



本体之上

AI时代的数据管理与智能决策

树懒老K



个人网站



个人微信



公众号

本体之上

树懒老K

作者：树懒老K

出版：树懒老K

年份：2026

版权所有 · 未经许可不得转载

目 录

第1章 AI数据管理的范式之变

【决策者摘要】

- 1.1 一个排产AI的"翻车"现场
- 1.2 数据管理的三次范式跃迁
- 1.3 AI对数据的三个新要求
- 1.4 语义层的"涌现"
- 1.5 本书的叙事路径
- 1.6 本章小结

第2章 为什么是本体

【决策者摘要】

- 2.1 管字段还是管知识
- 2.2 传统数据治理的边界
- 2.3 本体的三个独特能力
- 2.4 本体与企业现有治理框架的关系
- 2.5 本章小结

第3章 本体论：从哲学概念到企业AI基座

【决策者摘要】

- 3.1 哲学概念，企业实践
- 3.2 本体的核心构件：对应企业数据管理

3.3 本体 vs. 容易混为一谈的概念

3.4 联结主义+符号主义：企业AI的必然选择

3.5 本体与企业架构的嵌入

3.6 本章小结

第4章 "事实-事理-行动"三元模型

【决策者摘要】

4.1 一个框架，三个问题

4.2 事实数据：让AI"感知"状态

4.3 事理模型：让AI"理解"逻辑

4.4 行动指令：让AI"执行"操作

4.5 三层联动：从感知到行动的完整路径

4.6 本章小结

第5章 "7+1"语义规范体系

【决策者摘要】

5.1 从"一句话"到"一套语言"

5.2 RDF：基础语言——定义企业中的各类实体和它们的属性

5.3 OWL：高级语言——消除语义歧义

5.4 SKOS：术语管理——筑牢语义基础

5.5 SWRL：规则语言——支撑AI自主决策

5.6 OWL-S：流程语言——定义AI可以执行的步骤

5.7 ODRL：权限语言——构建AI操作的安全边界

5.8 SPARQL：查询语言——让AI随心所欲地"问问题"

5.9 "+1"：目标与评估——让本体工程有方向

5.10 本章小结

第6章 本体建模工程化方法

【决策者摘要】

6.1 本体建模的"工程化"在哪里

6.2 "29句话": 从业务专家到本体模型的预处理方法

6.3 AI辅助建模与专家审核

6.4 "建用优复"生命周期管理

6.5 "点线面"迭代式实施路径

6.6 本章小结

第7章 本体增强型AI Agent

【决策者摘要】

7.1 Agent的核心工作流: 意图→语义→推理→行动

7.2 意图对齐: 把"人话"翻译成"本体查询"

7.3 本体检索与调用

7.4 本体驱动增强推理

7.5 决策到行动的转化

7.6 Agent构建框架

7.7 本章小结

第8章 制造业落地案例

【决策者摘要】

案例A 精密机床故障根因分析

案例B 汽车零部件BOM变更影响分析

案例C 电子制造跨系统物料追溯

案例共性总结

第9章 实施路线图与组织保障

【决策者摘要】

9.1 三阶段路线图

9.2 组织：三种角色

9.3 五个常见失败模式

9.4 本章小结

第10章 展望：本体与大模型的融合演进

【决策者摘要】

10.1 LLM不能替代本体的三个原因

10.2 三条融合演进路径

10.3 企业的"三重世界"协同

10.4 结语

第1章 AI数据管理的范式之变

“AI进了工厂，听不懂物料编码。”

【决策者摘要】

企业数据管理正在经历第三次范式变革：从“记录型”（ERP时代管字段）到“分析型”（BI时代管指标），再到“AI型”（大模型时代管知识）。AI不是升级版的BI——它对数据的要求完全不同。AI需要的不只是“数据准确”，而是“数据可被理解”——知道物料之间是什么关系、质量标准代表什么含义、业务规则怎么推理。做不到这一点，AI Agent再聪明也是“睁眼瞎”。本章建立全书的叙事起点：为什么“让企业数据被AI理解”是企业AI落地的第一仗。

1.1 一个排产AI的“翻车”现场

2025年，华东某机械制造厂的智能制造负责人张总，做了他认为是“水到渠成”的一件事——把公司刚部署的AI排产Agent接入了ERP系统。

Agent的能力看起来很强：读懂本周的所有生产订单、分析各产线的产能负荷、给出优化后的排产计划、自动生成物料需求清单。实验室测试跑了三轮，准确率98%。

上线第三天，Agent犯了第一个错误。

产线上缺了0102003581这个螺栓，Agent根据库存数据和BOM用量算出：缺了480件，触发紧急采购。

问题是——ERP里0102003581这个螺栓有一个”代用料”：0102003582，规格一样只是材质从碳钢换成了不锈钢。车间一直在混用，MES系统里做的是真实的代用记录。但Agent不知道。

Agent只读了ERP的库存和BOM。它从不知道这个螺栓跟别的螺栓之间有一个叫”代用”的关系。它从不知道MES里有一个”实际消耗料号”跟”工单物料号”长得不一样——因为代用料的事情，不是写在数据库字段里的，是写在车间主任的脑子里的。

这个翻车不是Agent不够聪明，是它读到的数据不够”有含义”。

Agent看到的是字段值和数量，看不到它们之间的关系。对它来说，0102003581和0102003582就是两行主数据，各自独立，没有语义关联。而”代用关系”——这个对车间主任来说最自然不过的常识——对Agent来说完全不天然。

张总的教训：**把AI接上数据不等于AI能理解数据。** 连接和数据理解之间，隔着一整层”语义”。

1.2 数据管理的三次范式跃迁

过去三十年，企业数据管理经历了三次根本性的范式跃迁——每次都伴随着一个核心问题的改变。

范式一：记录型数据管理（1990年代–2010年代）

核心问题：数据”准不准”。

ERP系统大规模部署的时代，数据管理的核心任务是**确保数据的完整性和一致性**——物料号不重复、库存数量不遗漏、主数据的必填字段不能空。这一时期的代表框架是**DAMA数据治理**（质量、元数据、主数据管理）、**MDM（主数据管理）、编码标准化**。

数据在这个阶段扮演的角色：**记录和事务处理**。ERP的任务是把企业运行的事实数字化——采购了什么、生产了什么、交付了什么。

范式二：分析型数据管理（2010年代–2020年代）

核心问题：数据”有没有用”。

数据仓库、BI、大数据平台的时代。企业不再满足于”记录”，开始主动”分析”——看趋势、找异常、做预测。数据管理的核心任务升级为**数据的可用性和集成性**——把ERP、MES、CRM的数据拉通、清洗、建模。

这一时期的代表技术：数据仓库（DW）、数据湖（Data Lake）、ETL/ELT工具、数据虚拟化。

范式三：AI型数据管理（2020年代至今）

核心问题：数据”能不能被理解”。

范式一管字段完整性，范式二管数据集成性。两个范式管的核心对象都是**数据本身**——数据值对不对、有没有缺失、能不能拉通。

范式三管的东西变了。它管的不是数据，是**数据的含义**。

AI需要的不是干净的字段值，而是能推理的知识结构。一个AI Agent在分析排产时，它需要的不是”库存量的字段值是2800”，它能理解数字。它需要的是”2800是整个0201库位的库存，但实际可用于这条产线的只有那点因为代用料的关系”——这是一种复合的语义推理。

传统数据管理工具无法提供这种推理层的信息。字段的”准不准”管住了，数据的”有没有用”解决了——但数据的”被理解”这个维度，是完全新的。

三次范式跃迁的核心差异：

表 1-1

	记录型（ERP时代）	分析型（BI时代）	AI型（现在）
核心问题	数据准不准	数据有没有用	数据能不能被理解
管理对象	字段	指标	知识
典型框架	DAMA、MDM	数据仓库、数据湖	本体、知识图谱
数据角色	记录事实	辅助决策	驱动智能

1.3 AI对数据的三个新要求

为什么范式一变，数据管理的工作就得重做？因为AI对数据提出了三个传统数据管理从没想过要解决的问题。

要求一：数据必须”可理解”

传统系统只需要”能读”数据：字段类型是CHAR(18)、长度是18位、必填字段不能空。做到这一步，系统就能正常运作了。

AI需要的不止于此。一个物料对AI来说，”0102003581”这个字符串意味着什么？它不知道这是螺栓还是齿轮，不知道它跟0102003582之间有替代关系。AI需要的不是字段值，是这个字段

值背后的**业务概念**——“螺栓”是某种产品的子类, ”M8×20”是它的规格参数, ”0102003582是它的代用料”是关系。

可理解 = 数据的业务含义被形式化地表达, 可以被机器读取和解释。

要求二：数据必须”可推理”

传统系统的逻辑是业务规则写死的：IF inventory < minimum THEN order。这种规则只能处理已知的情況。

AI的目标是处理未知的情况。它需要能从已知数据中推导出新的事实。当发现A厂产的螺栓质量在连续三个批次中都不合格时, 不只是标记A厂的批次有问题, 还要能推理出”A厂的螺栓应暂时从供应商认证列表中移除, 涉及哪些BOM、哪些产线会受影响”。

可推理 = 数据之间的关系和规则足够清晰, AI能自主推导出新结论。

要求三：数据必须”可追溯”

当AI Agent给出了一个排产建议、推荐了一个替代料、判断了一个故障根因时, 业务人员必须能问:”你为什么这么判断?”——并且Agent必须能答得出来。它的推理路径必须能回溯、每一步的规则来源和数据引用必须清晰。

可追溯 = AI的每个决策有明确的推理链, 可人工验证每一步的正确性。

1.4 语义层的”涌现”

可理解、可推理、可追溯——这三个要求指向同一个解决方案：一个能让数据”有含义”的语义层。

传统的数据管理框架管的是”字段”。DAMA的11个数据管理领域中没有”语义管理”这个维度。数据血缘管的是数据从哪里来的（源系统），不是数据代表什么（含义）。元数据管的是数据结构（表名、字段类型），不是数据含义（这个字段在业务中代表什么）。

语义层的任务，就是补充这一层缺失。它不替代元数据或数据治理的工具，而是在它们之上加一层”翻译层”——把”0102003581”翻译为”物料：螺栓M8×20”，把”跟它相同的代用品是0102003582”翻译为一条关系，把”这个物料不可替代，BOM变更需要技术中心审批”翻译为一条规则。

有了语义层，数据就不仅是字段和记录——它们是”可理解的业务实体”。

1.5 本书的叙事路径

这本书将沿着以下路径展开：

- **上篇（第1-2章）：**回答”为什么”——为什么AI时代需要一个新的数据管理范式，为什么这个范式是本体

- **中篇（第3–5章）：**回答”是什么”——本体在企业AI中的角色，核心框架是”事实–事理–行动”三元模型
- **下篇（第6–9章）：**回答”怎么做”——建模工程化方法、Agent调用、三个制造业纵深案例、实施路线图
- **终篇（第10章）：**展望——LLM与本体的融合演进

本书的起点是”让企业数据被AI理解”，终点是”让企业AI能做对决策”。中间的路，就是本体。

1.6 本章小结

1. 数据管理正在经历第三次范式跃迁——从管字段到管知识
2. AI需要数据”可理解、可推理、可追溯”——这是传统数据管理框架从未涉及的维度
3. 语义层的缺失是企业AI落地的第一道墙，本体是补上这一层的方法论
4. 本书的核心命题：用本体论构建语义基座，让企业数据被AI理解

延伸阅读

- 《华为数据之道》 — 华为数据治理体系的系统化呈现，从实践层面了解企业数据管理的演变
- DAMA–DMBOK2 — 数据管理知识体系的国际标准，理解传统数据管理的边界和局限

- 《数据空间探索与实践》 — 华为的数据空间方法论，本书的核心参考框架

第2章 为什么是本体

“你管得住字段，管不住含义。”

【决策者摘要】

传统数据治理的核心对象是字段——完整性、唯一性、一致性。DAMA的11个数据管理领域中没有“语义管理”这个维度。AI时代需要的是一种新的治理范式：从管字段到管知识。本体提供了实现这一范式的形式化工具——它能表达业务概念、定义类与关系、编码规则与约束。本章回答“为什么是本体”：传统治理的边界在哪、本体的独特能力是什么、本体与企业现有数据治理框架怎么衔接。

2.1 管字段还是管知识

2024年，某电子制造企业的数据治理委员会给出了最新的主数据质量报告：物料完整率99.8%，供应商完整率97.5%，客户完整率96%。几个指标整体向上，黄色警告区缩小。

然后CIO问了三个问题：

1. “我们ERP里记录的供应商评级是基于什么？是依据交付准时率还是依据产品质量的一致性？”

2. “当我们把一个物料从‘A类’调到‘B类’时，这个分类调整会影响哪些采购审批流？哪些财务核算规则？”
3. “我们的数据治理报告上说‘99.8%的物料字段完整’。那我现在问一个简单的问题：哪些物料的供应商有双源备份，供应链中断后可以在48小时内切换到第二个供货源？”

第一个问题——数据治理委员会查了规程文档，发现供应商评级方法在2019年做过一次调整，新老标准同时用了两年，现在还有部分供应商的评级是按旧标准来的。

第二个问题——发现物料的A/B/C分级调整在系统中只是标记字段的变化，根本没有自动关联到对应的审批流和核算规则。审批流和分级之间的关系是靠负责的业务人员手动检查的。

第三个问题——**没有人能在系统中回答这个看似简单的问题。**因为“双源备份”的概念没有作为字段存在——它分散在供应商表中、在采购策略文档中、在供应链风险分析师的经验中。

这三个问题暴露了同一个根因：传统数据治理管的是字段结构，不管字段含义。而这三个问题，每个都涉及“含义”的层面——供应商评级的含义、A/B/C分类调整的含义、“双源备份”的含义。

2.2 传统数据治理的边界

DAMA数据管理知识体系（DMBOK）定义了11个数据管理领域：数据治理、数据架构、数据建模与设计、数据存储与操作、数据安全、数据集成与互操作、文档与内容管理、参考数据与主数据、数据仓库与BI、元数据、数据质量。

其中与“数据含义”最接近的两个领域是**主数据管理**（管核心业务实体）和**元数据**（管数据结构）——但这两个都停留在“语法层”。

主数据管理的边界：它告诉你”物料编号0102003581在数据库中是一个CHAR(18)的字段，必填且唯一”。它定义了物料主数据的标准字段——物料号、名称、分类、计量单位、标准价格。但它不会告诉你”0102003581这个物料在发动机总成BOM中扮演功能角色’密封连接’”。

元数据的边界：它告诉你”这张名为MARA的表的源系统是SAP，数据更新频率是每天，这个字段的类型是DECIMAL(13,2)”。它定义了数据的形态和结构。但它不会告诉你”MARA-MATKL字段中的‘0010’代表’紧固件’类型”——这条知识被锁在了T023T这张参考表中，靠人工查询和业务人员的记忆来维护。

传统治理管语法，不管语义。这就是它的边界。而AI需要的，恰恰是在这个边界之外的东西。

2.3 本体的三个独特能力

本体，不是另一个“更好的主数据管理系统”——它是一层不同的东西。它管的是“含义”。

传统主数据管理系统问：“这个数据完整吗？格式对吗？编码重复吗？”

本体问：“这个物料代表什么概念？它跟BOM之间是什么关系？它可以被谁替换？替换它需要审批哪些部门？”

具体来说，本体提供了传统数据管理缺少的**三个核心能力**。

能力一：概念统一

不同系统对“同一个东西”有不同的叫法和编码——第1章中0102003581这个物料在ERP中是一个编码，在MES中是另一个编码，在WMS中是又一个编码。传统数据管理用“映射表”来解决这个问题——每次都手动建一个新的映射。

本体换了一个思路：不建立更多映射，而是**定义一个独立于系统的“概念”**。物料“螺栓M8×20”是一个独立的概念——它是紧固件的一个子类，有独立的属性（规格、材质、强度）。每个系统里各自的物料编码，都被映射到这个概念上。这样，当需要跨系统查询时，你不需要知道0102003581在MES中的对应编码——你只需要查询“物料本体中的M8×20螺栓”，所有系统中的对应实例都会自动关联。

概念统一不是技术手段，是业务逻辑的成果。 需要业务专家坐下来，把”这个物料在不同的系统中代表什么”这件事情想清楚、写下来、形式化。

能力二：关系表达

传统数据管理表达关系的方式是外键——BOM表里有一个”物料ID”字段连到物料表。这种关系告诉系统”这两行数据是关联的”。

但它不说清楚”它们是什么关系”。BOM中的”物料ID”只是说”这个物料出现在这个BOM中”，没有说它在这个BOM中扮演什么角色（结构支撑？密封连接？减震缓冲？）、它的可替代性如何（唯一来源？多源？可替换？）、它变更后会影响什么。

本体把”它们是什么关系”明确写出来：

物料A - used-in → BOM节点B （物料A出现在BOM节点B中）
BOM节点B - has-function → "密封连接" （节点B的功能角色）
物料A - substitutable-by → 物料B （物料A和B之间可替代）
物料A - supplied-by → 供应商S （物料A从哪个供应商采购）

这些关系不是外键能做到的——因为它们需要**命名和分类**。关系名（used-in、has-function、supplied-by）是语义行为的一部分——它们本身就携带着业务含义。

能力三：规则推理

传统系统的规则是写死在代码里的。当业务规则变化时，需要重新开发和测试。

本体的方式不同：业务规则以机器可读的方式定义，与数据和代码解耦：

```
IF 物料类型 == "紧固件" AND 使用场景 == "户外"  
THEN 表面处理 == "达克罗" OR "镀锌"
```

这个规则写在本体层。当AI Agent分析一个用在户外的紧固件时，它能自动推断：这个物料的表面处理应该是达克罗或镀锌，不是的话意味着可能需要审查。

更重要的是：当规则变了（比如有了新的表面处理工艺），只在一条规则上改就行了，不需要修改系统和代码。

2.4 本体与企业现有治理框架的关系

一个常见的误解：本体是一种新的技术栈，需要替换现有的数据治理系统。

本体的角色不是替代，是**补充**。它像一座桥，从现有系统的“数据层”通向AI需要的“认知层”：

原有的数据治理（继续运行，不改动）：

主数据管理 → 管物料、供应商、客户的基本数据

元数据管理 → 管表结构、字段类型、数据血缘

数据质量管理 → 管完整性、唯一性、一致性

本体层（新增）：

概念统一 → 把一个物料在不同系统中的描述归为一个概念

关系表达 → 定义物料与BOM、供应商、质量标准之间的语义关联

规则推理 → 业务规则形式化后应用于AI推理中

一个新的质量管理维度：知识完整性

本体的引入把数据治理的范围从语法层扩展到了语义层。与此同时，传统的数据治理维度（字段完整率）仍然有效——“字段不完整”的数据，它的语义映射也无法执行。本体和数据治理是一个共生关系，不是竞争替代。

对应这一扩展，数据治理可以新增一个维度：**知识完整性**——企业的核心业务概念有没有被本体覆盖到？关系表达有没有遗漏？规则有没有过时？——这些新KPI正在成为评估企业数据对AI供给能力的关键指标。

2.5 本章小结

1. 传统数据治理管的是字段结构，本体管的是数据含义——两者不在同一个层次

2. 本体提供了传统数据管理缺失的三个核心能力：概念统一、关系表达、规则推理
3. 本体不是替代品，是补充——在现有数据治理体系之上加一层语义层
4. 新增的治理维度”知识完整性”将成为企业AI时代数据对AI供给能力的关键指标



延伸阅读

- **DAMA-DMBOK2 第3章: 数据治理** — 传统治理框架的全景，对照本章能更清楚地看到边界在哪
- **《华为数据之道》第7章: 信息架构** — 华为怎么做信息建模与数据资产目录
- **Tom Gruber, “A Translation Approach to Portable Ontology Specifications” (1993)** — 本体定义的经典论文，简短有力

第3章 本体论：从哲学概念到 企业AI基座

“本体不是学术词汇，是你企业数据缺失的那层‘含义’。”

【决策者摘要】

“本体”来自哲学——研究“存在本身”的学问。在计算机科学中，本体是对共享概念体系的明确形式化规范。这个词听起来很学术，但放在企业AI的语境下，它的含义非常实际：本体就是“让机器理解业务上下文”的语言。本章不讲学术黑话，用企业数据管理的场景讲清楚：本体是什么、它的核心构件（类/关系/属性/规则）在企业中对应什么、它跟数据字典、ER模型、知识图谱的区别。最后讨论联结主义+符号主义的融合——为什么纯概率推理的LLM需要本体来补上“确定性推理”这一块。

3.1 哲学概念，企业实践

“本体论”是古希腊哲学的一部分，亚里士多德用它来问“存在是什么”。两千四百多年后，这个词被计算机科学家Tom Gruber引入了信息科学。他给出了一个精妙的定义：

“An ontology is a formal, explicit specification of a shared conceptualization.”

这句话翻译过来是：“本体是对共享概念体系的明确形式化规范。”听起来又绕了。但拆开看，每个词都很实在：

- **共享** — 不是一个人的理解，是多个系统和人的共同理解
- **概念体系** — 不是数据本身，是数据背后的“含义”——类、关系、规则
- **明确** — 写在纸面上，不是只在人的脑子里
- **形式化** — 格式严格到机器能自动处理、推理

放在企业环境下，这段话的意思就是：**把你企业里所有核心的概念（物料、BOM、质量规范、供应商）和它们之间的关系（用什么、怎么替代、影响谁），用机器能读的形式写出来。**

这个概念对应到企业实践，就是给ERP数据加一层“含义说明书”。ERP记录了一个字段叫“MATNR”的值是多少——本体告诉你这个字段代表的业务概念是什么，跟谁有关，什么条件下能变。

3.2 本体的核心构件：对应企业数据管理

本体有四个核心构件。把它们跟企业数据管理的场景一一对应：

类：概念的抽象分类

在本体中，“类”对应的是业务概念的分类。制造企业最常见的类：



这个层级结构跟ERP的物料组或产品分类很像——但不同之处是：本体中的每个类携带该类固有的属性和规则。比如“紧固件”这个类有属性“螺纹规格”和规则“配合扭矩 $\leq 25\text{N}\cdot\text{m}$ ”。ERP的物料组只是给物料打了一个分类标签，不打规则。

关系：数据之间的语义连接

ERP用外键来表达两个表的连接：BOM表有一个“物料ID”指向物料表。外键告诉系统“这两行数据有关联”，但不告诉你“这是什么关联”。

本体用**命名的关系**来补充这个缺失：

表 3-1

关系	含义	企业场景
used-in	被用于	物料A被用于BOM节点B
supplied-by	由...供应	物料A由供应商S供应
substitutable-by	可被...替代	物料A可被物料B替代
has-standard	须满足...标准	物料A须符合质量规范Q
before	在...之前	工序A在工序B之前

关系名本身携带含义，这给AI推理提供了基础——当AI需要知道”变更物料A会影响谁”，它能查询所有以物料A为起点的关系链。

属性：类的特征维度

每个类有属于自己的属性集。属性比ERP的字段更”业务化”：

类：螺栓(Bolt)

属性：

螺纹规格(threadSize)：M8
长度(length)：20mm
材料(material)：碳钢
强度等级(grade)：8.8
功能角色(functionalRole)：密封连接
替代类型(substitutionType)：兼容替代

注意属性”功能角色”——这个值在ERP中一般不存在。它需要业务专家人工定义。但正是这类属性，让AI能理解”这个螺栓在BOM中起什么作用，替代它需要注意什么”。

规则：业务的约束和逻辑

本体的真正力量在于规则——嵌入在模型中的业务约束：

```
规则1: IF 紧固件.使用场景 == "高温" THEN 材料 ≠ "碳钢"  
规则2: IF BOM变更.类型 == "非兼容替换" THEN 需要技术中心审批  
规则3: IF 供应商.质量评级 == "C级" THEN 不能供应关键安全件
```

这些规则不写在代码中，而是作为本体的一部分独立维护。当AI Agent在评估BOM变更时，它能访问这些规则来判断变更的合规性——不需要有人记住每一条规则。

3.3 本体 vs. 容易混为一谈的概念

让数据架构师们最困惑的一点是：本体听起来跟很多已知概念很像，但它其实不在同一层次。

表 3-2

概念	与本体关系	关键区别
数据字典	只到语法层	定义字段类型和格式，不定义业务含义
ER模型	只到关系层	表达表之间的关联，不表达关系的语义和规则
分类编码 (eCl@ss)	是本体的”分类法” 部分	只覆盖is-a层级，没有跨类关系和规则推理
知识图谱	本体是它的”模式层”	知识图谱 = 实例数据 + 本体（模式层）
数据治理	本体管语义，治理管字段	共生关系——本体扩展了治理的边界到含义层

一个简单的分法：如果ERP的数据字典是一本书的目录（结构），ER模型是章节之间的交叉引用（关联），那本体就是这本书的”中文翻译”——它让你理解每一个字是什么意思。

3.4 联结主义+符号主义：企业AI的必然选择

人工智能领域一直存在两条路线之争：

- **联结主义(Connectionism)** — 用统计学习、神经网络从数据中找模式。LLM就是典型的联结主义产物

- **符号主义(Symbolism)** — 用形式化逻辑表达知识和规则，通过推理得出结论。本体是符号主义的核心工具

联结主义的LLM在企业应用中面临三个核心问题： 1. **幻觉** — LLM自信地编造了不存在的规则和关联 2. **不可追溯** — LLM无法告诉你它为什么得出某个结论 3. **不懂企业私域** — LLM不知道”0102003581”在特定工厂中代表什么

符号主义恰好补上了这些弱点。本体的规则是确定的、可追溯的、基于企业私域知识的。但它也有自己的弱点——覆盖面有限，无法应对模糊和不确定的情况。

融合是唯一的解法：LLM做不确定的感知和语言交互，本体做确定的推理和事实校验。

在企业环境中，一个有效的模式是：LLM理解用户的问题并转化为结构化的查询，本体承载业务规则并进行确定性推理，LLM最终将结构化结果组织为自然语言回答。两条线各司其职，没有谁替代谁。

《数据空间探索与实践》中也明确指出了这一点——纯概率推理难成企业AI的落地有效路径。联结主义和符号主义的融合，才是企业AI的必然选择。

3.5 本体与企业架构的嵌入

本体在企业架构中的位置，不是在哪个具体系统里，而是在所有系统的数据之上：

【业务应用层】ERP | MES | WMS | PLM | QMS | CRM

【数据分析层】BI | 报表 | 数据湖 | 数字孪生

【AI应用层】Agent | RAG | 智能决策

↑

【语义层—本体】 ← 新增的一层

↑

【数据层—字段和表格】 ← 原有的数据

语义层的角色是**知识整合和推理**，而不是数据的重复存储。数据的”物理存储”在原系统或数据湖中，语义层是数据的”认知视图”——它不复制数据，只是给数据加上含义。本体层链接底层的”事实”，并引入对”事理”的表达——即规则、逻辑和流程——赋能上层AI应用的”行动”。

一个企业的本体应该建在哪里？ 不是在ERP系统内，不是在数据湖内，而是在一个独立于系统的”语义数据库”中——专门用来存储本体模型和语义映射。现有的ERP、MES、PLM数据通过映射配置被解析到语义层中，而不是移动或复制。

3.6 本章小结

1. 本体的核心构件（类/关系/属性/规则）——对应企业数据管理的实际需求

2. 本体与数据字典、ER模型、知识图谱不是同一层次——它是“含义层”
3. 联结主义+符号主义的融合是企业AI的必然路径，本体是符号主义的核心工具
4. 语义层在架构中的位置：在所有系统的数据之上，为AI提供“认知视图”



延伸阅读

- 《数据空间探索与实践》第4章 — 本体从哲学到计算科学的完整脉络
- Gruber, T.R. “A Translation Approach to Portable Ontology Specifications” (1993) — 本体定义的奠基性文献
- Ontology Development 101 (Noy & McGuinness, Stanford) — 本体建模最好的入门指南

第4章 "事实—事理—行动"三元 模型

“让AI知道发生了什么、理解为什么会这样、执行该怎么办。”

【决策者摘要】

华为《数据空间探索与实践》的核心创新是把本体的作用拆解为三个层次——事实、事理、行动。事实数据让AI感知“当前状态是什么”，事理模型让AI理解“为什么会这样、应该怎么走到下一步”，行动指令让AI执行“该做什么”。本章详细展开这三层模型：每层的定义、构建方式、与企业数据管理的对应关系，以及三层之间的联动机制。这是全书的方法论核心——没有三层模型，本体只是一个分类系统；有了三层模型，本体成为AI推理的“大脑框架”。

4.1 一个框架，三个问题

传统的数据管理回答的是一个问题：“**这个数据是什么**”。

AI时代的数据管理需要回答三个问题：

1. **事实层**：当前的业务状态是什么？（“库存2800件，工单缺料480件”）
2. **事理层**：为什么会走到这一步？业务背后的规则和逻辑是什么？（“因为代用料未录入系统，BOM未及时同步到MES”）

3. 行动层：接下来该做什么？（“触发紧急采购，通知三个部门，暂停受影响工单”）

这三个问题分别对应本体驱动的数据管理的三个维度的构建：

- **事实数据** = 让AI知道发生了什么
- **事理模型** = 让AI理解为什么会这样
- **行动指令** = 让AI执行该怎么办

这组三元模型是《数据空间探索与实践》中最核心的方法论贡献。它把本体从“企业数据的分类系统”提升到了“AI推理的框架”——不再是只帮系统管理数据，而是帮AI做出有依据的决策。

4.2 事实数据：让AI“感知”状态

什么是事实数据

事实数据回答的是“**当前是什么状态**”。它涵盖的是企业运营中的所有事实性信息：

- **物理事实**：库存量、产线状态、设备温度
- **业务事实**：工单状态、订单金额、交付时间
- **关系事实**：物料A当前属于产品B的BOM，供应商S供应物料A

在本体语境下，事实数据的核心特征是**客观、可验证、有唯一来源**。一个物料在ERP中的库存数量是一个事实——它有一个确定的数据库记录作为来源。

事实数据的本体表达

事实数据在本体中以**实例**的形式存在。每个实例都是一个真实的数据行：

```
ex:material/0102003581 a ex:Bolt ;
    ex:materialId "0102003581" ;
    ex:materialName "螺栓M8×20" ;
    ex:belongsTo ex:category/Fastener ;
    ex:netWeight "0.020"^^xsd:decimal ;
    ex:inventory 2800 .
```

这组表达与ERP中的一个物料行对应——但不同之处在于，它不光记录了这个物料是什么，还记录了这个物料属于哪个分类（紧固件）——这个”属于”关系是本体补充的，ERP的数据行里没有。

事实数据的三要素

一个事实在本体中的表达需要满足三个要素：

表 4-1

要素	来源	示例
身份标识	系统编码/实例ID	物料号”0102003581”
概念归类	本体类	属于”螺栓”类
动态联系	关系	被产品A的BOM使用，库存位于库位0201

关键认知： 分类和关联往往比事实本身更有价值。知道一个物料的库存是2800件这个事实很容易，知道”这个物料的库存分布在两个库位、其中线边库位的200件已经被本次工单锁定”则需要关联多个事实——而AI需要的正是这种关联后的复合事实。

4.3 事理模型：让AI”理解”逻辑

什么是事理模型

事理模型回答的是“为什么会这样、接下来会怎么变”。

如果说事实数据是一张快照，事理模型就是张动态的”逻辑地图”——它描述了业务规则、流程逻辑和约束条件。AI用事理模型来**推导**从当前状态到目标状态的路径。

企业的事理来自多个源头：

- **业务规则：** IF 紧固件.使用场景 == “高温” THEN 材料 ≠ “碳钢”
- **流程逻辑：** BOM变更 → 评估影响 → 通知相关部门 → 执行变更 → 验证
- **约束条件：** 关键安全件不允许降低质检等级
- **分类逻辑：** A类物料需要双源备份，B类物料单源可接受

事理模型的本体表达

事理模型在本体中主要通过**规则引擎**和**关系链**来表达：

```
# 规则：高温场景下紧固件不能使用碳钢
ex:Rule_HighTemp_Fastener a swrl:Imp ;
    swrl:body ([ a swrl:ClassAtom ; swrl:classPredicate e
        [ a swrl:DataPropertyAtom ; swrl:propertyPr
            swrl:argument2 "高温" ]) ;
    swrl:head ([ a swrl:DataPropertyAtom ; swrl:propertyP
        swrl:argument2 [ a swrl:DifferentIndividu
            swrl:argument1 "碳钢" ]]) .

# 关系链：BOM变更的影响推导
# 物料A → used-in → BOM节点 → belongs-to → 产品P
# 物料A → has-standard → 质量标准Q
# 物料A → supplied-by → 供应商S
```

事理模型的三种表达形式

表 4-2

形式	描述	技术实现	示例
规则驱动	IF-THEN形式的业务约束	SWRL、SHACL	IF 供应商评级=C级 THEN 不可供应关键安全件
流程链	业务步骤间的因果链	OWL-S、BPMN+本体	BOM变更→影响分析→部门通知→执行→验证
约束推理	基于类、属性和关系的合规校验	OWL推理器	推理一个物料是否合规用于某个场景

4.4 行动指令：让AI”执行”操作

什么是行动指令

事实层让AI”知道”，事理层让AI”理解”，但AI真正的价值在于”做”。行动指令的作用是把”理解”转化为”操作”。

在企业环境下，一个AI Agent的输出不是一个概率值——它是明确的执行动作：

- 触发一个ERP中的采购申请
- 推送一条通知到质量部门的审批系统
- 暂停受影响的生产工单
- 创建一个BOM变更影响分析报告

行动指令在本体中的表达是：

```
ex:Action_TriggerP0 a ex:Action ;  
    ex:actionType "采购" ;  
    ex:targetSystem "SAP" ;  
    ex:actionPayload "{物料: 0102003582, 数量: 480, 紧急: t  
    ex:requiresApproval true ;  
    ex:notifies "采购科" .
```

行动的三类转化模式

表 4-3

模式	描述	适用场景	信任依赖
指令直驱	AI判断直接触发操作	影响小、规则明确的操作	高（已充分验证）
人工介入	AI推荐→人确认后执行	影响中、需要业务判断的操作	中
协同执行	AI生成方案→多部门审核→执行	影响大、跨部门的决策	低（必须人工）

4.5 三层联动：从感知到行动的完整路径

三层模型的真正力量在于联动——它们不是割裂的三个模块，而是一个完整的认知循环：

【感知】事实数据层：库存2800件，工单缺料480件

↓

【理解】事理模型层：查询代用关系 → 0102003582可替代

查询质量规则 → 0102003582满足Q5-0030

查询供应商 → 供应商S2可供0102003582

↓

【决策】

↓

【行动】行动指令层：生成采购建议 → 通知采购科

生成质量变更建议 → 通知质量科

生成受影响BOM清单 → 通知工艺科

这是第1章中那个翻车的AI排产Agent本应做到的事情——如果有一个完整的三层模型作为它的认知框架，它就能在感知到缺料后，不是盲目地触发采购，而是先经过事理层的推理——查代用关系、查质量规则、查供应商——再给出正确的行动指令。

4.6 本章小结

1. 三元模型是本体的核心框架：事实（知道）→事理（理解）→行动（执行）
2. 事实数据从ERP等系统中来，但不只是字段值——必须包含分类和关联
3. 事理模型编码业务规则、流程逻辑和约束条件——是AI”理解”的源泉
4. 行动指令把”理解”转化为可在企业系统中执行的具体操作

5. 三层的联动形成”感知–理解–行动”的完整认知循环



延伸阅读

- 《数据空间探索与实践》第5章 — 面向AI的三类数据资产目录, ”7+1”语义规范框架
- OWL–S: Semantic Markup for Web Services — 本体驱动的服务描述标准, 对应行动层
- SWRL: A Semantic Web Rule Language — 本体规则语言, 对应事理层的规则表达

第5章 "7+1"语义规范体系

“七种语言，一个目标。每一种对应本体的一项重要能力。”

【决策者摘要】

《数据空间探索与实践》提出了一个“7+1”体系：七种W3C语义标准语言（RDF、OWL、SKOS、SWRL、OWL-S、ODRL、SPARQL）配上一个“目标与评估”维度。它们分别对应本体的不同能力——定义资源、统一概念、管理术语、编码规则、流程串联、权限控制、查询操作、效果评估。本章用企业数据管理的场景逐一解读每个规范：它是什么、什么时候用、不用的时候会怎样。目标是让非技术背景的决策者也能理解”这套东西在解决什么问题”。

5.1 从“一句话”到“一套语言”

第3章提到：本体是让机器理解业务上下文的语言。但一种自然的语言是不够的——人类沟通需要说话、也需要写字、还需要画图表。

本体也一样。一套完整本体需要多种“表达方式”来覆盖不同的需求——分类需要层级语言、规则需要逻辑语言、流程需要行为语言、权力需要权限语言……而所有这些表达方式，都要基

于一个统一的框架来定义核心数据。

这就是“7+1”体系的出发点。它选的七种语言都来自W3C（万维网联盟）的标准体系，保证了开放性和互操作性。”+1”则是给这套技术体系加了一个“价值维度”——每一步都有清晰的评估点，确保本体工程不是“为了做而做”。

5.2 RDF：基础语言——定义企业中的各类实体和它们的属性

它做什么： RDF是本体最基础的语言。它做的事情很简单——描述事物：谁是什么，有什么属性，跟谁是什么关系。

在RDF中，每个数据行都变成了一个“三元组”：

主语	→	谓词	→	宾语
(物料0102003581)		(有名字)		(螺栓M8×20)
(物料0102003581)		(属于类)		(紧固件)
(物料0102003581)		(由...供应)		(供应商S1)

这看起来很像数据库的行。但重要区别是：三元组显式地声明了关系的“含义”——不是“BOM表有一个物料ID字段”，而是“这个物料被用于这个BOM节点，功能角色是密封连接”。

在企业中的场景： 当ERP物料主数据表被映射为本体的RDF实例时，跨系统的语义查询就能直接运行——不需要理会每个系统各自的字段名和编码规则。RDF是企业语义查询的基础。

5.3 OWL：高级语言——消除语义歧义

它做什么： RDF定义了基本的三元组，但无法表达更复杂的逻辑——比如“所有的紧固件都有一个螺纹规格，不是所有的物料都有”——这是类的约束，需要用OWL来表达。

OWL定义了类之间的关系（子类关系）、类的约束（“螺栓是一种紧固件，所以它必须有螺纹规格”）、等价关系的定义（“如果两个物料的规格和材料完全相同，则它们是等效的”）。

在企业中的场景： 当两个部门的分类体系不一致——工艺科把“不锈钢螺栓”归为“紧固件”，采购科归为“五金件”——通过OWL，可以定义“不锈钢螺栓同时是紧固件和五金件”——这种跨域的概念统一正是OWL的核心价值。

不用OWL会怎样： 当AI执行跨域查询时，“紧固件”和“五金件”是两条互不了解的数据路径。AI不会知道这两个类在描述同一组物品。

5.4 SKOS：术语管理——筑牢语义基础

它做什么： OWL管的是类的关系。SKOS管的是术语——同义词、缩写、翻译、术语之间的关联。

在SKOS中，你可以定义：

术语: "螺栓" = 英文 "bolt" = 缩写 "BLT" = SAP字段 MATNR
术语: "代用料" = 正式名称 "采购替代物料" = 简称 "代用料"

在企业中的场景： 当一个新来的AI工程师读取ERP数据表，看到”MATNR”这个字段名时——他需要去翻SAP的手册才能知道这是”物料编号”。SKOS把这项知识保存在本体中，让任何调用它的系统（包括AI）都能自动获得完整的术语信息。

不用SKOS会怎样： 每个人都需要”现学”术语。一个AI Agent在处理代用料替换时，需要被明确告知”代用料 = 采购替代物料”——否则它会创建一条新的、不同的物料类型。

5.5 SWRL：规则语言——支撑AI自主决策

它做什么： 规则是本体的核心能力之一。SWRL是用来表达IF-THEN规则的W3C标准语言。

SWRL规则：

IF 物料.使用场景 == "户外" AND 物料.表面处理 == "发黑"
THEN 物料.不符合.使用要求—需要审查

在企业中的场景： 最常见的企业场景是**合规检查**。当BOM变更时，新的物料必须满足旧物料关联的所有质量标准、安全规范和行业规定——SWRL规则能在变更前自动检查所有相关约束，而不是等着质量部门在提交后才发现问题。

不用SWRL会怎样： 规则必须写在代码中——每次业务规则改变都需要重新开发和测试。而供应链政策的变化（比如某个国家的关税调整影响A/B类件的平衡）可能需要改动多行代码。

5.6 OWL-S：流程语言——定义AI可以执行的步骤

它做什么： RDF定义了”是什么”，OWL定义了”是什么关系”，SWRL定义了”什么是对的”——但AI还需要知道”怎么动”。这就是OWL-S的角色。

OWL-S描述的是**服务的流程**——包括前置条件（开始前必须满足什么）、影响效果（完成后改变了什么）、输入输出和步骤。在制造环境下，每个业务动作（发起变更请求、生成采购订单、推送通知）都需要一个明确的流程定义。

在企业中的场景： 当AI Agent决定”需要发起一个紧急采购”时，OWL-S告诉Agent：先查供应商可用性 → 创建采购申请 → 提交到审批系统 → 等待审批 → 如果审批通过，生成采购订单并通知WMS。

不用OWL-S会怎样： Agent知道”该做什么”，但不知道”怎么做”。它会生成一份采购建议，但无法推进后续步骤——这意味着一大部分的服务价值没有被转化为行动。

5.7 ODRL：权限语言——构建AI操作的安全边界

它做什么： ODRL描述的是“谁在什么条件下可以做什么操作”——这是AI Agent安全的关键。没有权限控制，AI可能会执行超出其授权范围的操作。

在企业中的场景： “AI Agent可以查询所有物料库存，但只能对非关键物料发起采购建议；对于关键物料，必须经人工审批后才能创建采购订单。”

ODRL在本体层操作，确保权限规则不依赖于某个具体系统的安全配置——而是由本体的独立语义来定义。

5.8 SPARQL：查询语言——让AI随心所欲地“问问题”

它做什么： 如果RDF是“写”，SPARQL就是“读”。它是W3C的RDF数据查询标准——相当于SQL之于关系数据库。

在企业中的场景：

```
# 查询：所有由C级供应商供应的、用于高温场景的关键零件
SELECT ?part ?supplier WHERE {
  ?part a ex:Bolt .
  ?part ex:scenario "高温" .
  ?part ex:criticality "关键" .
  ?part ex:supplied-by ?supplier .
  ?supplier ex:rating "C级" .
}
```

这个跨多个维度的查询（分类+场景+关键性+供应关系+供应商评级）在传统SQL中需要多次join，且无法表达“分类”（不是字段值，而是类层级）这个维度。SPARQL直接就能处理。

在企业中的场景： AI Agent通过SPARQL查询语义层——不需要知道底层数据的物理位置和格式，只需要用概念和关系来表达查询。

5.9 “+1”：目标与评估——让本体工程有方向

“7+1”的“+1”是一个元维度——**每一步都有评估点**，确保本体工程不是为了“做本体”而做本体。

评估的三个层级

表 5-1

层级	评估内容	指标示例
业务目标	本体解决了什么业务问题？	BOM变更影响分析从2小时缩短到30秒
模型质量	本体是否正确、完整、无冲突？	概念覆盖率达到X%，规则通过率Y%
AI效果	本体对AI性能的提升	推理准确率从X%提升到Y%

《数据空间探索与实践》中的评估维度强调：本体的建设必须围绕业务场景展开，每一步都有可测量的效果评估——不是为了“符合本体标准”而建，而是“解决了业务问题”。

5.10 本章小结

1. “7+1”语义体系涵盖从基础描述（RDF）到安全边界（ODRL）的本体全链路
2. 每个规范解决一个明确的问题——不需要全盘用上，按企业场景需求选择
3. 各规范之间互相衔接——RDF提供基础数据，OWL扩展语义，SWRL编码规则，OWL-S串联流程
4. “评估”维度确保本体工程不脱离业务目标



延伸阅读

- **W3C Semantic Web Standards** — 所有7种规范的官方文档
- 《数据空间探索与实践》第5.4节 — “7+1”语义规范框架的原始定义
- **SPARQL 1.1 Query Language** — W3C推荐标准，RDF数据的查询语言参考

第6章 本体建模工程化方法

“不玄乎。29句话，两个大模型，就能把一个业务域的概念体系看清楚。”

【决策者摘要】

本体建模最怕两件事：一是变成学术论文——花三个月建了一个完美模型，业务部门看不懂也不用；二是变成手工编码——让本体工程师手动写OWL，效率低、门槛高、积累慢。华为提出了一套工程化方法来解决这个问题：用”29句话”引导业务专家把隐性知识说出来，用大模型辅助生成本体模型并做校验，用可视化工具辅助专家审核——让本体建模从”少数人的技能”变成”团队的标准化流程”。最后通过”点线面”路径从小场景开始迭代推广。本章是全书最”可以上手”的一章。

6.1 本体建模的”工程化”在哪里

传统本体建模的主要问题是三高三低：门槛高（需要懂OWL的人）、沟通成本高（领域专家和本体工程师互相听不懂）、迭代成本高（改一个概念可能要改很多地方）——效率低、复用率低、业务参与率低。

工程化方法的”破局”做了三件事：

1. **引入AI做建模助手**——把最繁琐的形式化编码过程交给大模型，把人类的精力保留在概念定义和质量审核上
2. **标准化隐性知识的提取**——用”29句话”的结构化对话引导领域专家把脑子里的业务知识系统地讲出来
3. **“建用优复”生命周期管理**——确保本体模型不只是在某个人的电脑上，而是在团队中持续运营和迭代的资产

6.2 “29句话”：从业务专家到本体模型的预处理方法

“29句话”是华为的创新——不是一个固定的话术列表，而是一套结构化的概念提取引导框架。

工作原理

29句话覆盖了本体建模需要的四类核心信息，按照逻辑顺序引导领域专家逐条回答。大模型辅助提炼出核心句（约29句话就是在一个典型业务域中能完整描述概念体系的关键陈述数量）：

表 6-1

类型	示例提问	引导出的本体构件
身份类	“在这个业务中，最核心的东西有哪些？”	类
属性类	“每个东西有哪些重要特征？”	属性
关系类	“这些东西之间有什么关联？”	关系
规则类	“什么情况下会变？变了影响谁？”	约束和规则

一个简化的对话示例

以BOM变更场景为例：

本体工程师："在你们的BOM变更这个业务中，最核心的东西有哪些？"

工艺科专家："物料、BOM、供应商、质量标准、变更单。"

本体工程师："这些物料怎么分类？"

工艺科专家："按自制件和外购件分。外购件里又分标准件和电子元器件。"

本体工程师："一个物料跟它对应的BOM之间是什么关系？"

工艺科专家："一个物料是被BOM使用。一个BOM里的每个物料都有一个'位置'属性。"

本体工程师："什么情况下BOM变更需要审批？"

工艺科专家："如果是兼容替换——螺栓换螺栓——只要工艺科自己批准就行。但如果是非兼容替换，比如螺母换螺栓，就需要审批。"

对话中的每一句，都在为后续的建模打基础。过程结束后提取出引导定义的核语句，大模型可以直接从中抽取候选类、关系、属性——领域专家不需要懂OWL，只需要会说自己的业务。

6.3 AI辅助建模与专家审核

双大模型协同

华为的方法中，有两个大模型参与本体建模过程：

表 6-2

模型角色	任务
Model A（生成模型）	将”29句话”转化为结构化本体（生成OWL/类和关系/属性/规则）
Model B（校验模型）	审核Model A的输出：语法对不对、语义有没有歧义、逻辑是否一致

两个模型互相校验的优点：生成模型可能会”幻觉”（编造一个不存在的类或关系），而校验模型能检测到这种不一致。

可视化工具辅助专家审核

AI生成的本体模型不能直接投入使用——需要人类专家的最终审核。但不是让专家看OWL代码（领域专家看不懂），而是用可视化工具审核。

工具能把本体显示为一张图：圆形 = 类，连线 = 关系，框里的文字 = 属性。专家看图就能指出：”这个关系不对——螺栓不是’替代’紧固件，是’属于’紧固件”。

测试用例验证

最终一步是用场景测试来验证本体模型：

测试用例： "BOM变更： 物料A（螺栓M8×20碳钢） → 物料B（螺栓M8×20不锈钢）"

期望： 本体能自动判断：

- 1. 两个物料属于同一类（螺栓） → 兼容替换
- 2. 新物料的材料属性符合户外场景的约束 → 无需额外检查
- 3. 变更需要通知的部门列表 → 采购科、质量科

结果：  三个判断全部通过 → 模型合格

6.4 “建用优复”生命周期管理

本体模型不是一次性产物，需要全生命周期管理。华为提出的“建用优复”四个阶段：

表 6-3

阶段	活动	交付物
建	AI生成本体 + 专家审核 + 场景验证	初版本体模型
用	加载到语义数据库，接入AI Agent或查询系统	可运作的语义层
优	根据使用反馈补充概念、修正关系、优化规则	优化版本体模型
复	跨场景复用——一个业务域的本体可以复用到其他场景	多场景共享的本体库

6.5 “点线面”迭代式实施路径

本体建模不需要一开始就覆盖整个企业。华为提出了“点线面”路径：

点（1-2个月）：选一个“小切口”场景

- 选一个业务痛点最明显、数据范围最小、影响面最可控的场景
- 只建5-10个核心类，10-20条关键关系
- 目标：验证本体+AI在解决一个具体问题上比传统方法更好

线（3-6个月）：纵向整合一个领域

- 在一个完整的业务条线上拉通多个场景的语义
- 把所有相关场景的本体模型合并成一个统一的领域本体
- 目标：一个领域的“语义一致”

面（6-12个月）：跨领域贯通

- 在不同领域之间（如采购+生产+质量）建立跨域语义连接
- 解决跨域的语义冲突（比如采购科和工艺科对“重要物料”的不同定义）
- 目标：全企业的“语义互联”

6.6 本章小结

1. 本体建模工程化做了三件事：AI辅助建模、标准化知识提取、“建用优复”全周期管理

2. “29句话”是一套结构化的对话框架，帮助领域专家系统输出隐性知识
3. 双大模型协同——一个生成、一个校验——显著降低了建模门槛和错误率
4. “点线面”迭代路径避免了全盘建模的陷阱，从小场景起步，逐步扩展



延伸阅读

- 《数据空间探索与实践》第6章 — “29句话”方法论和AI辅助本体建模的完整流程
- **Ontology Development 101** (Noy & McGuinness, Stanford) — 经典的本体建模七步法
- **Protégé Ontology Editor** — 最成熟的开源本体建模工具，配合可视化插件进行专家审核

第7章 本体增强型AI Agent

“本体让Agent从‘调接口’变成‘理解业务’。”

【决策者摘要】

一个AI Agent要能真正在企业里干活，不是靠API调用就能做到的。它需要的是一套“认知框架”——能理解用户意图、能找到对应的业务概念、能基于规则推理、能转化为具体操作。本体正是提供这套认知框架的工具。本章从Agent的工作流出发——意图对齐→本体检索→增强推理→决策到行动，逐一展开每一步中本体的角色。然后给出Agent构建的完整框架和实施路径。

7.1 Agent的核心工作流：意图→语义→推理→行动

一个企业级AI Agent的运行流程可以分解为四个环节：

用户输入："0102003581这个料影响哪些工单？"

↓

【意图对齐】Agent理解用户要什么

↓

【语义检索】从本体层拉出相关概念和数据

↓

【增强推理】基于事理模型做推理

↓

【行动执行】生成可操作的结果

四个环节中，每个都离不开本体。我们逐一来看。

7.2 意图对齐：把”人话”翻译成”本体查询”

用户对Agent说的一句话——“这个物料影响哪些工单”——需要被拆解为多个结构化的查询。

Agent需要理解： - “这个物料” → 哪个实体ID → 本体中的哪个实例 - “影响” → 什么关系 → used-in → belongs-to → 推导出一组受影响的实体 - “哪些工单” → 什么类型的实例 → 生产工单类的实例

这就是**意图对齐**——把自然语言映射到本体的类和关系上。

传统方式下，Agent只能做关键词匹配——“工单”“物料”→查ERP的工单表，没有理解”影响”这个语义。本体驱动的Agent不同——它能把”影响”映射到关系链上：物料 → used-in → BOM节点 → belongs-to → 产品 → corresponds-to → 工单。

《数据空间探索与实践》中的做法是：为每个常见的业务意图维护一个”意图-本体映射表”——当用户说”影响分析”时，Agent自动选择对应的关系链进行查询。

7.3 本体检索与调用

意图确定后，第二步是从本体层检索相关的概念和数据。

检索方式

表 7-1

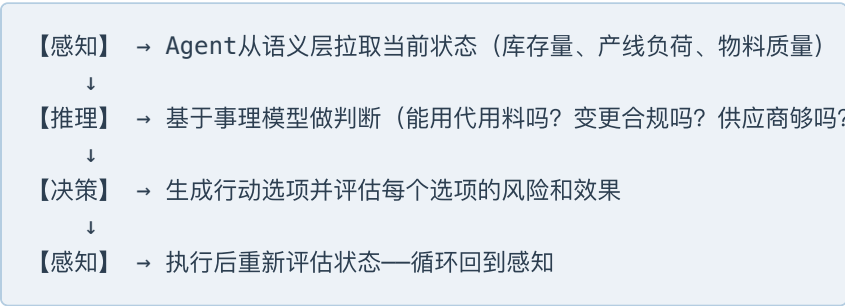
方式	描述	适用场景
SPARQL查询	直接查询RDF语义数据库	需要精确查询的场景
本体导航	从概念到实例逐层下钻	需要理解的场景
关系链推导	从实体出发，沿着关系链扩散	影响分析和追溯

本体嵌入AI模型可以有不同方式： – **直接嵌入** — 把本体三元组嵌入到向量数据库中，与RAG系统集成 – **运行时检索** — Agent在执行时实时查询语义数据库，获取当前的准确数据

7.4 本体驱动增强推理

感知-推理-决策循环

第4章介绍了三元模型——事实、事理、行动。在Agent的推理过程中，这三个层次形成循环：



推理的三个深度

表 7-2

深度	描述	示例
直接判断	基于单条规则的简单推理	“这个物料过期了吗？”
链式推理	多条规则和关系链的联合推理	“变更这个物料会影响哪些工单和供应商？”
综合决策	多维度评估的复杂决策	“三个供应商中选哪个？综合考虑价格、质量和交付”

本体的核心价值在**链式推理**和**综合决策**两个层面——这正是LLM自身无法可靠执行的。LLM能给出建议，但不能保证每个推理步骤的逻辑正确。本体的规则保证了推理链的确定性。

7.5 决策到行动的转化

AI Agent不能只是”想得好”，必须能”做得到”。本章借鉴《数据空间探索与实践》的三类行动转化模式。

指令直驱型：AI直接执行

适用场景：影响小、规则明确、已验证准确率高的操作。

示例：Agent判断”物料0102003581当前库存2800件，工单需求480件——库存足够”→ 直接更新工单物料分配状态 → 无需人工确认。

前提条件：这个操作经过了充分的测试和验证，并且其影响范围有限（如果出错，局部可逆转）。

人工介入型：AI推荐→人确认

适用场景：影响中等、需要业务判断的操作。

示例：Agent判断”物料0102003581可由0102003582替代，建议执行替代”→ 生成替代建议推送给工艺科 → 工艺科确认后，Agent代执行后续步骤。

关键是：**AI推荐有推理链可追溯**——工艺科的人能看到Agent为什么建议这个替代，而不是收到一个黑盒决策。

业务协同执行型：AI生成方案→多部门审核→执行

适用场景：影响大、跨部门的决策。

示例：一个大规模BOM变更影响三个产品线 and 五个部门。
Agent生成整套行动方案→推送到各相关部门→各部门给出反馈→方案汇总修订→最终执行。

7.6 Agent构建框架

一个完整的企业本体增强型AI Agent需要以下几个组件：

表 7-3

组件	功能	技术实现
事实关联层	实时获取和关联企业数据	语义数据库+数据同步管道
事理关联层	查询和调用业务规则	规则引擎+SWRL/SHACL检验
行动关联层	抽象到具体的操作转化	OWL-S+API适配器
本体服务层	提供语义检索和推理能力	SPARQL端点+推理引擎
大模型引擎	意图理解和自然语言交互	LLM+意图对齐映射
运行框架	多轮对话、任务编排、异常处理	Agent框架（如AutoGen/CrewAI）

7.7 本章小结

- 1. Agent的核心 workflow 是”意图→语义→推理→行动”，本体在每个环节都有明确角色
- 2. 本体驱动的推理是确定性的——每一步推理都有规则支撑，可追溯到规则源
- 3. 三种行动转化模式对应不同的风险级别和信任程度

4. Agent构建需要六大组件协同，不是”一个LLM + 一个数据库”



延伸阅读

- 《数据空间探索与实践》第7章 — 本体增强型AI Agent的完整工程实现
- SWRL 规范 (W3C) — 语义Web规则语言的技术标准
- OWL-S — 本体驱动的服务建模标准

第8章 制造业落地案例

| 三个行业，三种场景，一条主线

—— 本体怎么让AI读懂企业数据。"

【决策者摘要】

本章通过三个真实的制造业案例，从头到尾展示本体驱动的AI Agent如何落地。案例A是精密机床企业的故障根因分析——设备+参数+缺陷三元本体让Agent从”感知异常”到”定位根因”只需要几分钟。案例B是汽车零部件企业的BOM变更影响分析——物料+BOM+供应商+质量标准四域本体让变更影响分析从2小时缩短到30秒。案例C是电子制造企业的跨系统物料追溯——ERP+MES+WMS多域本体让跨系统查询从人肉关联变成一条SPARQL。三个案例独立可读，可跳看。

案例A 精密机床故障根因分析

企业背景

一家年产值约12亿的精密数控机床制造商，两条总装线、六条零部件加工线。产品是高端多轴联动数控机床，对零部件精度和装配一致性要求极高。

痛点

设备故障后，诊断流程依赖经验丰富的设备工程师逐台排查。平均每次故障定位耗时两小时以上，严重时会造成整条加工线停工四到六小时。企业尝试过部署设备监控系统——数据是有

了，但”有数据”和”能诊断”之间的距离仍然很大。

核心问题：设备监控系统能告诉你”振动超限”，但不能帮你理解”振动为什么会超限”。振动超限可能是因为刀具磨损、轴承松动、润滑不足或进给参数异常——每种根因对应不同的维修方案。看监控屏幕的人需要自己判断。

建模——设备+工艺参数+质量缺陷三元本体

企业跟外部本体工程师合作，建了一个三层的故障根因本体：

表 8-1

本体类	核心概念	来源系统
设备	机床型号、主轴参数、轴承类型、维护历史	设备监控系统、EAM
工艺参数	加工转速、进给速度、切削深度、冷却液流量	MES、CNC控制器
质量缺陷	尺寸超差、表面粗糙度超标、刀纹异常	QMS、三坐标测量机

关键关系链：

设备状态 - abnormal → 振动超限（感知）
振动超限 - has-cause → 刀具磨损
刀具磨损 - detected-by → 主轴负荷上升（参数变化）
主轴负荷上升 - correlates → 表面粗糙度超标（质量缺陷）
↓
根因推理：刀具磨损 → 需要更换刀具

关键规则（事理层）：

1. IF 振动幅度 > 0.05mm AND 工艺阶段 = 精加工 → 工件表面粗糙度将超标 → 需要停机
2. IF 连续3次加工后的工件都出现尺寸超差 → 根因不是刀具问题 → 检查设备校准状态
3. IF 振动异常发生在换刀后 → 根因可能是刀具安装不当或刀具规格不符

AI Agent的运行

感知阶段：Agent实时接收设备振动数据和工件质量检测数据。当振动读数连续30秒超过阈值、且加工后的工件出现表面粗糙度超标时，Agent触发异常感知信号。

推理阶段：Agent在事理层执行以下推理链：

1. 振动在哪个工艺阶段发生的？ → 精加工阶段
2. 振动发生前更换过刀具吗？ → 两个小时前更换过
3. 检查换刀记录：新刀具的型号和材质 → 型号正确但材质与主轴不匹配

4. 查询刀具本体中定义的使用规范：“该主轴最高转速12000rpm，换刀后必须经30分钟磨合运行”
5. 检查换刀记录：换刀后没有执行磨合运行 → 异常根因

行动阶段：Agent生成维修建议 → 推送给设备工程师：

“主轴振动异常的根因分析结果：可能原因为换刀后未执行磨合运行协议。建议：(1)立即停机，执行磨合运行；(2)检查工件表面质量；(3)如仍有异常，更换刀具并重新磨合运行。”

效果

- 故障定位从平均2小时 → 15分钟（降低87.5%）
- 误判率从人工诊断的约30% → 系统诊断的约8%
- 设备工程师的日常诊断工作量减少约70%，可以把精力集中在复杂和非常规故障上

案例B 汽车零部件BOM变更影响分析

企业背景

一家年产值约28亿的汽车变速箱总成制造商，30多条产线，5000+个装配物料。产品给多家整车厂供货，BOM频繁变更——每月平均收到30–40个正式变更请求。

痛点

一次BOM变更需要协调五个部门——工艺、质量、采购、生产计划、物流——每个部门需要查阅不同的系统，确认物料可替代性和合规性，评估对产线、库存、供应商的影响。整个过程平均耗时2小时，涉及3个系统的人工查询。

变更的准确率全靠工艺工程师的经验和责任心。一旦有一个部门没能被通知到或忽略了影响，下游就会出现停工待料。

建模——物料+BOM+供应商+质量标准四域本体

表 8-2

本体类	核心概念	来源系统
物料	物料号、描述、分类、功能角色、可替代性	ERP+PLM
BOM	节点结构、位置号、用量、版本、有效日期	ERP+PLM
供应商	供应商编码、供货范围、资历、评级	ERP+SRM
质量标准	检验规范、公差要求、合规认证、测试方法	QMS+PLM

关键关系链：

物料A - used-in → BOM节点B
BOM节点B - has-function → "密封连接"
物料A - substitutable-by → 物料B
物料A - has-standard → 质量标准Q
质量标准Q - verified-for → 物料B (√) 或 (×)
物料A - supplied-by → 供应商S1
物料B - supplied-by → 供应商S2
BOM节点B - belongs-to → 产品P
产品P - manufactured-at → 产线L

关键规则（事理层）：

1. IF 变更类型 = 兼容替换 AND 功能角色不变 → 工艺科审批即可
2. IF 变更类型 = 非兼容替换 → 技术中心必须参与审批
3. IF 变更后的物料供应商评级 = C级 → 需要质量科额外审核
4. IF 新物料未经所关联的质量标准认证 → 变更搁置 → 等待供应商提供书面认证
5. IF 新物料在产线的设备兼容性未经验证 → 需要产线试运行

AI Agent的运行

感知阶段：Agent接收一个BOM变更请求：物料A（螺栓M8×20碳钢）→ 物料B（螺栓M8×20不锈钢），由工艺科发起，使用场景=户外。

推理阶段：Agent在事理层执行完整的合规推理：

1. 物料A和B是否属于同一类？→ 同属螺栓类 → 兼容替换

2. 新物料的材料（不锈钢）是否符合户外场景的规则？→ 是（不锈钢满足户外表面处理要求）
3. 新物料是否满足物料A关联的所有质量标准？→ 3个标准全部满足
4. 变更会涉及哪些产品？→ 3个产品
5. 这些产品的产线是否需要试运行来验证变更？→ 需要（规则5）
6. 哪个供应商可提供新物料？→ 供应商S2
7. 供应商S2评级如何？→ B级 → 不需要额外质量审核

行动阶段： Agent生成变更影响报告 → 推送给5个相关部门：

“BOM变更影响分析（物料A→物料B），变更类型：兼容替换，结果： - 受影响的3个产品：发动机总成A、变速箱总成B、缸盖总成C - 部需要通知的部门：工艺科（审批）、采购科（供应商确认）、生产计划（产线试运行） - 供应商S2已确认可提供新物料B - 3项质量标准全部满足 ✓ - 需要产线试运行”

效果

- BOM变更影响分析从平均2小时 → 30秒（降低99.6%）
- 变更通知准确率从约85% → 97%
- 产线停线次数由每月平均2.5次降至1次以下

案例C 电子制造跨系统物料追溯

企业背景

SMT贴装代工厂，年产值约8亿，客户是消费电子和工业电子品牌。物料复杂度高——每个PCBA可能包含100–300个独立的电子元器件。

痛点

物料的”身份漂移”是最头疼的问题：

- 研发用供应商原始编码从PLM指定物料
- 采购用ERP的物料号做管理和下单
- 生产用MES的物料版本号做追溯
- 仓库用WMS的库位编码做存储

不同系统对”同一个物料”有不同编码和识别方式。当一个物料出现质量问题时，追溯”这个物料被用在了哪些客户的哪批产品中”需要跨三个系统手动关联。财务和质量科每月花约40–50人时做物料数据的手工对齐。

建模——物料+版本+供应商+库位多域本体

表 8-3

本体类	核心概念	来源系统
物料	企业物料号、供应商料号、客户料号、封装类型、规格参数	ERP+PLM+供应商清单
物料版本	版本号、变更原因、生效日期、关联BOM版本	ERP+PLM
供应商	供应商编码、供应料号、评级、替代物料	ERP+SRM
库位	库位编码、存储类型、温湿控制、货架位置	WMS

关键关系链：

供应商料号(0603-R103) – maps-to → 企业物料号(030201005)

企业物料号(030201005) – has-version → V2

V2 – corresponds-to → ERP物料记录#D39K

ERP物料记录#D39K – maps-to → MES物料版本(030201005-V2)

MES物料版本(030201005-V2) – stored-at → 库位(08A-03-01-05)

↓

质量追溯查询: "物料030201005质量异常" →

找出它当前在哪个库位 → 哪批采购 → 用在了哪些产品BOM中 →

这些BOM对应哪些客户订单

效果

- 单次物料追溯从4个系统跳到3个手动步骤缩减为1个统一的SPARQL查询

- 质量科每月的物料数据手工对齐从40–50人时 → 约5人时
- 物料数据置信度大幅提升——错误从”靠人发现”变成了”系统自动检测”（如不一致的库位编码）

案例共性总结

表 8-4

维度	案例A：机床故障	案例B：BOM变更	案例C：物料追溯
本体类型	三元本体	四域本体	多域本体
核心关系数	~20条	~40条	~25条
AI作用	异常感知→根因推理	变更推导→合规检查	跨系统查询→替代推荐
关键收益	故障定位 2h→15min	影响分析 2h→30s	手工对齐40人时→5人时



延伸阅读

- 《数据空间探索与实践》第8.2节 — 本体与AI协同的五种核心应用模式
- 《本体驱动：制造业主数据语义化实战》第5–6章 — 物料/BOM/工艺路线的本体建模方法

- Edwards et al., “Agentic AI in Engineering and Manufacturing” (arXiv:2604.09633, 2026) — 30+家企业AI在制造业落地实证

第9章 实施路线图与组织保障

“不用推翻重来。选一个小切口，建一套本体，跑通一个场景。”

【决策者摘要】

本章回答的是CEO/CIO最先会问的问题：“本体这个项目，怎么起步？”企业的本体建设项目不是大型IT改造项目。它更像一个产品迭代过程——选一个小场景，跑通语义化流程，验证价值后再推广。本章给出三阶段路线图（诊断→试点→推广），每阶段的交付物、投入预估和检查点；组织层面需要的三种角色（领域专家、本体工程师、数据架构师）和协作节奏；以及企业本体项目的五个典型失败模式。读完可以拿着这份路线图去评估自己企业现阶段适合从哪个力度开始。

9.1 三阶段路线图

第一阶段：诊断与试点（1-3个月）

目标： 选一个能快速验证价值的场景，走完整条语义化流程。

选择试点场景的原则：

- 业务痛点明显且可量化——“我们现在BOM变更每次花2小时人工分析”比“我们的数据质量不太好”更适合试点
- 数据范围小——只涉及1-2个源系统，5-10个核心类
- 影响面可控——试点结果对业务的影响可接受，即使失败也不造成严重损害
- 数据质量及格——不需要完美，但也不能在完全没有清洗的数据上建模

第一阶段交付物： – 试点场景的本体模型（≤10个类，≤20种关系） – ERP源数据的映射配置 – 一个简单的Web查询工具或AI Agent原型 – 试点总结报告（关键指标对比、收益预估、推广建议）

投入预估： 2-3人×3个月，技术栈用Python脚本+开源语义数据库（如Apache Jena Fuseki），约20-40万。

第二阶段：核心推广（3-6个月）

目标： 扩展到完整的业务域，覆盖多个场景和更多数据源。

第二阶段交付物： – 完整领域本体模型（一个业务域的所有核心概念） – 多源数据映射管理（ERP+PLM+MES+QMS多个系统的数据源并行集成） – 基本本体治理流程（版本管理、冲突检测、变更审批） – 服务化部署的本体语义数据平台

投入预估： 3-4人×6个月，部署专用语义数据库+PaaS平台，约100-200万。

第三阶段：AI融合与长效治理（6-12个月）

目标： 让语义化后的数据服务于AI应用，建立长效治理机制。

第三阶段交付物： – AI Agent接入语义层（本体增强型的排产/质检/根因分析Agent） – 跨系统语义查询平台 – 知识图谱与业务本体的结合 – 本体治理组织常态化运行（季度评审会、年度路线规划）

9.2 组织：三种角色

表 9-1

角色	职责	来源	时间投入
领域专家	定义业务概念、验证模型准确性、引导场景分析	业务部门（工艺/质量/采购）	每周2-3天
本体工程师	建模、配置映射、维护技术平台	IT部门或外部顾问	全职
数据架构师	治理策略、版本管理、全企业协同	IT部门	每周1-2天

协作节奏

表 9-2

节奏	活动	参与者
周度	领域专家提出本周场景 → 本体工程师建模 → 领域专家审核	核心团队
月度	评审新增的类和关系，检查有无跨部门语义冲突	核心团队+部门负责人
季度	路线规划、资源评估、向CIO报告	核心团队+CIO/CTO

9.3 五个常见失败模式

表 9-3

失败模式	典型表现	预防措施
一上来铺全量	试图一次性覆盖物料 +BOM+工艺+质量+供应商 +财务	坚持”点线面”路径：选一个小切口，3个月不出成果就是失败
纯IT驱动无业务参与	本体模型在技术层面无懈可击，业务部门表示”这不是我们的业务逻辑”	第一天就拉领域专家进场，不要假设IT能替代业务专家定义概念
建模脱离数据质量	本体建好了，但源数据质量太差，映射出来的实例全是垃圾	先盘数据质量。如果连基本的编码唯一性都做不到，先推动数据治理补齐基础
工具过热	第一阶段就部署了完整的中台+大数据平台+多套本体编辑器	第一阶段用Python脚本+开源语义数据库就足够。性能瓶颈出来后再考虑升级工具栈
无退出机制	项目推进但未产生明确价值，又不敢停	每个阶段设go/no-go决策点。如果三个月未达到预定指标，果断暂停并诊断问题

9.4 本章小结

1. 本体建设不是大型IT改造项目，而是一个产品迭代过程——从小场景做起，逐步推广
2. 三阶段路线（诊断→试点→推广）各有明确的交付物和检查点
3. 三种角色的组合是关键——缺人的风险远大于缺技术的风险
4. 五个常见陷阱的共性：急于求成，脱离实际



延伸阅读

- 《数据空间探索与实践》第8.3节 — 企业AI落地的穿刺式验证策略
- 《本体驱动：制造业主数据语义化实战》第8章 — ERP系统语义重构的三阶段路线

第10章 展望：本体与大模型的融合演进

| LLM不是本体的替代品

—— 它是本体的接口。"

【决策者摘要】

大语言模型能不能替代本体建模？结论——不能。LLM擅长语言生成但不擅长确定性推理。两者融合才是企业AI的正途。短期的方向是本体增强RAG（让LLM的检索建立在结构化知识上），中期的方向是LLM辅助本体工程（用大模型加速本体模型的创建和维护），长期的图景是混合推理——确定性判断交给本体，模糊判断交给LLM。最后讨论企业的“三重世界”模型（物理、逻辑、智能）的协同演化路径。

10.1 LLM不能替代本体的三个原因

原因一：LLM没有企业私有知识

LLM的“知识”来自训练数据——公开的互联网、论文、代码。你的企业中的“0102003581”代表什么、A/B/C件分类的规则是什么、Q标准中关于温度范围的约束是什么——这些具体的、私有的、依赖上下文的信息，LLM根本不知道。

原因二：LLM的推理不能保证正确

LLM对问题的回答是基于统计模式匹配的，不是基于确定的规则和事实的。对一个企业级决策——比如“这个物料能不能换给这个产品”——LLM能给一个猜测，但这个猜测可能对也可能错。

对于合规、安全、质量相关的判断，“可能对”是不够的。

原因三：LLM不能独立维护企业知识

企业数据每天都在变——新物料、新供应商、新的质量规定。LLM的更新成本极高（全量重训或重新索引大型语料库），而本体的更新是增量式的——改一条规则、加一个新实例，不需重训AI。

正确的定位：LLM是企业AI的”自然语言接口”，本体是企业AI的”知识基座”。各司其职，互补不足。

10.2 三条融合演进路径

短期（1-2年）：本体增强的RAG

这是目前最成熟的融合模式。传统RAG的弱点是：文档被切块后，语义关系丢失了。本体增强RAG的做法是：检索步骤分两层进行：

用户查询

↓

第一层：本体驱动的结构化检索 → 获取确定的、可追溯的知识

第二层：向量检索 → 从文档中补充非结构化的上下文

↓

LLM综合两层信息生成回答

中期（2–3年）：大模型辅助本体工程

本体建模目前大部分工作还是人工的。大模型可以在以下环节辅助：

- 自动从企业文档（工艺规程、质量标准、审批流程）中识别候选的类和关系
- 自动建议ERP字段到本体属性的映射（根据字段名和值的分布推断）
- 自动检测多个本体模型之间的语义冲突

长期（3–5年）：混合推理

当LLM的能力进一步发展后，企业智能决策可以采用”混合推理”模式：

- 本体推理器处理**确定性判断**——“这个物料符合质量标准Q吗？”→ 有明确的规则支撑
- LLM处理**模糊判断**——“这个供应商的长期风险趋势如何？”→ 需要综合多维度的模糊信息
- 两种推理各司其职，结果合并后给出综合建议

10.3 企业的”三重世界”协同

《数据空间探索与实践》提出了一个三重世界模型——物理世界（真实的工厂、设备、物料）、逻辑世界（本体、规则、知识）、智能世界（AI、Agent、大模型）。本体的角色是连接物理世

界和智能世界的”桥梁”——把真实的企业运行状态翻译成AI能理解的知识结构。

三重世界的协同演进意味着：

- **本体必须不断更新**——物理世界（新的产品、新的产线、新的规则）在不断变化，本体也必须随之演进
- **AI的能力会不断提升**——未来的大模型将更好地辅助本体建模和调用
- **人的角色转变**——不再需要手动查询多个系统来做出业务判断，而是审核AI的判断是否符合业务预期，并对高价值的判断进行审批

10.4 结语

这本书是从第一句话开始的——第1章讲了一个翻车的AI排产Agent。它翻了，因为数据只是数据，没有含义。

本书的核心主张很简单：在AI能发挥其真正的价值之前，数据要先能”被理解”。这本体的任务。它既古老（从古希腊哲学来），又新鲜（在AI时代获得新的生命力）。

如果你读到了这里，大概率是企业的数据管理负责人或决策者。希望你已经有了一个清晰的图景——本体不是你还需要评估要不要上的工具；它是你的数据架构中缺失的那层”含义”。从一

个小场景开始，选一个痛点最明显的业务场景，建一个只有几个类的小本体，跑通一次“感知-推理-行动”的闭环。这，就是起手式。



延伸阅读

- 《数据空间探索与实践》第9-10章 — 碳硅协同的治理理念、三重世界模型
- Pan et al., “Unifying Large Language Models and Knowledge Graphs” (IEEE, 2024) — LLM与知识图谱的三种融合模式
- 《本体驱动：制造业主数据语义化实战》第9章 — AI Agent与语义化企业数据



树懒老K（拙一）

30年企业服务经验 · 专注AI智能体与组织变革



个人网站



个人微信



公众号

慢一点，深一度