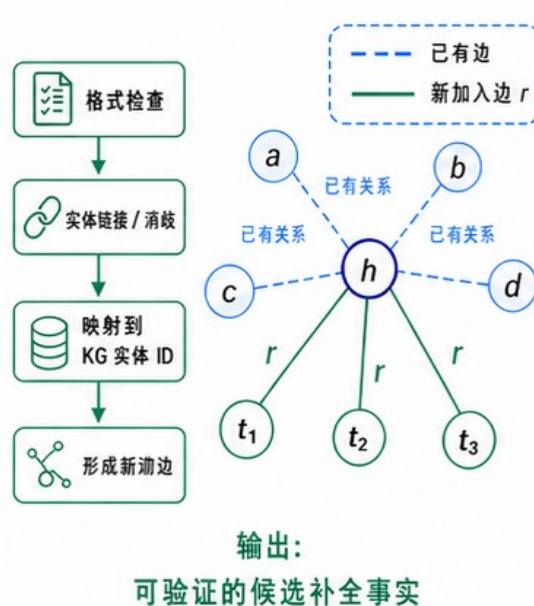


每轮观察结果进入 Agent 状态,
指导下一次行动。

3 C. 证据到候选三元组



4 D. 规范化并写回 KG



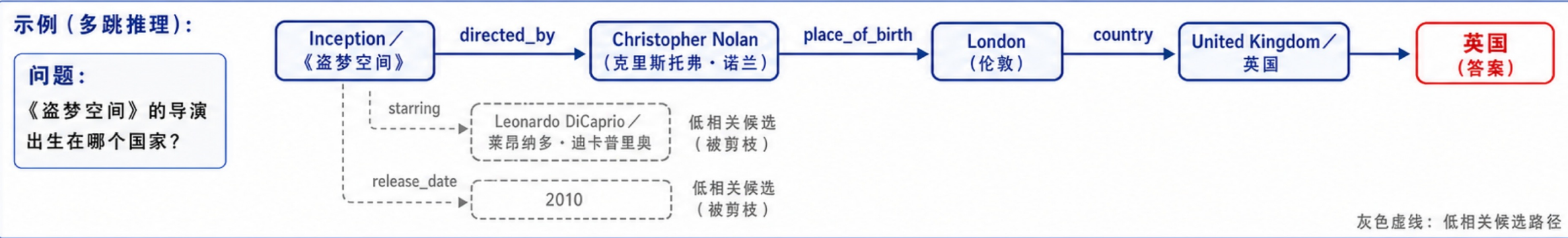
KG 提供任务约束与局部上下文; Agent 主动寻找外部证据; LLM 跨轮综合证据; 实体链接将文本答案重新落到图谱空间。

AgREE 的 multi-step reasoning = 多轮检索决策 + 跨证据综合推理, 而非传统路径推理模型的 KG 多跳遍历。

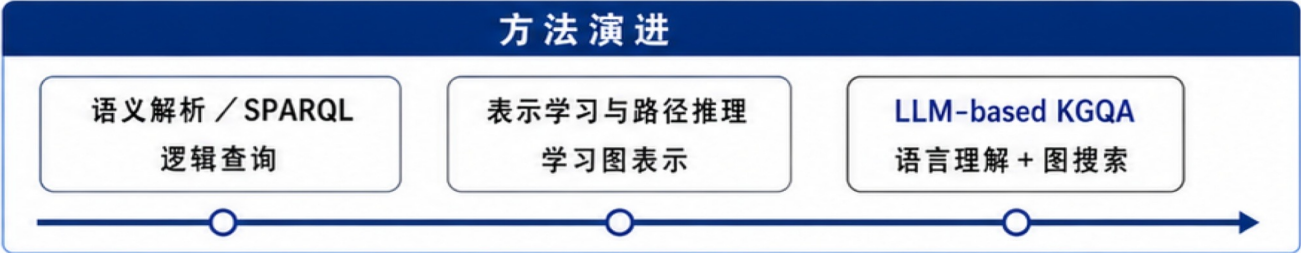
知识应用：知识问答

- 知识问答将自然语言问题依次转化为问题理解、实体链接、检索推理和答案生成
- 其主要难点包括语义歧义、多跳推理、复杂约束、知识时效性与答案可解释性

任务定义：输入自然语言问题 q 与知识图谱 $G=(E,R,T)$ ，输出答案实体或实体集合。核心目标：在知识图谱中找到 **可执行、可验证的知识路径**。



任务类型			
单跳问答 一个关系即可到达答案	多跳问答 组合多个事实链	约束型问答 类型 / 时间 / 属性约束	聚合型问答 计数 / 比较 / 排序

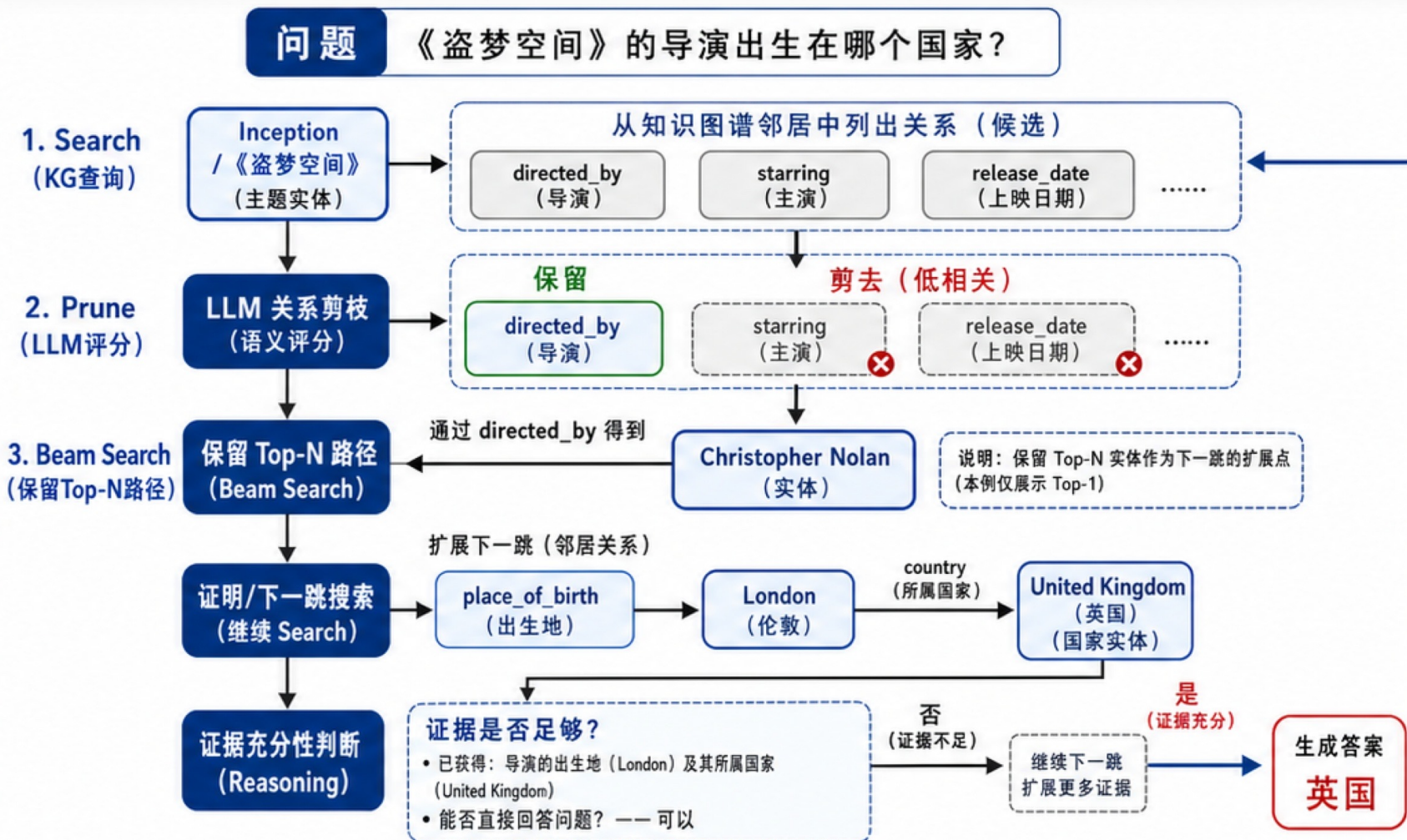


知识问答典型方法

□ 引导大模型在知识图谱上逐步搜索，通过关系筛选、路径剪枝和束搜索保留最有价值的推理方向。模型沿候选路径持续检索并判断证据是否充分，最终基于可验证的知识路径生成答案，减少事实幻觉和错误传播

为何需要 ToG

- 1 LLM-only:**
参数知识推理 →
易产生事实幻觉
- 2 传统 KGQA:**
一次生成 SPARQL →
前序错误导致整体失败
- 3 ToG:**
逐跳搜索与判断 →
形成可验证证据路径



CONTENTS

汇报提纲

- 01 研究背景
- 02 知识本体的发展
- 03 知识构建关键技术
- 04 知识增强的智能体应用

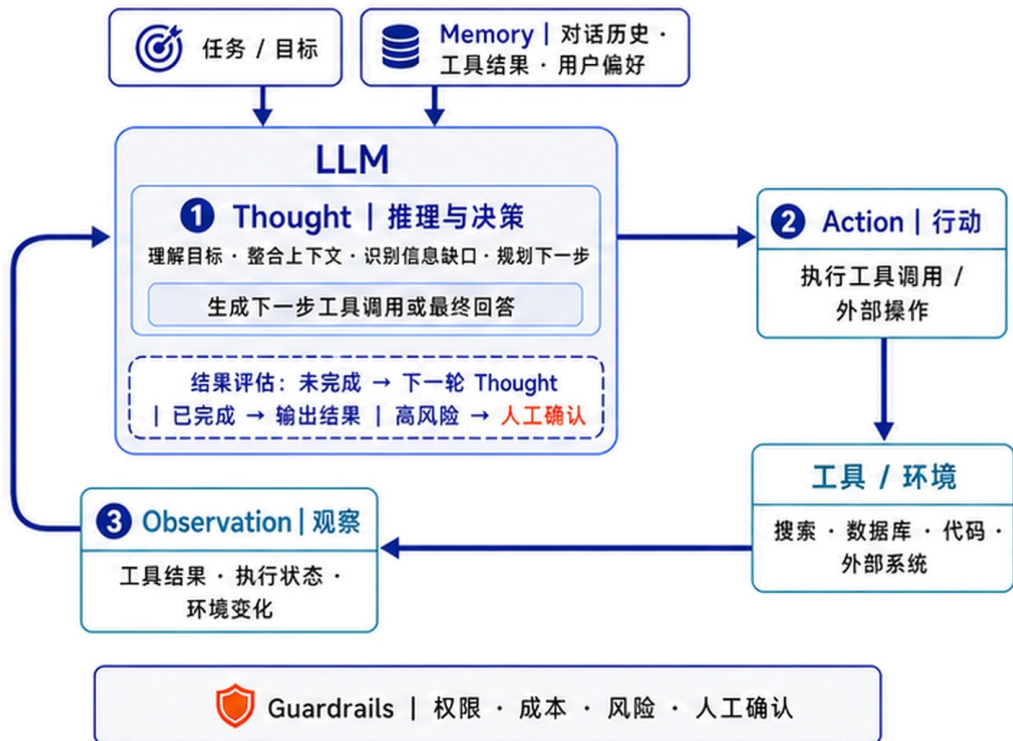
智能体的基本框架：ReAct

- ReAct 让大模型交替生成推理过程与任务行动，并根据环境返回的观察结果持续修正后续决策
- 这一闭环使模型能够调用外部工具和知识源，是规划、记忆、反思与多智能体协作等机制的基础

概念定义

智能体是由大语言模型（LLM）驱动的闭环系统：围绕目标整合上下文、形成决策、调用工具，并依据环境反馈持续调整行动。

ReAct：LLM 控制的智能体运行骨架



LLM 在内部完成 Thought 并生成 Action；工具与环境执行行动并返回 Observation，反馈驱动下一轮决策。

案例 | 旅行规划智能体的一次 ReAct 轨迹

目标：预算 8,000 元内完成五天旅行，兼顾时间、舒适度与景点偏好

第 1 轮 | 补全交通信息

Thought:	缺少价格与余票，暂时无法确定方案。
Action:	查询候选城市的航班、高铁及余票。
Observation:	直飞超预算；高铁有票，但增加约 3 小时路程。

第 2 轮 | 验证外部条件

Thought:	交通成本可行，还需验证天气与开放状态。
Action:	查询天气和景区开放信息。
Observation:	第三天强降雨，原定户外路线风险较高。

第 3 轮 | 重排并校验约束

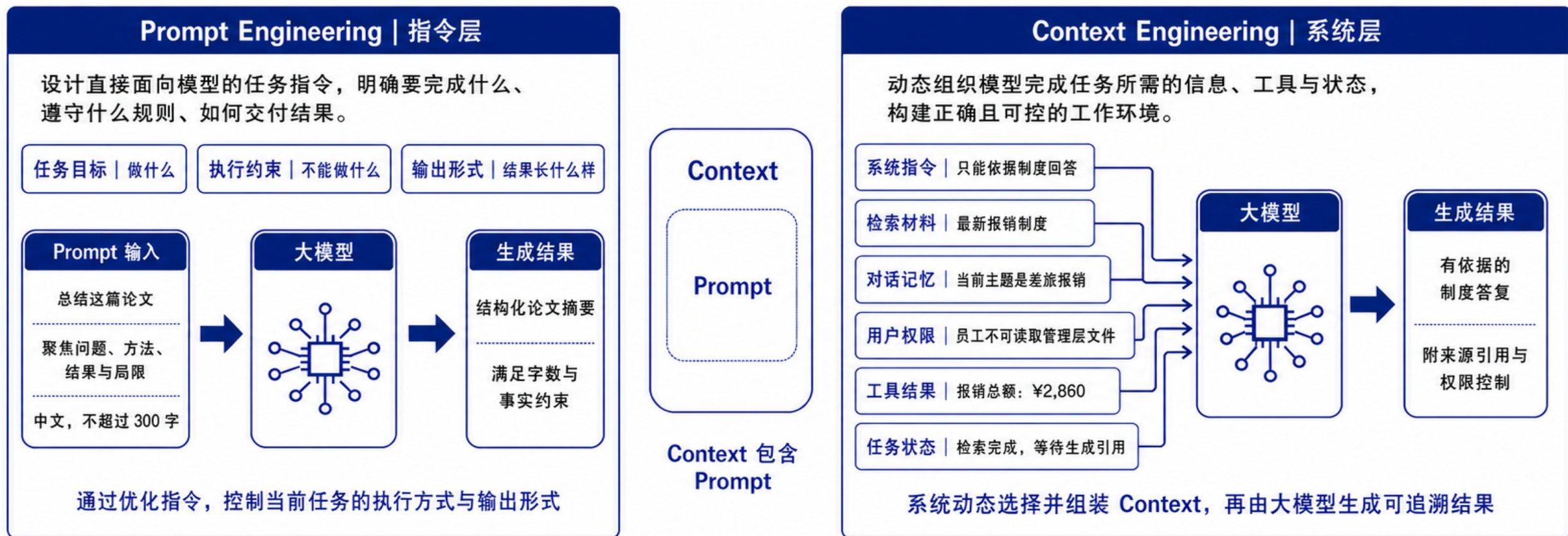
Thought:	将博物馆调整至雨天，并重新检查预算。
Action:	重排行程、计算总费用并检查冲突。
Observation:	总费用 7,620 元，无时间冲突，硬约束全部满足。

终止 / 升级

条件满足 → 输出带依据的方案 | 信息不足 → 继续循环 | 支付或退订 → 人工确认

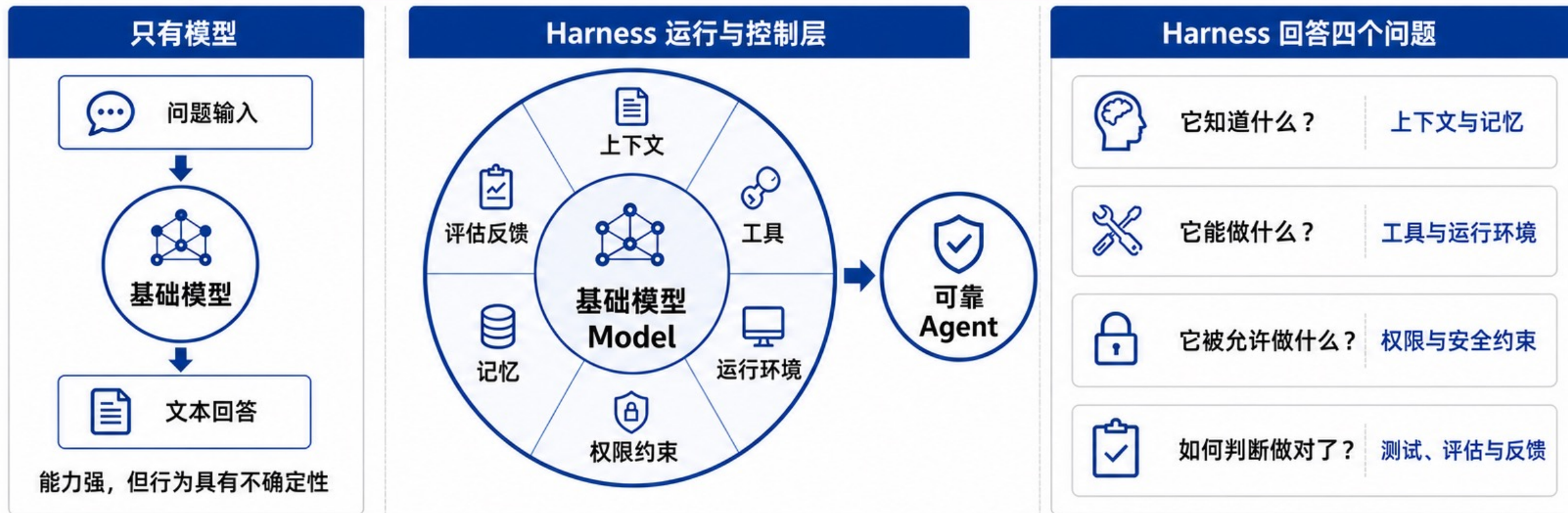
智能体的内容输入：Prompt 与 Context

- Prompt 工程设计角色、目标、约束与输出格式，明确模型“当前应当做什么”
- Context 工程动态组织任务状态、检索知识、工具说明与历史记忆，决定模型“此刻能够依据什么行动”



智能体的可靠执行：Harness 工程

- Harness 是包裹在模型周围的运行与控制系统，负责组织上下文与记忆，并连接工具、数据和执行环境
- 它同时管理权限、预算、超时和异常，通过结果验证与反馈提高执行的可靠性



Reliable Agent = Model + Harness



研究对象从“如何让模型回答得更好”，
转向“如何让 **Agent** 在真实环境中可靠地完成任务”。

智能体的知识利用框架

- 智能体通过的闭环流程，将环境输入转化为可执行决策，并持续评估与修正后续行为
- 知识体系为各环节提供支撑，涵盖领域本体、事实资源、规则策略、任务工具以及记忆经验等核心内容



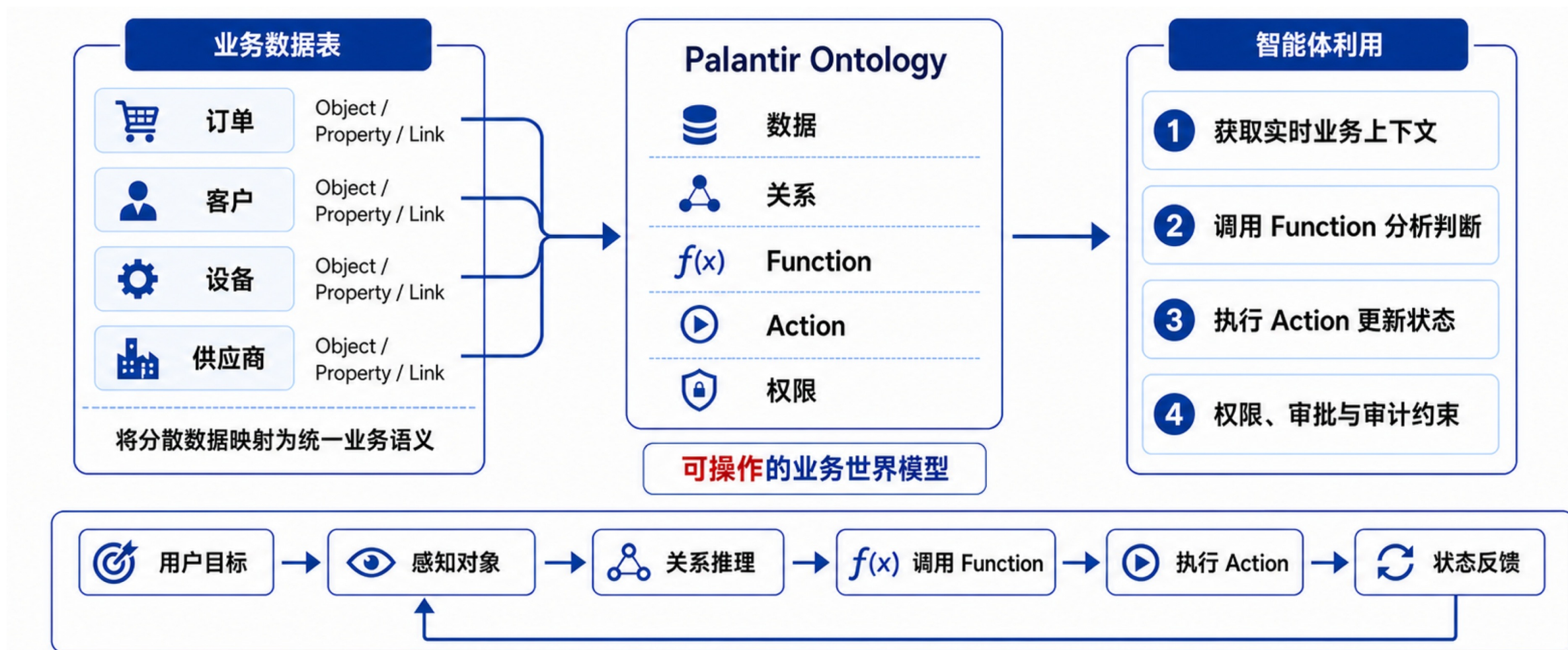
知识对智能体各环节的多阶段支撑

	感知	理解	检索	推理	规划	行动	反馈
1. 领域本体与语义模型	●	●	●	●	●		●
2. 事实知识与知识资源		●	●	●	●		●
3. 规则、约束与策略			●	●	●	●	●
4. 任务、工具与能力知识				●	●	●	●
5. 记忆与经验知识		●	●	●	●		●

● 表示主要支撑关系，不表示绝对边界

知识本体：为智能体建立世界模型

- 本体以对象、属性和关系描述业务世界，以函数和 Action 定义可调用的计算与操作
- 权限进一步划定访问和行动边界，使智能体既理解“世界中有什么”，也知道“能够做什么”



知识检索：为任务决策提供依据

- 文档 RAG 提供非结构化事实与证据，Graph RAG 利用实体关系完成关联和多跳检索
- Ontology-guided RAG 进一步以本体约束查询扩展、结果组织与推理路径

? 统一问题：我的航班班次被更换，会影响后续行程吗？应如何处理？

能力增强

1 文档 RAG | 找相似内容

机制：切分文档 → 向量化 → Top-k 召回 → 重排序 → 生成
典型方法：Dense Retrieval; BM25 + Embedding; Cross-Encoder Reranker



政策与退改签须知
•
•
•

召回航班变更政策与退改签须知，
生成通用处理建议



相关文本可覆盖 |
但难判断具体联程影响

2 Graph RAG | 沿关系找证据链

机制：实体链接 → 子图召回 → 多跳路径 → 图文证据融合
典型方法：KG-RAG; Global-Local GraphRAG; Path Retrieval



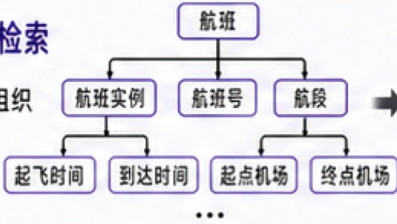
连接乘客—订单—原/新航班—
中转机场—后续航段，计算衔接
时间并定位受影响节点



关系路径可追踪 |
支持跨文档推理

3 Ontology-guided RAG | 用领域语义约束检索

机制：意图映射 → 类型约束 → 实体消歧 → 规则推理 → 结果组织
典型方法：Query Expansion; Schema-constrained Retrieval; Rule-guided Reasoning



- ☑ 转机需满足最短中转时间
- ☑ 同一订单下的航段关联
- ...

检索结果（结构化）
•
•
•
•

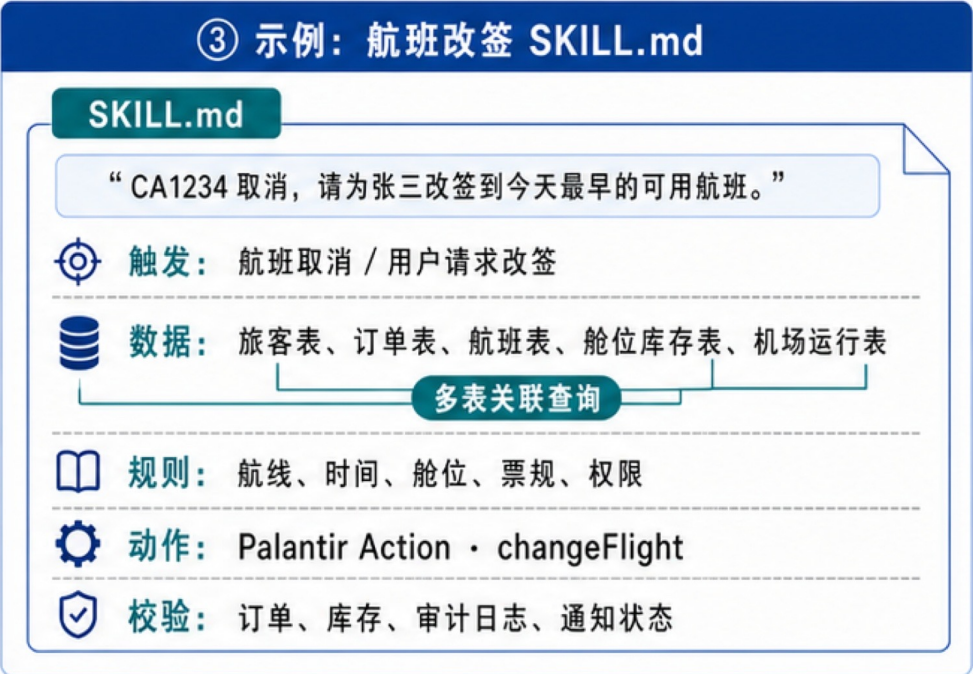
区分航班实例/航班号/班次，
以联程类型和最短中转时间约束
检索，输出受影响航段、原因、
改签方案与政策依据



语义一致、
约束明确 |
结果可执行

智能体技能：为智能体复用过程知识

- 技能描述的不只是“知道什么”，更是“如何完成任务”
- 它将适用条件、任务目标、执行步骤、工具调用、异常处理和验证标准封装为可复用的执行单元



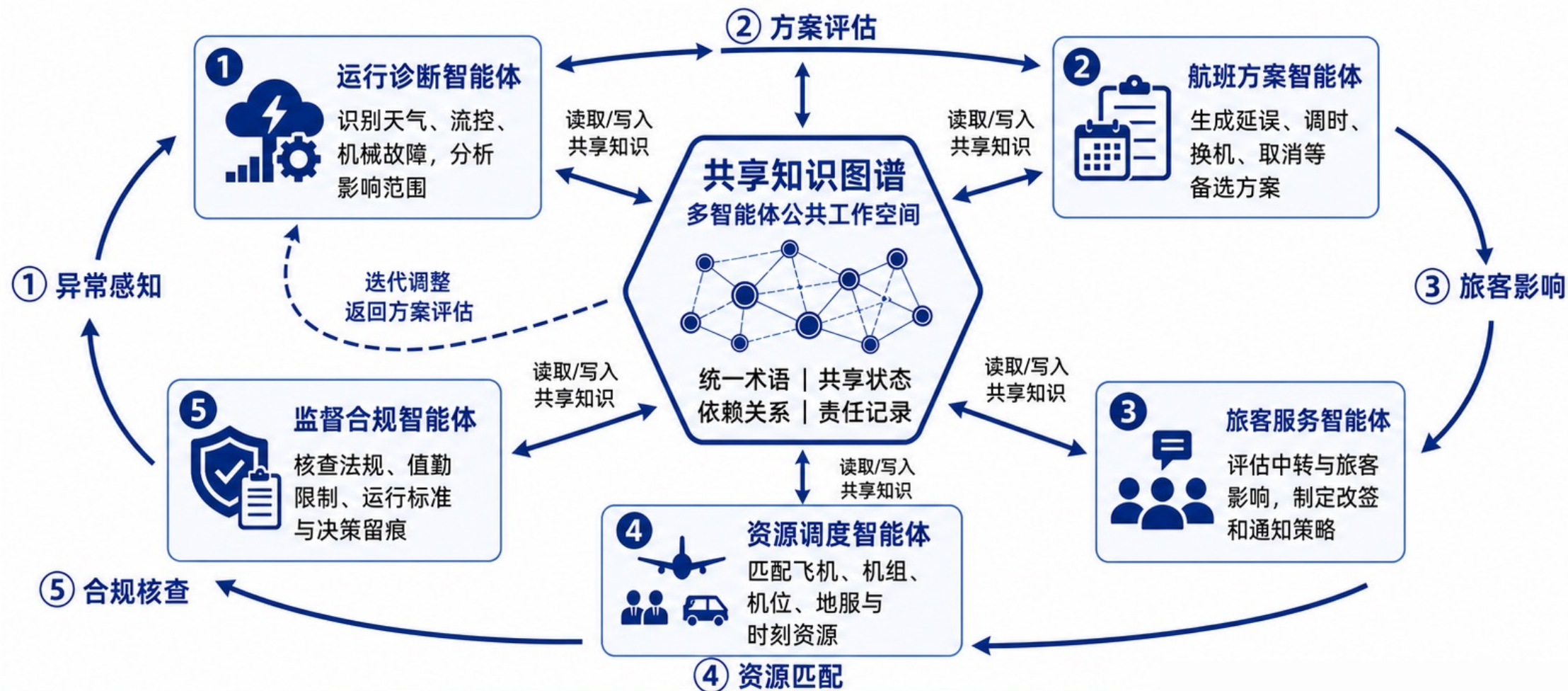
智能体记忆：为智能体提供历史经验支持

智能体通过外部记忆系统，实现经验的持续沉淀与复用。短期记忆维护当前任务状态，长期记忆沉淀历史经验，领域知识提供事实与规则支撑，三者协同提升决策的准确性、连贯性与适应性



多智能体协作：知识作为公共工作空间

- 多智能体系统，通过共享知识空间维护共同目标、任务状态、证据、中间结果与角色分工
- 版本、冲突和责任记录使协作过程保持一致、可追踪和可追责



总结

本体建模

- ✓ 业务对象、关系与规则分散在多系统，智能体在利用时缺乏统一语义
- ✓ Palantir 本体统一语义，为智能体建立世界模型，既理解"世界有什么"，也知道"能做什么"

知识构建

- ✓ 多源数据天然存在重复、冲突与缺失，难以直接应对复杂应用
- ✓ 知识构建通过知识的获取、融合、加工，让事实和规则持续可追溯、可校验、可维护

智能应用

- ✓ 智能体在应对复杂任务存在不可靠问题，难保事实正确与能力边界
- ✓ 知识工程为其补上可更新、可验证、可治理的事实与行动约束

敬请批评指正！

汇报人： 盛傢伟
shengjiawei@iie.ac.cn