ONTOLOGY MODELING CHECKLIST

本体建模自检清单

六步自查 · 每步的卡点问题 · 分段抽取的坑 · 立项判断四条标准

心晴九天 · 小红书 howru999

小红书 · 关注更新

Agent 助理 · 扫码交流

本体建模自检清单

> 用法：按六步顺序自查。每一步先回答"卡点问题"，答不上来就别进下一步。
>
> 最后四条是立项判断标准，四条同时满足才建议启动。

第一步

定边界

要做的　说清楚这个本体管什么、不管什么。

卡点　大模型无法自主判断本体该有多大。你不给边界，它会一直往外扩。

自查

- [ ] 能用一句话说出本体的范围吗（例如"只管进项抵扣，不管出口退税"）
- [ ] 这个范围是业务方定的，还是技术方猜的
- [ ] 范围外的东西，有明确的"不做"清单吗

不合格信号：范围描述里出现"以及相关的""等等"。

第二步

抽实体

要做的　从文档里把名词找出来。

卡点　整个流程里绝大多数错误在这一步产生，而且越往后越难发现。

自查

- [ ] 同一个东西的不同叫法，有没有归成一个（"小规模纳税人"和"小微企业"是两个概念， 但"进项税"和"进项税额"是一个）
- [ ] 业务方看过这份实体清单吗，有没有他们认为重要但清单里没有的
- [ ] 清单里有没有其实是数值、日期、状态的东西被误当成实体

不合格信号：实体清单是模型跑出来直接用的，没人逐条看过。

第三步

定类型

要做的　判断每个词是什么类型的东西。

卡点　类和个体的混淆。"小规模纳税人"是一个类（一群纳税人）， 不是某个具体企业。搞错了，这个节点下面就再也长不出东西，规则也挂不上去。

自查

- [ ] 逐条问："这个词指的是一群东西，还是一个东西"
- [ ] 是类的，能举出至少两个属于它的具体例子吗
- [ ] 是个体的，它属于哪个类，说得出来吗

不合格信号：某个节点既没有子类也没有实例，且不被任何属性引用。

第四步

建层级

要做的　搞清楚谁是谁的子类。

卡点　方向会反，而且"子类"的定义会被放得很松。方向反了不报错， 只是把规则往错的方向传播。

自查

- [ ] 每一对父子关系，念一遍"每一个【子】都是一种【父】"，念得通吗
- [ ] 有没有把"部分-整体"当成了"是一种"（发票是发票台账的一部分，不是一种台账）
- [ ] 有没有把"相关"当成了"是一种"（销项税额和进项税额相关，但不是父子）

不合格信号：两个概念经常一起出现，就被判成了父子关系。

第五步

抽非分类关系

要做的　抵扣、视同销售、纳税义务发生时间这类业务关系。

卡点　公认最难的一步。层级关系在词形里有痕迹，业务关系没有—— 它不在词里，在制度文件里。

自查

- [ ] 每条关系，能指出它的依据在哪份文件的哪一条吗
- [ ] 关系两端的类型限制写了吗（谁可以抵扣，抵扣什么）
- [ ] 有没有关系是"看起来合理"但找不到依据的

不合格信号：关系是靠常识推出来的，没有制度依据。

实测数据：同一个模型同一套题，只靠通用知识做非分类关系判定准确率 60%， 把制度原文附上去之后是 100%。差别不在模型，在有没有给材料。

第六步

验证

要做的　至少过三道——语法、逻辑一致性、结构缺陷。

卡点　绝大多数方案没有这一步，这正是它们停在原型阶段的原因。

自查

四个实测提醒：

- [ ] 语法：文件能被标准工具解析吗
- [ ] 逻辑：有没有跑推理机，有没有自相矛盾的公理
- [ ] 结构：有没有孤立节点、缺少定义域值域的属性、多个概念共用一个标识
- [ ] 推理机报错时，能定位到具体是哪个概念出问题吗

1. **用两个推理机交叉验证。** 实测同一个本体，HermiT 判一致、Pellet 判不一致， 原因是对 langString 值域这种写法的处理严格程度不同（本体本身没错）。 只信一个，就会去修一个本来没错的东西。
2. **报错回灌让模型自己修的时候，明确禁止它删规则。** 否则它会用"删掉内容"的方式让报错消失——实测出现过规则从 15 条被删到 9 条。 做法：把当前规则数和例外数写进修复指令，要求修完不得少于这个数。
3. **本体不完整的时候，不要纠缠一致性报错。** 实测：抽了一部分内容时有一条一致性错误怎么都消不掉， 逐条剥离 45 条值域声明、45 个属性都没定位到； 等把剩下的范围补齐、重跑一遍，这条错误自己没了，一轮就收敛。 缺失的上下文会让剩下的公理无法自圆其说。**先补齐范围，再看还剩什么。**
4. **修复要用补丁，不要整体重写。** 本体上了几百条三元组之后，让模型重新输出全文一定会被截断， 连正确的部分一起丢掉。改成只输出"要删哪几条、要加哪几条"，在图层面应用。

不合格信号：交付物里没有验证报告。

分段抽取的三个坑（大领域本体必踩）

大领域的本体没法一次生成完，必须分段。实测踩到的：

- [ ] 输出被截断时，不要加大输出上限 加大上限会触发别的限制（超长请求被拒），反而更糟。 有效做法是反过来：裁剪输入（只喂相关章节）、拆分目标（一段再拆两半）。
- [ ] 空产物必须当错误处理 输出被截断 → 提取到空串 → 空本体自然零报错 → 验证器误判"通过"。 实测踩过一次。验证器要显式检查"有没有类"。
- [ ] 截断的产物可以救 按空行做块级回退，从末尾一块块往回退到能完整解析的位置。实测救回 306 条里的 300，只丢最后一条不完整的规则。

三样最容易缺的东西

跑完一圈开源方案，缺的都是同样三样。自查你的方案有没有：

- [ ] 例外 规则覆盖八成情况，钱在剩下两成里。 举例：法条写"购进并直接用于消费的餐饮服务不得抵扣"， 关键是"直接用于消费"——转售是可以抵扣的。 这个限定条件如果没被建成显式的例外，系统会给出错误结论。
- [ ] 动作 只有实体、关系、度量的图是只读的。要改变业务，动作必须是一等公民。 而且每个动作要能授权、留痕、撤回。
- [ ] 验收标准 每条规则靠什么判定被正确执行了？ 论文用 F1 比对"标准答案本体"，但企业没有标准答案。 口径要事先跟业务方约定：比人工基线、看专家采纳率、还是算处理周期。

立项判断：四条标准

四条同时满足，建议启动；缺任何一条，先补齐再说。

- [ ] 1 高频重复，且直接影响收入、成本、履约或风险 一年发生三次的事，不要上系统。
- [ ] 2 输入材料可数字化，且能连到必要的数据与系统 材料在别人手里、系统连不上，本体建得再好也跑不起来。
- [ ] 3 有愿意共同定义规则和例外的专家 Owner，且有明确经济买方 这两个可以不是同一个人，但都必须存在。 没有专家，例外没人给；没有买方，做完了没人签字。
- [ ] 4 结果可对人工基线，验收口径双方事先能约定 事后再谈标准，一定谈不成。