

观点与分析

编辑日期: 2026年11月1日

昨天发布的通告，活在去年的模型

> 一句话可以正确，链接可以官方，原文可以一字不差，整份答案却仍然错误。问题不在系统说了什么，而在它以为自己谈论的是哪一天。

从九月寄往二月的信

2025 年 9 月 24 日，沙特中央银行发布第 472021766 号通告，更新场外金融衍生品交易的登记、披露与风险缓释要求。其中采用统一产品标识符（Unique Product Identifier, UPI），以便在报告记录中一致标识金融工具。通告明确引用第 42056371 号先前要求，并规定新要求取而代之。全部内容均可在央行规则库核验。1

关键一行并不描述产品，而把 2026 年 2 月 1 日定为新要求开始适用的日期。主管机关九月发布的文本在年底前已经公开，却不会仅因进入档案就自动成为所有十月事项的适用规则。九月诞生的文件未必九月生效；把“最新文件”等同于“现行依据”的系统会在这里犯错。12

详细要求更复杂：生效后，新交易没有国际证券识别码（ISIN）或 UPI（视可用情况）便不能进入登记披露中心。生效前已执行的交易，则要求银行通过回补录入（Backloading）更新既有组合数据，截止日为 2026 年 12 月 31 日。旧交易在新规则生效后承担新的报告完善义务；这不是把过去追认成违规，而是按指引把旧记录纳入新披露要求。3

若合规人员问检索系统“这份报告该填哪个标识符”，助手可以准确引用新要求并附官方链接；但若不先问报告日期、合同执行日期，以及这是新交易还是既有组合回补，它就把三个问题压成一个漂亮却无法用于运营的答案。

本文不虚构违规银行、失业员工或真实罚款。可核验事实是文本与日期；助手对话只是分析练习。它已经足以揭示风险：法律文本不是静态信息，适用条件决定答案何时才成立。

忠实引用也可能是一种错误

检索增强生成（Retrieval-Augmented Generation, RAG）让人学会检查：系统是否找到正确文件，回答是否得到片段支持，来源是否没有捏造。这些都是必要问题，却暗含一个不完整假设：相关文本被找到，就足以开始判断其是否适用。

定义稳定医学术语时，最新版指南可能是最佳起点；但问“这项披露义务在交易执行时是否存在”，最新不一定最佳，旧版也不一定错误。合适版本必须连接规则、事实、日期、主体、类别与例外。正确答案依赖规范权威的排序，而不只是按最后修改时间排列文件。

系统可能检索到语言高度相似的多个文本：一个规定期限，另一个修改期限；一个给予豁免，另一个缩小豁免；第三个为生效前既有主体保留过渡安排。语义相似度会偏爱时间上错误的文本。知识库若删掉“过时”版本，还可能在多年后审计旧决定时失去唯一正确证据。

模型不必产生幻觉（Hallucination）才会犯错。它可以完全忠于来源，却不忠于事实适用关系；在引用正确性上成功，在适用正确性上失败。这是工程和法律差别，不是文风差别。

一份文件里的四个日期

应先拆开搜索界面常混在一起的时钟：

1. 发布决定日（Issue Date）：主管机关何时发布文件或通过决定？
2. 公布或可获得日（Publication Date）：文件何时经有效渠道正式公开？
3. 生效日（Effective Date）：义务何时开始，是否分期或受暂停影响？
4. 事件时间与适用期间（Event Time and Applicable Period）：被询问行为何时发生，规则覆盖哪些交易、类别和时期？

这些日期可能相同，也可能相隔数月或数年。机构内部还有第五个日期：系统或员工何时实际收到信息。官方今天发布、知识库一周后收录，会形成两种记录：来源层面当时有效的规则，以及决策当日系统里有什么。二者不自动相同；不知道修订不当然免责，却与追查决定成因有关。

时间数据库（Temporal Databases）早已区分现实中的有效时间（Valid Time）与系统中的事务时间（Transaction Time/System Time）。前者问描述何时在所建模世界中成立，后者问数据库何时保存该描述。二者并存形成双时间数据（Bitemporal Data），可以重建法律记录中的真实状态，也能重建数据库在过去知道什么。IBM 在语言模型进入办公室前就记录了这种方法。4

但法律“有效时间”不是计算机能自行解释的裸日期，可能取决于原法、实施条例、过渡条款、司法命令与地域管辖。数据库可保存解释结果、来源和异议，却不能自行创造有约束力的解释。因此设计应区分已证事实、经解释的适用规则与执行决定。

从旧制度到新制度，但不是所有人同时到达

沙特商业登记新法提供第二个案例。内阁于 2024 年 9 月 17 日批准，9 月 22 日颁布，10 月 4 日在官方公报公布；规定公布 180 天后实施，商务部宣布 2025 年 4 月 3 日正式生效。[56]

新法改为全国范围内一个企业登记，并以年度确认替代旧续期概念；但法令给既有分支登记持有人自生效日起五年整改期。禁止新设分支登记，不等于生效第一天所有旧分支登记都失效或违法。禁止新增与过渡处理既有状态是两回事。[56]

假设机构用内部助手审查供应商。2025 年 4 月，员工问“公司仍有旧分支登记，是否自动拒绝”。若系统只抽取单一登记规则、忽略过渡条款，会在不该确定时给出确定答案；若只查 2025 年前旧版，称仍可按旧方式新设分支，也会从另一边犯错。它必须区分生效前既有登记与生效后新申请，并核验其他规则和有权决定者。这里仍是工程分析，不是一次真实拒绝事件。

生效日因此不是魔法钥匙。一部法整体生效时，内部仍可为旧类别保留过渡安排。若数据库只给整份文件标注 2025 年 4 月 3 日，却没有把过渡条件拆成规则，就会保存正确日期、产生错误结论。理解单位不是文件，而是带范围、类别与时间的具体条款。

正式公布、最终却从未生效的规则

美国案例更复杂：一份真实最终文件和预定生效日都存在，法院却在开始前阻止执行。

2024 年 4 月，美国联邦贸易委员会（Federal Trade Commission, FTC）宣布限制劳动者竞业限制协议的联邦规则。最终规则于 2024 年 5 月 7 日刊入《联邦公报》，预定 9 月 4 日生效。语言模型可以从政府网站找到全文，准确概括既有与新协议条款。[7]

但 2024 年 8 月 20 日，法院命令在预定日期前阻止执行。2025 年 9 月 5 日，FTC 采取撤回上诉的步骤。FTC 官方页面明确说，该规则没有生效，也不可执行。这不代表美国所有竞业协议适用一个结论；州法和其他现行规则另当别论。[8]

这不是给 PDF 增加“生效日”字段就能解决。检索器可能读到 2024 年 9 月 4 日并认定规则已经开始，却漏掉更早改变状态的司法事件。问“今天适用什么”时，它可能引用措辞最正式的原始文本，却忽略说明规则没有生效的状态页。公布文本证明某机关曾在某时作出决定，却不能单独证明该决定后来在每个时点都可执行。

第一个沙特案例要求等待未来日期，第二个要求尊重生效法律中的过渡例外，美国案例则要求理解预定日期根本没有实现。三种法律时间，三种不说谎的模型也会犯下的错误。

计算机知道一个日期时，究竟知道了什么？

设想一条监管规则记录含文件号、条款、发布机关、适用范围、生效与终止日期、知识入库与更正日期、已核实状态和原文链接。同一版本还可能让不同条款在不同日期生效。只为整份文件存一个日期，无法重建这些状态。

问“2026 年 1 月 31 日应履行什么义务”时，不只查数据库在提问日有什么，还要找对该事件类型在 1 月 31 日有效的规则。问“作出旧决定的员工当时能看到什么”则需查系统时间历史。Microsoft SQL Server 的系统版本时间表查询可以重建历史数据库状态，但它不会自动判断某条监管义务的法律效力。

设规则生效起点为 (b)，终点为 (e)，初步检索可检查交易日期 (t) 是否落在半开区间：(b ⩽ t < e)，即包含起点而不包含终点。这是为避免边界时刻冲突而采用的实用计算约定，并非计算法律日期与期限的一般法律规则；法规可能另行规定历法、工作日、例外和具体截止时刻。时间字段还必须保存时区与适用历法，并按原文核验需要精确到日期还是时刻。即便如此，这也只是一层过滤，不能证明实体、地域、交易类型和例外都匹配。

还需用谓词（Predicates）描述实体类型、实体设立日期、当事人身份、所在地、合同类型、事件执行日期、报告日期、豁免以及规则对有关主体的适用性。对于商业登记，只问“这是不是分支登记”并不够，还要问“它是否在规则生效前已经存在”；对于衍生品，只问“合同日期是什么”也不够，还要知道问题针对新报告，还是既有组合的回补录入。时间区间交叉不等于法律适用；系统必须保存条件的来源和解释责任，而不是把代码逻辑伪装成文本自带的唯一含义。

后续更正也不应抹除历史。机构要同时保存当时可见内容与后来更正内容，并明确区分“知识记录在这个日期被替换”与“规则的法律效果从另一个日期起发生变化”。这正是双时间设计的核心。此类数据库可以回答只保留当前行的数据库无法回答的两个问题：对于一项旧事件，今天应把法律效果归于哪条规则？系统提出先前建议时，当时实际持有什么认识？49

时间检索研究实际做了什么？

2026 年 10 月，本文准备前数日，TimelyRAG 论文发表于 *Findings of EMNLP 2026*。研究者把修订时保留大量原文、只改动有限部分的监管文件、政策和服务条款称为“重叠演化文档”（Overlapping-Evolving Documents）。在这种环境中，时间上错误的候选文本在主题上往往并不遥远，反而可能与正确文本相似到让语义检索把它排在前列。10

他们建立了 TimelyQABench，包含修订、例外、有效期间和事件触发条件问题。它是受控的合成基准，不是真实诉讼案件集合，也不是沙特银行决策样本。研究使用语言模型在法律法规、大学规章、公司政策和服务条款四个领域生成韩语问题，再以包括语言模型核验和对有限样本进行人工复核的方法验证；这些限制由研究者明确披露。因此该结果不能证明模型能在所有司法语境中独立作出适用结论。1011

该方法结合语义相关性与时间相关性重排检索结果，并考虑问题是否明确给出日期。在论文测试的比较设置中，报告的最高改进为 nDCG@10 提升 28.6%。这是研究基准上检索排序质量的相对改进，不是法律决定正确性提高 28.6 个百分点，也不能证明系统已经适合提供监管裁断或法律建议。论文还明确指

出，方法依赖初始检索先找到潜在版本，并不能解决所有涉及多个事件的逻辑复杂性。1011

其价值在于隔离了一个不同于“阿拉伯语回答好不好”或“有没有找到相关文本”的问题：两段极相似文字对问题日期的有效性不同，检索器能否正确排序？真正法律答复还需解释范围、规则状态和来源权威，并由专业人员检验。这是相关证据检索与判定适用规则的区别。

LegalBench-RAG 研究精确定位法律支持片段的难度，另一团队提出需要推理的法律检索基准。它们改善系统找材料的能力，却不能替代对某条材料在事件日期是否有效的测试。每项指标只捕捉正确性的一片；把所有碎片压成一个总分，会让不完整系统显得完全可信。1213

从问题开始，而不是从答案开始的运行架构

为银行或公共机构设计监管审查系统时，不应先让大型语言模型“记住最新更新”，而要先把开放问题转成可检查的适用案例：事件日期、待评决定日期、实体或合同类型、管辖、所求行动，以及用户问当前状态还是旧决定是否合法。关键字段缺失时，系统应追问，不用温柔猜测填空。

随后建立四层，各司其职：

第一层，核实来源登记：保存法律、通告、条例及影响效力的裁判原件、链接、数字身份、发布权威、发布与抓取日期。网页后来更新不能伪装成新“发布日期”，历史版本必须保留。

第二层，条款与过渡登记：把开始、终止、修订、例外、延后、暂停、分阶段和特定类别规则关联到来源。版本 1、版本 2 的线性链不够，因为一条可能废止、另一条仍有效，法院也可能暂停条款而不改原文。

第三层，案例约束检索：在确认数据允许时排除不适用候选，再按主题与时间排序，并主动搜索过渡条款、例外、停止命令和较新解释。旧片段有时仍是证明旧状态或特定类别的必要证据，不能激进删除。

第四层，受约束、可审查执行：明确且经过测试的机器可执行政策（Policy-as-Code）处理能够清楚编码的运营义务；模型只提出由文本支持的推理。来源冲突或含义需要专业判断时，交由法律或合规审查。代码不能解决所有解释争议，却能暴露采用的假设。

还要分离回答环境与执行环境。助手可以解释为何需要 ISIN 或 UPI；真正修改组合数据、关闭申请或拒绝登记，必须通过授权渠道和机构条件。目标不是每次点击前长篇哲学，而是明确区分初步解释与会改变权利或产生财务影响的决定。

适用证明：两年后给审计员看什么？

本文提出“时间适用证明”（Temporal Applicability Proof），这不是已发表国际标准或现成产品，而是可开发和测试的工程编辑方案。它随有监管影响的答案附上，不只证明句子来自哪里，还说明文本为什么适用于这个人和这个日期。

证明包含七项：事件身份与日期；来源权威、版本、条款链接；规则有效边界；类别、管辖与例外条件；它与先前、暂停或修订条款的关系；知识何时进入机构数据库；以及置信、解释与复核限制。另保存规则版本的无歧义标识与内容签名。它不是宣告结论定义上正确的印章，而是一组审计员可以拒绝或纠正的证据。

审计体验因此变成两个问题：事件日期什么法源使答案适用？决定日期员工面前是哪版知识？若后来发现此前漏收的官方文件，可以重审特定规则，不必依靠员工记忆重演整个档案。

风险在于把可记录误当成可裁决。证明可能保存正确日期却漏掉例外，案例本身也可能对规则适用有争议。数据库没发现冲突，不等于法律中没有冲突。每次成功测试都必须保留“证据不足”的出口；这不是人工智能失败，而是对自动推断边界的诚实描述。

测试实验室：不捏造事实的系统仍会犯错的五种方式

用一组简单问题让监管助手取得 90% 准确率没有意义。更有用的测试，应让错误答案非常合理，而且文本真实存在。专家建立含明确事实、日期和有理由参考答案的未见案例集：

1. 生效前发布：文件提前数月公布，事件发生在适用日前。2. 已废止但当时有效：复核旧决定需找旧规则，同时禁止将其用于新决定。3. 预定效力从未实现：最终文本遭法院阻止，FTC 案例可作原型。4. 旧类别过渡：新法禁止新增，却给旧主体整改期。5. 新义务覆盖旧交易记录：既有衍生品组合需要回补披露，必须区分交易日与数据更新义务日。

可提出“时间适用准确率”（Temporal Applicability Accuracy）：在经核实、复核的参考判定下，系统选择了时间与范围均正确规则的案例比例；关键数据不足时的拒答单独计算。不能把询问缺失日期当成编造日期，也不能奖励一律拒答。因此还应测量回答覆盖率、澄清请求正确性、有效条款引用准确率、自信错误率，以及官方发布到生产系统更新的延迟。

若测试只用模型生成问题，就不能把分数称作沙特银行流程安全证据。需要人类设计样本、转换日边界案例、交叉通告、独立解释复核和跨日期回测。真实数据受保密限制时，可从公开规则构造相似案例，但不能冒充运营表现。TimelyRAG 的公开数字是研究起点，不是合规部门的现成及格线。

值得认真对待的反对意见

资深技术负责人可能说：通告不多，法务每月复核，文档系统也记录版本，为何还要双时间、条款管理和例外引擎？在许多低风险用途上，他部分正确。若助手只解释不会触发决定的稳定术语，官方更新来源、良好索引、明确复核日期与“不得独立用于决定”的提示可能已足够。

复杂不是美德。糟糕双时间系统会制造虚假控制感和额外成本；保存所有修订也无助于不会解释或分类的团队。架构应与错误可能造成的伤害、来源变化数量、决策频率和机构为旧决定辩护的需求相称。

但当答案进入具有财务、监管或人身影响的执行链，而且机构日后必须说明为何适用此规则而非彼规则时，反对理由变弱。此时数字档案不再只是阅读仓库，而是决策过程的记忆。没有时钟的记忆，即使保存了全部文件，也无法重建原因。

人工智能拥护者也不能用夸大模型能力回应。模型可以熟练解释长文，却无法自行知道所有法院命令、配套通告和豁免类别是否已进入知识库。它能读记录，不能为记录之外世界的完整性作证。这是知识边界，不是更自信文风能够解决的问题。

同一天，两个人，两种答案

最终受影响的是现实中的人，尽管我们不虚构其故事：完成银行披露的员工、理解旧登记状态的企业主，或复核数月前机构决定的负责人。时间错误会把旧时正确行为用新文本判成错误，把被暂停规则说成强制，或抹掉本为避免伤害而设置的过渡期。

软件世界习惯认为最新版修复旧错误；在权利与监管中，最新版可能是旧事件的错误答案，甚至在尚未生效或被暂停时也是今天的错误答案。所有证据仍以正式语言保存，错误因此更诱人：一切看上去都有依据。机构模型最难发现的错误，可能正是它精确证明了一个文本，却没有证明文本适用于案例。

检验机构人工智能成熟度时，不应只问“最新文本是什么”，而要问：作出这项决定的那一天，哪条规则对这一事实有效，我们依据什么知道，系统当时又知道什么？如果它不能拆开三个问题，无论文字多漂亮，也不能称作机构事实。

参考文献

1. SAMA Circular, 24 September 2025.
2. SAMA Rulebook, Effective Date, 1 February 2026.
3. 登记、披露与风险缓释要求，第 10 条，沙特中央银行.
4. IBM, Best Practices for DB2 Temporal Data Management, April 2012.
5. 商业登记法，内阁专家委员会.
6. 商务部，商业登记与商号制度生效，2025 年 4 月 3 日.
7. Federal Register, Non-Compete Clause Rule, May 7, 2024.
8. FTC, Noncompete Rule: status and court order.
9. Microsoft Learn, Query data in a system-versioned temporal table.
10. Nam et al., TimelyRAG, Findings of EMNLP 2026.
11. Nam et al., TimelyRAG full paper and stated limitations, 2026.
12. Pipitone and Alami, LegalBench-RAG, 2024.
13. Zheng et al., A Reasoning-Focused Legal Retrieval Benchmark, 2025.