计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
[doi: 10.15888/j.cnki.csa.009722] [CSTR: 32024.14.csa.009722]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于 PROV 和智能合约的电力市场清算溯源模型①

徐占洋[1], 侍虹言[1], 岳紫玉[2], 赵　鸿[1], 许　健[1], 王　哲[1]

[1](南京信息工程大学 软件学院, 南京 210044)
[2](中国电力科学研究院有限公司 南京分院, 南京 210003)
通信作者: 侍虹言, E-mail: 202212490751@nuist.edu.cn

摘　要: 在当前的电力市场中, 现货日清数据量已达百万或千万级. 随着交易活动的增加和市场结构的复杂化, 确保交易数据的完整性、透明性和可追溯性是我国现阶段市场清算领域待研究的关键问题. 为此, 研究提出了一种基于 PROV 模型和智能合约的电力市场清算数据溯源方法, 旨在通过智能合约自动化存储及更新溯源信息, 从而提高清算过程的透明度和参与方信任. 本方法利用 PROV 模型中的实体、活动和代理等元素, 结合区块链技术的可层次存储及不可篡改性, 记录和追踪电力市场中的交易活动和规则变更. 本方法不仅增强了数据的透明度和市场参与方的信任度, 也优化了数据管理和存储策略, 降低了操作成本. 此外, 本方法为电力市场清算提供了合规性证明, 帮助市场参与方满足日益增长的法规要求.

关键词: 数据溯源; PROV 模型; 电力清算; 数据血缘; 大数据

引用格式: 徐占洋,侍虹言,岳紫玉,赵鸿,许健,王哲.基于 PROV 和智能合约的电力市场清算溯源模型.计算机系统应用. http://www.c-s-a.org.cn/1003-3254/9722.html

Electricity Market Clearing Provenance Model Based on PROV and Smart Contracts

XU Zhan-Yang[1], SHI Hong-Yan[1], YUE Zi-Yu[2], ZHAO Hong[1], XU Jian[1], WANG Zhe[1]

[1](School of Software, Nanjing University of Information Science & Technology, Nanjing 210044, China)
[2](Nanjing Research Division, China Electric Power Research Institute, Nanjing 210003, China)

Abstract: In the current electricity market, the volume of daily spot market clearing data has reached millions or tens of millions. With the increase in trading activities and the complexity of the market structure, ensuring the integrity, transparency, and traceability of trading data has become a key issue to be studied in the field of market clearing in China. Therefore, this study proposes a data provenance method for power market clearing based on the PROV model and smart contracts, aiming to automate the storage and updating of provenance information through smart contracts to improve the transparency of the clearing process and the trust of the participants. The proposed method utilizes the elements of entities, activities, and agents in the PROV model, combined with the hierarchical storage and immutability of blockchain technology, to record and track trading activities and rule changes in the electricity market. The method not only enhances data transparency and trust among market participants but also optimizes data management and storage strategies, reducing operational costs. In addition, the method provides proof of compliance for power market clearing, helping market participants meet increasing regulatory requirements.

Key words: data provenance; PROV model; power clearing; data lineage; big data

① 基金项目: 国家电网有限公司科技项目 (5108-202218280A-2-289-XG); 江苏省研究生科研与实践创新计划 (SJCX23_0410)
收稿时间: 2024-06-04; 修改时间: 2024-06-28; 采用时间: 2024-07-11; csa 在线出版时间: 2024-11-15

电力市场经历了从传统的垂直整合到现代的去中心化交易[1]的演变. 这种转变旨在促进竞争, 提高效率、降低成本, 并推动可再生能源的使用. 随着电力市场结构的复杂化, 清算过程成为确保市场透明度和公平性的关键环节. 然而, 在去中心化的电力市场中, 交易数量庞大且频繁, 这给清算过程带来了巨大的挑战. 电力市场清算涉及多个参与方、复杂的计算规则和大量的数据[2]交互, 这使得追踪和监控数据在清算过程中的流动和变化变得至关重要.

在这样的背景下, 数据溯源成为一个关键需求. 数据溯源[3]以最广泛的形式描述了数据的来源、如何衍生, 以及随时间的更新情况[4]. 溯源不仅可以帮助市场参与方验证交易数据的真实性, 还可以增强市场的整体透明度, 建立参与方之间的信任. 然而, 手动更新和维护溯源信息耗时耗力[5], 且容易出错. PROV 模型 (W3C provenance data model) 作为一种国际标准, 提供了一套描述数据溯源信息的框架, 而智能合约技术, 以其执行自动化规则的能力和安全性, 被视为维护溯源信息的理想工具. 将 PROV 模型与区块链技术结合, 为自动化管理和更新溯源信息提供了可能.

鉴于此, 本研究旨在设计和实现一种基于 PROV 模型和智能合约的电力市场清算数据溯源方法. 该方法首次将 PROV 模型和智能合约应用在电力市场清算领域, 旨在构建全链路清算数据溯源模型, 自动化处理电力市场清算过程中的溯源信息更新, 以确保溯源信息的完整性、透明性和可追溯性. 此外, 本文讨论了如何通过智能合约的设计与实现来优化数据存储策略、自动更新溯源信息、增强市场参与方之间的信任以及提供合规性证明, 为电力市场提供了一种新的数据管理解决方案, 有助于构建更加公平、透明和高效的市场环境.

本文的主要贡献如下.

(1) 提出了基于 PROV 模型的电力市场清算数据溯源关系模型 (PROV_ELce 模型), 支持多粒度的溯源追踪;

(2) 基于智能合约技术设计并实现了溯源信息动态更新 DPTrace_C 算法和存储位置动态更新 DSLD 算法, 自动化处理规则变更、数据修订及存储位置动态决策, 实现了溯源模型的实时更新;

(3) 提出了基于数据属性的动态存储位置决策机制, 优化了数据存储效率和成本, 同时确保了数据的安全性和访问性.

1 相关工作

数据溯源领域的研究[6]覆盖了从基础理论研究到各种实际应用的多个方面, 集中在数据安全监管[7,8]、供应链管理[9]、地理信息系统[10,11]等方面.

目前数据溯源研究工作主要有以下几个关键方面. 从溯源数据的捕获和存储方面, 文献[12]使用 Manta Flow 分析和实现数据沿袭存储的增量更新. 文献[13]提出了一种词嵌入的方法来追踪近似血缘, 使用机器学习 (ML) 和自然语言处理 (NLP) 技术. 其基本思想是通过一组小的常量向量 (每个元组的向量数量是一个超参数) 来总结 (和近似) 每个元组的血缘. 文献[14]在词嵌入方法的基础上进行优化改进, 提出元组向量化编码机制和基于属性重要性的优化算法, 提高溯源的精确率并降低溯源时间复杂度. 但基于词嵌入的方法, 通过计算相似度, 需要丢失一些信息为代价, 不适用于精确率需求高的行业领域.

从溯源信息的查询分析方面, 文献[15]在 1990 年提及数据溯源的概念, 文章提出了一种 Polygen 模型 (标记或注释形式), 用于描述多源异构的关系数据库数据查询结果的溯源信息. 随后, 文献[16]在此基础上增加了 where-provenance 和 why-provenance 的概念, 对 SQL 查询语句进行更进一步的细化. 文献[17]基于半环多项式的概念提出 how-provenance 概念. 文献[18]为了弥补查询与现有的溯源派生方法之间的差距, 提出 what-provenance 概念, 为复杂的 SQL 查询提供溯源推导, 但在处理涉及多表联合、子查询等 SQL 特性时, 可能会产生溯源信息的冗余或遗漏.

从溯源框架和模型开发方面, 文献[19–22]针对数据组件 (Hive、Spark) 提供了一系列数据溯源解决框架. 文献[23]提出了一个可扩展框架 Newt, 用于捕获和使用记录溯源信息, 虽然在一定程度上降低了开销, 但相对昂贵. W3C[24]发布的溯源模型 PROV 是目前为止最成功的模型, 是目前溯源技术最成功的模型. 众多学者基于 PROV 模型进行溯源技术的扩展研究[25,26]. 文献[27]对 PROV 模型进行扩展, 使 PROV 模型支持可溯源的数据库系统, 以实现可溯源的数据库系统与其他支持溯源的系统的互操作性. 文献[28]对 PROV-DM 模型进行扩展, 提出用于数据监管的大数据溯源模型 BDPM, 支持多种数据类型的来源表示和多种数据处

理方式.

从溯源技术的具体应用方面, 文献[29]针对 BIM 领域 (建筑信息建模), 提出基于区块链的溯源模型. 文献[30]提出新颖的可视化方法实现用户交互溯源信息的元分析, 但是该方法只针对特定应用程序.

与已有工作相比, 本文侧重电力市场清算流程的溯源机制研究, 提出了一个面向电力市场清算数据的溯源模型, 并设计了基于数据属性的动态存储位置决策机制和溯源模型自动更新算法, 以应对数据规模的快速增长和复杂性提高.

2 PROV 模型及电力市场清算溯源问题描述

2.1 PROV 模型

PROV 模型是由 W3C (world wide Web consortium) 制定的一种用于表示数据血缘和溯源关系的标准, 用于描述数据的来源和历史, 即数据的溯源 (provenance). 它旨在帮助用户理解数据背后的故事: 数据从哪里来, 经过了哪些处理, 以及谁对数据进行了操作. PROV 模型提供了一套丰富的框架, 允许对数据和过程进行详细的描述, 增加了数据管理过程的透明度和可信度. PROV 模型为数据溯源提供了一种规范化的、通用的方法, 使得在不同领域和应用中都可以有效地记录和分析数据的演化过程.

2.2 问题描述

在电力清算领域, 数据的高度动态性和复杂性使得数据溯源成为一项具有挑战性的任务. 传统的数据溯源技术在电力市场清算中面临种种挑战, 如数据不一致性、溯源信息的不透明以及溯源过程效率低等问题. 这些问题可能导致错误的清算结果, 影响市场的稳定运作. 当前的电力清算数据管理面临着诸多问题, 包括数据血缘关系难以追踪、数据变更管理的困难以及数据质量监控的需求. 这些问题影响了电力清算系统的透明度、可信度和数据质量.

本研究旨在解决如何通过技术手段提高电力市场清算数据的溯源能力, 具体研究问题包括 3 个.

(1) 如何增强数据溯源的透明度和可靠性?

探索基于 PROV 模型的方法来详细记录电力市场清算数据的来源和变化, 提供全面的数据流跟踪和验证.

(2) 如何优化存储资源利用?

探索如何利用数据结构减少冗余数据存储, 以及通过智能合约自动化数据处理步骤, 从而有效管理和优化数据存储.

(3) 如何提高数据处理的时间效率?

利用智能合约来自动执行溯源信息变更, 以减少人工操作和等待时间, 实现实时或接近实时的数据处理.

通过解决上述问题, 本研究期望填补现有研究的空白, 为电力市场清算提供一种新的、更高效和安全的数据溯源方法. 此外, 这项研究的成果也可推广至其他领域, 为广泛的数据密集型行业问题提供解决方案.

3 电力市场清算溯源方法

电力市场清算数据溯源方法的总体框架如图 1 所示. 主要分为前期溯源模型构建部分、溯源信息存储部分和溯源动态管理部分. 溯源模型构建部分将经过初步处理的数据送入数据融合引擎, 该引擎对来自不同源的数据进行合并和冲突解决, 确保数据的统一性. 在此过程中, 每一个数据处理步骤都通过 PROV_ELce 模型进行标记, 记录详细的数据来源和处理历史, 生成溯源信息, 构建完整的溯源图. 溯源信息存储部分, 使用区块链进行层级存储策略, 优化数据管理. 溯源动态管理部分, 智能合约承担起动态管理溯源信息和数据存储位置的重要角色: (1) 存储位置动态变更: 为优化数据存取效率和响应速度, 智能合约根据数据访问频率和相关性自动调整数据存储位置, 例如将频繁访问的数据迁移到更高速的存储系统; (2) 溯源信息动态更新: 智能合约根据预定规则自动更新溯源信息, 如规则变更、数据修订等, 确保溯源信息的时效性和准确性.

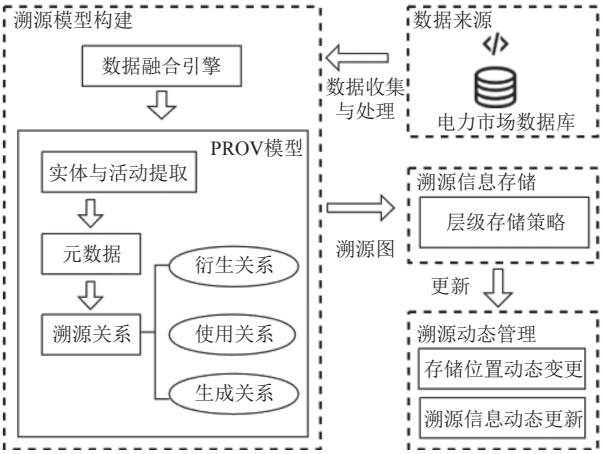

图 1 电力市场清算数据溯源方法总体框架

3.1 PROV_ELce 模型构建

本文基于 PROV 模型提出了一个电力市场清算数据溯源关系模型 (PROV_ELce 模型), 以实现市场清算数据的全面追踪和溯源, 保证数据的可信度和可靠性, 提高清算过程的透明度和效率.

3.1.1 电力市场清算数据溯源信息概念模型

本文重点关注电力清算过程中产生的各类实体关系及元数据, 仅考虑电力市场中涉及清算过程的步骤, 而不考虑清算过程中涉及的数据格式与类型转换.

在电力市场清算中, 涉及的清算主体主要有 4 类: 用户、售电公司、发电商以及电网企业. 电力市场清算的主要原因包括计算数据修正、计算规则变更、账单计算错误、结算价格调整.

本文将电力市场清算数据元信息作为研究对象, 将数据溯源信息抽象为活动、实体、关系和属性等 4 类要素. 记录电力市场清算的处理流程, 以及数据源、使用的算法、软件环境的配置以及活动执行人等关键信息. 构建的通用模型如图 2 所示. 活动表示在电力市场清算过程中发生的具体事件或操作, 包括数据处理、结算计算、报表生成等; 实体表示在电力市场清算过程中涉及的具体对象或数据, 包括清算规则、人/机构等; 关系表示活动、实体之间的关联关系, 用于描述它们之间的相互作用和影响 (例如交易与结算结果的产生关系); 属性表示活动、实体和关系的具体特征或属性信息, 用于进一步描述它们的细节或状态 (比如结算结果的属性有结算金额、结算时间等信息).

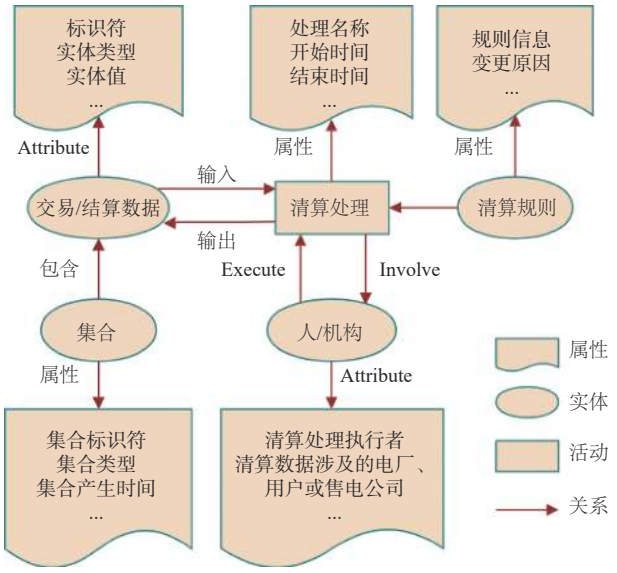

图 2 清算数据溯源信息概念模型

为了进一步对概念模型进行阐述, 表 1 列出了几类核心要素的概念及元素示例.

表 1 溯源信息概念模型的核心要素及示例

要素	概念	示例
实体	表示在电力市场清算过程中涉及的具体对象或数据	交易数据、结算数据、报表数据
活动	表示在电力市场清算过程中发生的具体操作或事件	数据处理、结算操作、报表生成
关系	活动与实体间的关系	结算操作与结算结果的生成关系 结算操作与交易数据的使用关系、影响关系
	实体与实体间的关系	报表数据与结算数据的依赖、衍生关系 交易数据与结算数据的衍生和使用关系
	活动与活动间的关系	顺序关系、依赖关系、并行关系
属性	表示活动、实体和关系的具体特征或属性信息	交易的属性包括交易类型、交易数量、交易价格等信息

以电力交易实时市场为例 (如图 3 所示), 介绍清算数据溯源信息概念模型 (清算过程中涉及的主体、规则、过程等复杂且数据量巨大, 图 3 中示例仅为一小部分). 中国电力科学研究院为本文研究提供了部分清算数据集、电力清算市场数据结构以及市场清算流程.

图 3 中交易数据 2 与清算数据之间经过了结算操作, 因此交易数据 2 与结算数据之间存在衍生和使用关系. 交易数据 1 与交易数据 2 之间经过了数据处理, 该操作对交易数据 1 进行验证、清洗和分析的操作, 得到交易数据 2, 因此交易数据 1 与交易数据 2 之间存在衍生关系. 在报表生成活动中, 通过聚合过滤算法将交易数据 2 和结算数据整合得到报表数据. 根据上文提到电力清算的 4 类原因, 图 3 中将 4 类原因统一表示为异常、变更处理. 当电力交易中心收到变更请求时, 执行该活动, 对交易数据 2 及结算数据及时进行变更 (这里不考虑电力数据的版本化记录问题).

3.1.2 PROV_ELce 模型扩展与映射

本文通过对电力交易实时市场进行数据清算来介绍本文模型. 在市场清算数据溯源关系模型中, PROV 模型用于记录数据的生成过程、转换过程和消费过程. PROV 模型主要包括 3 个核心概念: 实体 (entities)、活动 (activities) 和代理 (agents), 以及这些概念之间的关系 (relations), 用于描述清算数据之间的关系和操作过程. 实体代表电力市场清算过程中的数据、文档或任何有价值的信息资源; 活动代表在电力市场清算过程

中发生的操作或过程, 这些活动会影响实体的状态或属性. 代理代表在电力市场清算过程中执行活动或影响实体的个人、组织或系统. 关系描述实体、活动和代理之间的相互作用和依赖. 在 PROV 的核心结构中, 实体、活动以及代理之间主要有 7 种关系, 例如生成 (wasGeneratedBy)、使用 (used)、归因于 (wasAttributedTo) 等.

本文针对电力实时市场清算过程, 对 PROV 模型进行了进一步的抽象和扩展, 构建了清算数据溯源信息概念模型和 PROV_ELce 模型的映射框架, 如图 4 所示. 不同形状分别代表 PROV_ELce 模型中的实体、活动与代理 3 种元素.

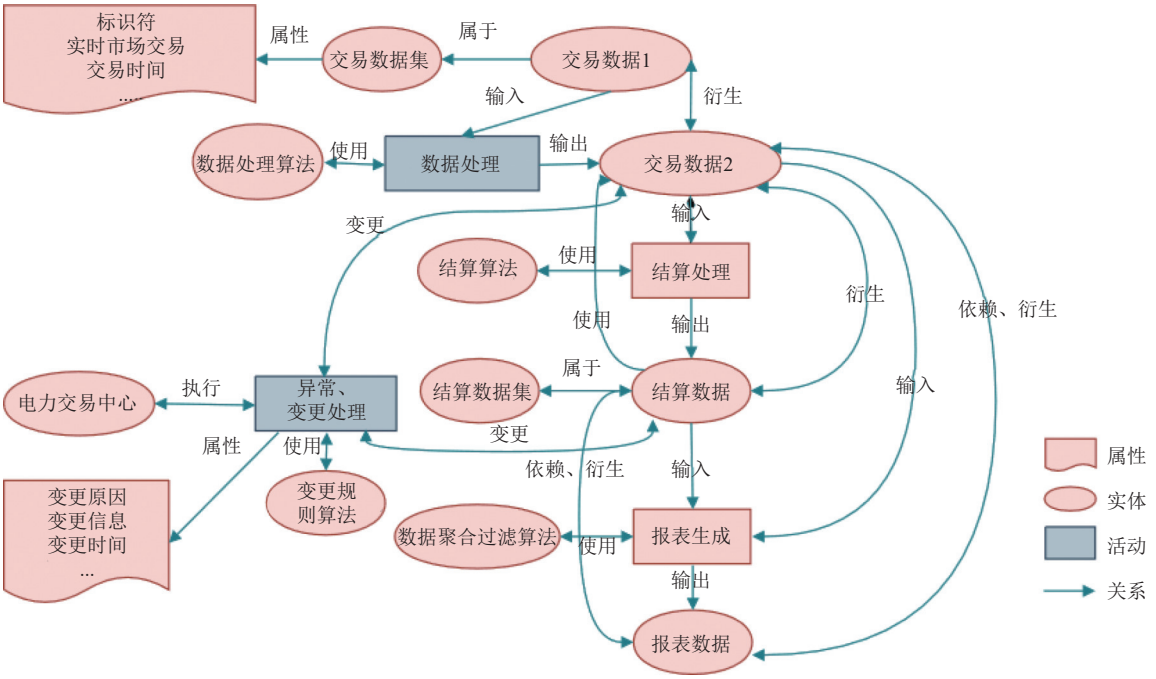

图 3 清算溯源信息表达简单示例

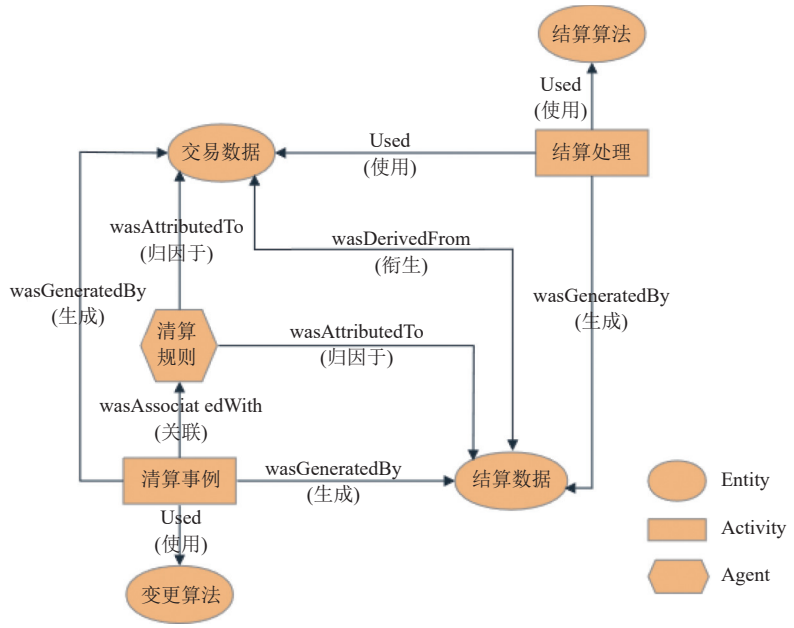

图 4 PROV_ELce 模型映射框架

这里将清算过程中发生的具体操作 (即溯源概念模型中的活动) 抽象为活动; 将清算过程中涉及的对象 (即溯源概念模型中的实体) 抽象为实体; 当发生一个清算事例, 均涉及一个清算规则变更或执行, 考虑到清算过程中涉及的对象实体巨大, 为了降低模型复杂度及简化后续查询过程, 这里将清算规则抽象为代理, 执行机构及时间等信息作为附加属性, 而不作为一个单独代理. 活动、实体和代理之间的关系由溯源概念模型中的关系抽象而来.

定义 1. 清算过程数据溯源[31]. 给定一个数据清算过程$P=\{p_1,p_2,\cdots,p_n\}$、一个实体集$E=\{e_1,e_2,\cdots,e_n\}$、一个规则集$R=\{r_1,r_2,\cdots,r_n\}$. 数据清算过程中的每一个步骤p_i对应于溯源模型中的一个活动, 实体集中的e_i对应于模型中的一个实体, 规则集中的r_i对应于模型中的代理. p_i、e_i、r_i之间的关系继承于 PROV_ELce 中实体、活动和代理之间的关系.

考虑电力市场清算的透明度问题, 本文选择图形化表达 (清算数据溯源图) 进行实例化表达. 聚焦于计算规则的变更 (某时期交易价格调整), 介绍清算数据溯源图的形成过程. 假定存在两个不同的规则, 分别为旧规则r_1和新规则r_2. 首先, 进行计算规则变更过程, 记录该过程的溯源信息, 即从r_1到r_2计算规则的变更信息.这一步骤涉及记录原始计算规则与变更后计算规则之间的对应关系, 以及变更的原因和执行的活动.

接下来对影响到的交易记录进行更新, 识别基于旧规则计算但需要按照新规则重新计算的交易记录, 交易记录e_1按照r_1计算, 需要更新为按照r_2计算的e_2. 这一过程中, 每个交易记录的变更可以视为一个活动. 在这个过程中, e_1是源实体, e_2是目标实体. 随着时间推移, 更新后的交易记录e_2因为进一步的规则变更或其他因素而再次发生变化, 得到最终的演化实体e', 这一演化过程也被详细记录下来. 最后, 基于上述计算规则变更、交易记录更新和演化过程, 形成了溯源图. 溯源图不仅记录了从原始交易记录到更新后交易记录的变更过程, 还包括了计算规则变更的详细信息, 以及每个变更活动的执行情况和影响范围.

通过溯源图, 可以清晰地追踪电力市场清算中计算规则的变更历史, 以及这些变更对交易记录和清算结果的具体影响. 为电力市场的管理、分析和审计提供了一种有效的数据溯源手段, 增强了电力市场运作的透明度和可追踪性.

3.2 溯源信息存储

电力市场清算数据溯源模型覆盖了整个电力市场清算过程, 这就导致存储的数据维度复杂、溯源信息所需的成本很高等问题. 因此, 本文采用层级化的数据存储策略 (BProv-HS), 将电力市场清算数据溯源信息存储于区块链. 该方法通过层级化存储来优化数据管理和访问效率. 下文以电力市场清算实时市场为例, 详细说明基于层级区块链的清算数据溯源模型存储策略 BProv-HS.

3.2.1 层级存储策略

在研究电力市场清算数据溯源模型的层级存储过程中, 本研究详细考虑了清算数据的多样性和不同的查询需求. 为了更高效地组织和管理这些数据, 本文根据数据的类型和访问频率来设计存储层级, 同时特别设置了受限数据层来存储需要保护的敏感数据. 在此对层级存储策略作说明.

定义 2. 数据类型层级L_T. 定义一个集合$T=\{T_1,T_2,\cdots,T_n\}$, 其中每个T_i表示一种特定的清算数据类型, 包括原始交易数据、清算规则、清算结果、变更记录等. 对应的数据类型层级为$L_T=\{L_{T_1},L_{T_2},\cdots,L_{T_n}\}$, 每个$L_{T_i}$表示$T_i$数据类型的存储层级.

定义 3. 访问频率层级L_F. 定义访问频率集合$F=\{F_1,F_2\}$, 其中F_1表示高频访问数据 (如近期交易记录、实时市场价格), F_2表示低频访问数据 (如旧的清算结果、历史交易数据). 对应的访问频率层级$L_F=\{L_{F_1},L_{F_2}\}$, 其中L_{F_1}用于存储F_1类型的数据, L_{F_2}用于存储F_2类型的数据.

定义 4. 受限数据层级L_S. 定义数据敏感性集合$S=\{S_1,S_2\}$, 其中S_1表示受限数据, S_2表示非受限数据. 对应的受限数据层级为$L_S=\{L_{S_1},L_{S_2}\}$, 仅L_{S_1}用于存储S_1类型的敏感数据, 采用加密和权限控制保护.

定义 5. 层级存储策略 HSS-SD. 定义一个三元组 HSS-SD(L_T,L_F,L_S)来表示电力市场清算溯源模型的层级存储, 其中 P 表示数据的存储位置, L_T表示T数据类型的存储层级, L_F表示高频访问层或低频访问层, L_S表示受限数据层或非受限数据层.

例如, 一个数据项是高频访问的原始交易数据且不涉及敏感信息, 它的层级三元组为(T_1,F_1,S_2), 其中T_1表示处于原始交易数据层, F_1表示处于高频访问层, S_2表示处于非受限数据层.

3.2.2 存储位置决策编码方案

区块链存储实现通过将不同类型和特性的数据项按照预设的层级结构有效地映射到区块链结构中, 这种映射基于数据项的特性 (类型、访问频率、敏感性) 来优化存储资源使用、提高查询效率. 本文使用主链与侧链的层级区块链存储模型.

● 主链 (main chain): 存储全局共识数据, 包括清算规则变更历史、关键交易数据等. 主链的目的是保证整个系统的安全性和统一.

● 侧链 (side chain): 存储需要频繁更新或具有特定应用需求的数据项, 包括实时交易数据、频繁变动的清算结果等.

针对层级存储策略, 本文设计一种动态存储位置决策编码方案, 通过设定的规则条件, 基于 T、F 和 S 的特定组合来自动决定数据项的存储位置.

编码规则如下所述.

数据类型 T: 占 n bit, 根据数据的具体类型赋予相应的编码值;

访问频率 F: 占 1 bit, “0”表示低频访问, “1”表示高频访问;

数据敏感性 S: 占 1 bit, “0”表示非敏感数据, “1”表示敏感数据;

存储位置 P: 用于记录当前存储位置的决策, 特定值对应主链或不同的侧链.

存储位置决策规则:

规则 1. 主链规则: 对于具有高数据敏感性 (S=1) 的数据, 无论其类型和访问频率如何, 均应存储在主链上以确保最高级别的安全和不可篡改性.

规则 2. 高频访问侧链规则: 对于高频访问 (F=1) 但非敏感 (S=0) 的数据, 选择存储在特定的高性能侧链上, 以优化访问速度和处理效率.

规则 3. 低频访问侧链规则: 低频访问 (F=0) 的数据, 根据其类型 (T) 和非敏感性 (S=0), 存储在专门的低频访问数据侧链上, 以节约主链资源.

基于编码规则, 为每个新产生或变更的数据项按照其敏感性、访问频率和类型进行分类, 并赋予相应的 S、F、T 编码. 根据数据项的 S、F、T 编码, 利用智能合约按照存储位置决策规则, 自动决定数据项的初始存储位置 P (P 值不作为数据编码的固定部分静态记录, 而是每次根据需要通过智能合约动态计算得到), 确定数据项应当存储在主链还是特定的侧链. 如敏感性高的数据优先存储于主链以保障安全, 而高频访问的数据则存储于某个侧链以提高查询效率.

3.3 溯源动态管理

溯源动态管理是确保溯源信息的持续性和准确性的关键过程, 包括动态存储位置更新和溯源信息动态更新两个方面. 动态存储位置更新涉及根据数据量和存储需求的变化, 动态调整溯源信息的存储位置, 以保证存储资源的有效利用和系统性能的优化. 而溯源信息动态更新则包括对溯源信息的实时更新和修订, 以反映数据的变化和修订历史, 确保溯源信息与数据的一致性和完整性. 下文详细说明动态存储位置更新和溯源信息动态更新两个方面.

3.3.1 动态存储位置更新

在区块链系统中, 智能合约是自动执行的程序, 能够在满足预设条件时执行定义好的逻辑. 智能合约在动态存储位置更新中的应用流程图如图 5 所示.

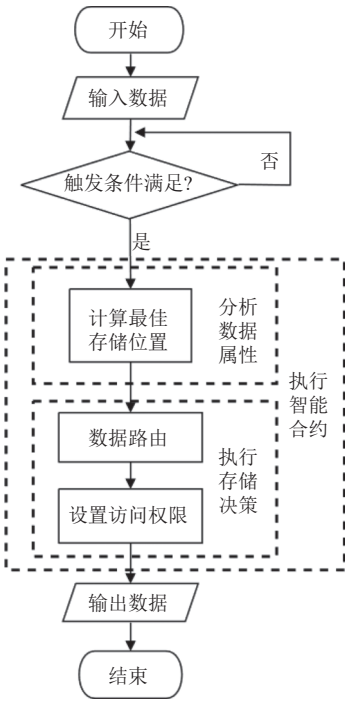

图 5 智能合约执行动态存储位置更新流程图

首先进行触发条件检测, 这些条件包括新数据的创建、现有数据的更新. 触发后, 执行动态存储决策算法 (DSLD). 算法的具体过程如算法 1 所示.

算法 1. 动态存储位置决策算法 (DSLD 算法)

输入: 变更数据 D.
输出: 更新的存储位置 P_{new}.

```
1. 初始化存储位置 P_new←Φ
2. T ← getDataType(D)
3. F ← getAccessFrequency(D)
4. S ← getDataSensitivity(D)
5. If S == 'Sensitive' Then
6.     P_new ← MainChain
7. Else
8.     If F == 'High' Then
9.         P_new ← HighPerformanceSideChain
10.    Else
11.        P_new ← LowFrequencyAccessSideChain
12.    EndIf
13. EndIf
14. D.LocationUpdated = P_new // 更新位置
15. SetAccessPermissions(D) // 设置访问权限
16. NotifyStakeholders (D) // 通知相关方数据项 D 的状态变更
17. Return P_new
```

智能合约分析数据属性. 这包括确定数据类型 (T)、访问频率 (F) 和敏感性 (S). 基于分析的属性, 智能合约计算数据项的最佳存储位置. 这一步骤涉及比较不同的存储选项 (主链或侧链) 并选择最合适的位置. 接下来, 执行数据路由操作, 将数据项转移到计算出的存储位置. 同时, 智能合约根据数据的敏感性设置相应的访问权限, 确保数据安全. 最后, 智能合约更新链上的相关记录, 以反映数据项的当前存储位置和访问权限状态, 并告知相关方数据存储位置的变更或操作的成功执行.

3.3.2 溯源信息动态更新

溯源信息动态更新的原因包括规则变更、数据修订等方面. 考虑到电力市场特有的需求和挑战, 本文选择结合智能合约设计 DPTrace_C 算法实现电力市场清算溯源模型更新, 以自动化地响应和处理规则变更和数据修订. DPTrace_C 算法设计过程中考虑了电力市场的特殊需求, 包括清算的实时性、数据的准确性和透明度, 以及参与各方对于合规性和安全性的高要求.

图 6 展示了 DPTrace_C 算法的核心组件, 由规则变更监听器、数据修订处理器、存储位置动态决策机制以及溯源信息更新引擎构成. 算法的具体过程如算法 2 所示.

算法 2. 溯源信息动态更新算法 (DPTrace_C 算法)

输入: 事件 E (RuleChange、DataRevision)、相关数据 D.
输出: 更新后的溯源信息 I.

```
1. 初始化 I ← Φ
2. EventType ← getEventType(E)
3. Begin
4.     Switch EventType do
5.         Case RuleChange:
6.             UpdateRule(EventData) //更新溯源模型中的规则
7.             I ← RecordRuleChange() //记录规则变更细节到区块链
8.             TriggerRuleChangeEvent() //触发规则变更通知事件
9.         Case DataRevision:
10.            CompareRevision(EventData) //对比修订前后数据
11.            UpdateDataItem(EventData) //更新溯源模型中的数据项
12.            If AffectsStorageLocation(EventData) //修订影响存储位置
13.            Then
14.                do DSLD //执行存储决策算法
15.            EndIf
16.            I ← RecordDataRevision() //记录数据修订细节到区块链
17.            TriggerDataRevisionEvent() //触发数据修订通知事件
18.    EndSwitch
19. End
20. Return I //返回更新状态
```

当接收到规则变更通知或者数据修订请求时, 触发 DPTrace_C 算法; 根据事件类型 (规则变更或数据修订) 执行相应的处理逻辑; 接下来, 自动更新溯源模型中的相关信息, 包括交易记录、清算规则等; 如果需要存储位置决策, 根据数据属性和市场当前状态, 重新计算数据的最佳存储位置, 并执行数据迁移. 最后, 记录更新操作的详细信息, 并通过智能合约事件通知所有相关方溯源信息已更新.

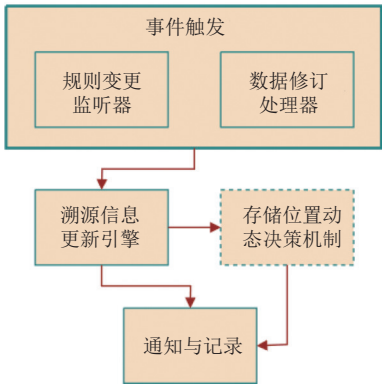

图 6 DPTrace_C 算法概念图

4 实验结果

本节基于提出的电力市场清算数据溯源方法, 在电力科学研究院提供的电力实时市场数据集上测试本文算法.

4.1 数据集

本文使用的清算数据集由 2023 年 3 月–2023 年

5 月江苏省电力交易中心生成 (交易中心每 15 min 生成一条交易数据, 选取其中 1 000 台机组的交易数据). 数据集覆盖了电力实时市场的核心活动和元素, 包括详尽的交易记录、参与方的详细信息、电力需求与供应的日常动态以及规则变更的记录. 具体统计信息如表 2 所示. 将这些数据应用于 PROV_ELce 模型, 以记录并存储市场清算流程的溯源信息, 进而分析交易的溯源、验证数据的完整性和一致性, 实现电力市场清算数据的溯源机制.

表 2 清算数据集统计信息

名称	记录数	字段示例	描述
交易记录表	5 680 000	交易ID、发起方ID、接收方ID、交易时间、交易金额、交易状态、交易类型	记录电力市场的所有交易详细信息
机组信息表	1 000	机组ID、名称、类型、注册时间、位置	描述机组的基本信息
电力需求与供应表	90	日期、总需求量、总供应量、平均价格	每日电力需求与供应情况及平均交易价格
规则变更记录表	41	变更ID、变更日期、变更描述、变更前公式、变更后公式	记录市场规则的变更历史

4.2 实验结果及分析

4.2.1 溯源精确率分析

(1) 精确率评价指标

“准确”的溯源信息表示溯源信息能够真实、完整地反映数据的来源、变更历史、数据流向以及数据处理过程. 本文构建基准数据集 (Base_collect), 包含已知溯源信息的数据记录, 这些记录详细描述数据的来源、变更和处理过程. ProvLineage 表示 PROV_Elec 模型生成的溯源信息集合. 实验过程中使用 $F1$ 值作为精确率的评价指标, 公式如下.

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{1}$$

$$Precision = \frac{|P \cap B|}{|P \cap B| + |P - (P \cap B)|} \tag{2}$$

$$Recall = \frac{|P \cap B|}{|P \cap B| + |B - (P \cup B)|} \tag{3}$$

其中, P 表示 ProvLineage 中的数据集合. B 表示 Base_collect 中的数据集合. $Precision$ 代表返回的精确率, $Recall$ 代表返回的召回率,

(2) 实验结果分析

如表 3 所示, 本文比较了 3 种不同溯源方法的 $F1$ 值, 包括基于日志的溯源方法 (Log_trace)、未进行映射扩展的 PROV 模型 (PROV)、基于 PROV 的电力市场清算数据溯源关系模型 (PROV_ELce). 由表 3 的 $F1$ 值对比结果可知, 在小数据量处理时, 所有方法都能维持较高的精确率, 因为数据较少, 处理和管理相对容易, 溯源错误的机会较低. 随着数据量的增加, 各方法的精确率进一步下降, 尤其是对于 Log_trace 和 PROV, 因为大数据量可能增加溯源过程中的错误和遗漏.

表 3 精确率对比结果

数据大小 (MB)	Log_trace	PROV	PROV_ELce
0.5	0.743 55	0.761 69	0.884 87
5	0.729 54	0.749 65	0.861 33
50	0.694 20	0.714 48	0.855 26
500	0.545 96	0.684 73	0.810

在不同数据大小下, PROV_ELce 模型相比 PROV 模型的提升幅度不尽相同. 在数据大小为 0.5 MB 和 5 MB 时, PROV_ELce 模型的 $F1$ 值相对提升较大, 但在数据大小为 50 MB 时, 提升幅度略有下降. 这可能是由于数据量增加导致的复杂性增加, PROV_ELce 模型的优势减弱. 但当数据量级从 50 MB 提升到 500 MB 时, PROV_ELce 模型的降低幅度最小 (幅度为 5%), 这表明随着数据量级的增长, PROV_ELce 模型保持了一个稳定的精确率. 与 Log_trace 和 PROV 相比, PROV_ELce 模型在所有数据大小下均表现出较高的 $F1$ 值. 这表明采用 PROV_ELce 模型可以显著改善清算市场数据溯源的准确性. 在 500 MB 数据量级下, 虽然 PROV_ELce 模型的 $F1$ 值最高, 但仍存在 19% 的溯源失败率. 这是因为电力市场清算是一个复杂的过程, 存在频繁的规则变更和数据更新步骤, 随着数据量的不断扩大和数据关系复杂性的提升, 系统性能逐渐不足, 影响了溯源的精确率.

4.2.2 溯源时间效率分析

(1) 时间效率评价指标

本文将从数据更新到得到最终溯源信息的消耗时间作为评价指标, 单位为 s.

(2) 实验结果分析

如图 7 所示, 本文比较了 5 种不同溯源方法在处理不同数据量下 (0.5 MB、5 MB、50 MB) 的更新时间效率, 包括基于日志的溯源方法 (Log_trace)、基于 PROV 的电力市场清算数据溯源关系模型 (PROV_ELce)、仅加入 DSLD 算法的 PROV_ELce 方法 (PROV_

ELce+DSLD)、加入 DSLD 算法和 DPTrace_C 算法的 PROV_ELce 方法 (PROV_ELce+ DSLD+DPTrace_C)、基于智能合约的 PROV_ELce +DSLD+DPTrace_C 方法.

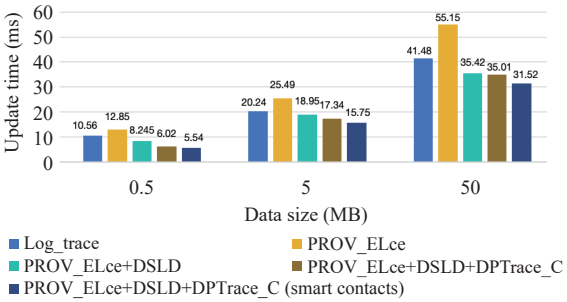

图 7 不同方式时间效率对比

所有方式在数据量增大时, 更新时间均有所增加. Log_trace 方式在小数据量下表现良好, 但随着数据量的增加, 其性能提升不如基于 PROV_ELce 模型的方法, 可能是由于日志处理和分析的复杂度随数据量增大而显著增加. 由图 7 可以看出, PROV_ELce 模型加入 DSLD 算法后, 更新时间降低了 20% 左右. 引入 DPTrace_C 算法后, 更新时间进一步降低, 这表明 DSLD 算法优化了数据存储位置, 减少了处理时间, DPTrace_C 算法通过动态更新减少了数据处理的时间, 提高了时间效率. PROV_ELce+DSLD+DPTrace_C (smart_contacts) 在所有数据量级中均显示出最佳的更新时间性能, 尤其在大数据量 (50 MB) 下, 基于智能合约能够显著减少更新时间. 进一步验证了将动态存储位置更新和溯源信息动态更新机制结合并基于智能合约使用, 对提高时间效率的重要性.

4.2.3 存储资源消耗分析

(1) 存储资源消耗评价指标

本文使用溯源更新时数据存储所占用的空间大小作为评价指标, 单位为 MB.

(2) 实验结果分析

如图 8 所示, 本文比较了 5 种不同溯源方法在处理不同数据量下 (0.5 MB、5 MB、50 MB) 的存储资源消耗. 由图 8 显示, 随着数据量级增加, PROV_ELce+DSLD 方式的存储消耗比 PROV_ELce 方式降低了 20% 左右, 在一定程度上优化了存储消耗, 因为它能够通过智能地安排数据存储位置来减少不必要的存储空间使用. PROV_ELce+ DSLD+ DPTrace_C 方式比起 PROV_ELce + DSLD 方式, 存储资源消耗进一步降低, 这表明 DPTrace_C 算法通过动态更新减少了重复信息的存储, 可以提高存储效率, 减少不必要的资源占用. 基于智能合约的 PROV_ELce+ DSLD+DPTrace_C 方法在所有数据量级别下的表现最佳, 尤其在大数据量下能够显著减少存储消耗, 这是因为智能合约通过自动化执行和实时处理, 进一步优化了数据存储位置, 减少了冗余数据的存储. 本文方式在所有数据量级别上展示了最低的存储消耗.

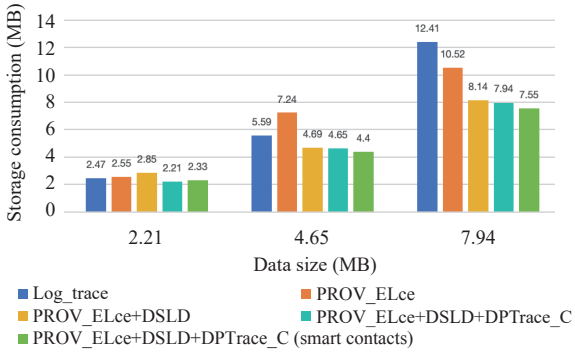

图 8 不同方式资源消耗对比

5 结论与展望

本研究成功设计并实现了一种基于 PROV 模型和智能合约的电力市场清算数据溯源方法. 通过引入动态存储位置更新策略和溯源信息的动态更新机制, 本方法能够有效地管理和跟踪电力市场清算过程中的数据变化和规则应用. 实验结果表明, 与传统数据溯源管理方法相比, 本方法在数据溯源的准确性、存储资源消耗和更新时间效率方面展现出显著的改进.

本研究证明了本文方法在电力市场清算数据管理中的有效性. 特别是, 动态存储位置更新策略和溯源信息动态更新机制的应用, 不仅提高了数据处理的时间效率, 而且降低了存储资源的消耗. 这些改进使得电力市场运营商能够更加快速和准确地进行决策支持和风险管理.

目前数据溯源领域的研究成果较少. 尽管本方法在多个方面表现出优越性, 但溯源精确率仍需进一步提高及在实际应用中仍面临一些挑战. 其中包括处理高频更新的数据溯源信息所需的计算资源管理、维护溯源信息的长期存储成本等. 下一步的研究将聚焦于开发更高效的算法来提高精确率、减少溯源信息更新的计算开销和提升处理速度, 并探索新的数据存储和

处理架构, 以支持更大规模的电力市场清算数据溯源.

参考文献

1 裴林, 黄成, 杨啸, 等. 考虑隐私保护和去中心化的分布式能源交易模式研究. 电力系统保护与控制, 2024, 52(2): 143–154. [doi: 10.19783/j.cnki.pspc.230549]

2 龙苏岩, 盛祥祥, 周天翔, 等. 基于统一数据模型的电力现货市场清算方法及其应用. 电力系统自动化, 2021, 45(6): 169–175. [doi: 10.7500/AEPS20200602004]

3 Buneman P, Tan WC. Provenance in databases. Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data. Beijing: ACM, 2007. 1171–1173.

4 Ikeda R, Widom J. Data lineage: A survey. Technical Report 918. Stanford InfoLab, 2009.

5 Wang JW, Crawl D, Purawat S, *et al*. Big data provenance: Challenges, state of the art and opportunities. Proceedings of the 2015 IEEE International Conference on Big Data. Santa Clara: IEEE, 2015. 2509–2516. [doi: 10.1109/bigdata.2015.7364047]

6 高明, 金澈清, 王晓玲, 等. 数据世系管理技术研究综述. 计算机学报, 2010, 33(3): 373–389.

7 Pan BF, Stakhanova N, Ray S. Data provenance in security and privacy. ACM Computing Surveys, 2023, 55(14s): 323. [doi: 10.1145/3593294]

8 Porkodi S, Kesavaraja D. Secure data provenance in Internet of Things using hybrid attribute based crypt technique. Wireless Personal Communications, 2021, 118(4): 2821–2842. [doi: 10.1007/s11277-021-08157-0]

9 陈华. 食用油产品溯源查询系统的建立与应用 [硕士学位论文]. 长沙: 湖南农业大学, 2010.

10 王宏, 于雪鸥, 乔东玉, 等. 基于“四库合一”的地质大数据管理研究及应用. 能源与环保, 2023, 45(5): 110–116. [doi: 10.19389/j.cnki.1003-0506.2023.05.018]

11 吴敏, 张明达, 李盼盼, 等. 面向多源遥感影像数据的溯源模型研究. 地球信息科学学报, 2023, 25(7): 1325–1335. [doi: 10.12082/dqxxkx.2023.230085]

12 Pokorný J, Sykora J, Valenta M. Data lineage temporally using a graph database. Proceedings of the 11th International Conference on Management of Digital EcoSystems. Limassol: ACM, 2019. 285–291. [doi: 10.1145/3297662.3365794]

13 Leybovich M, Shmueli O. ML based lineage in databases. arXiv:2109.06339, 2021.

14 杨彬, 高俊涛, 王志宝, 等. 基于词嵌入的元组级数据溯源方法. 计算机技术与发展, 2023, 33(12): 49–57. [doi: 10.3969/j.issn.1673-629X.2023.12.007]

15 Wang YR, Madnick SE. A polygen model for heterogeneous database systems: The source tagging perspective. Proceedings of the 16th International Conference on Very Large Data Bases. Brisbane: Morgan Kaufmann Publishers Inc., 1990. 519–538.

16 Buneman P, Khanna S, Wang-Chiew T. Why and where: A characterization of data provenance. Proceedings of the 8th International Conference. London: Springer, 2001. 316–330. [doi: 10.1007/3-540-44503-x_20]

17 Müller T, Engel P. How, where, and why data provenance improves query debugging: A visual demonstration of fine-grained provenance analysis for SQL. Proceedings of the 38th IEEE International Conference on Data Engineering. Kuala Lumpur: IEEE, 2022. 3178–3181. [doi: 10.1109/icde53745.2022.00292]

18 Müller T, Dietrich B, Grust T. You say ‘what’, i hear ‘where’ and ‘why’: (Mis-)interpreting SQL to derive fine-grained provenance. Proceedings of the VLDB Endowment, 2018, 11(11): 1536–1549. [doi: 10.14778/3236187.3236204]

19 Apache Atlas. https://atlas.apache.org/index.html#/. [2024-05-20]

20 Tang MJ, Shao SS, Yang WQ, *et al*. SAC: A system for big data lineage tracking. Proceedings of the 35th IEEE International Conference on Data Engineering. Macao: IEEE, 2019. 1964–1967.

21 Armbrust M, Xin RS, Lian C, *et al*. Spark SQL: Relational data processing in spark. Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. Melbourne: ACM, 2015. 1383–1394.

22 Scherbaum J, Novotny M, Vayda O. Spline: Spark lineage, not only for the banking industry. Proceedings of the 2018 IEEE International Conference on Big Data and Smart Computing. Shanghai: IEEE, 2018. 495–498.

23 Logothetis D, De S, Yocum K. Scalable lineage capture for debugging DISC analytics. Proceedings of the 4th annual Symposium on Cloud Computing. Santa: ACM, 2013. 13. [doi: 10.1145/2523616.2523619]

24 Missier P, Belhajjame K, Cheney J. The W3C PROV family of specifications for modelling provenance metadata. Proceedings of the 16th International Conference on Extending Database Technology. Genoa: ACM, 2013. 773–776. [doi: 10.1145/2452376.2452478]

25 Pérez B, Rubio J, Sáenz-Adán C. A systematic review of provenance systems. Knowledge and Information Systems, 2018, 57(3): 495–543. [doi: 10.1007/s10115-018-1164-3]

26 Han RZ, Byna S, Tang HJ, *et al*. PROV-IO: An I/O-centric

provenance framework for scientific data on HPC systems. Proceedings of the 31st International Symposium on High-Performance Parallel and Distributed Computing. Minneapolis: ACM, 2022. 213–226.

27 Niu X, Kapoor R, Glavic B, *et al*. Interoperability for provenance-aware databases using PROV and JSON. Proceedings of the 7th USENIX Conference on Theory and Practice of Provenance. Edinburgh: USENIX Association, 2015. 6.

28 Gao YZ, Chen XY, Du XH. A big data provenance model for data security supervision based on PROV-DM model. IEEE Access, 2020, 8: 38742–38752. [doi: 10.1109/access.2020.2975820]

29 Celik Y, Petri I, Barati M. Blockchain supported BIM data provenance for construction projects. Computers in Industry, 2023, 144: 103768. [doi: 10.1016/j.compind.2022.103768]

30 Walchshofer C, Hinterreiter A, Xu K, *et al*. Provectories: Embedding-based analysis of interaction provenance data. IEEE Transactions on Visualization and Computer Graphics, 2023, 29(12): 4816–4831. [doi: 10.1109/tvcg.2021.3135697]

31 杨斐斐, 沈思妤, 申德荣, 等. 面向数据融合的多粒度数据溯源方法. 计算机科学, 2022, 49(5): 120–128. [doi: 10.11896/jsjkx.210300092]

(校对责编: 张重毅)