

一种安全可控的数据协同开发环境研究与实现



杜文杰^{1,*}, 王欣¹, 龚君², 史婷婷¹

¹北京计算机技术及应用研究所, 北京 100854

²中国人民解放军 63921 部队, 北京 100094

摘要: 针对多源异构、跨域协同数据管理困难、共享困难、数据确权困难等问题, 提出了一种构建多方、加密、协同、受控的沙箱式数据协同开发环境, 研究了沙箱式的开发运行环境构建、销毁, 云服务编排以及多种运行资源调度方法, 构建支持将业务系统中需要的具体数据资源、算法模型动态加载到一个安全可控、虚拟的运行环境。支持可视化操作, 支持不同的操作系统, 支持按需加载模型以及权限内数据, 确保沙箱内数据资源安全可控。沙箱是一种隔离的安全环境, 在其中运行的数据资源通常具有安全可控的特点, 沙箱隔离包含进程隔离和文件系统隔离。最后, 本文试图在应用开发环境中, 建立一套数据资源贡献评价模型, 推动数据可持续共享共用。

关键词: 数据协同; 应用环境; 安全可控; 沙箱

DOI: [10.57237/j.cst.2025.01.002](https://doi.org/10.57237/j.cst.2025.01.002)

Research and Implementation of a Secure and Controllable Data Collaborative Development Environment

Du Wenjie^{1,*}, Xin Wang¹, Gong Jun², Tingting Shi¹

¹Beijing Institute of Computer Technology and Applications, Beijing 100854, China

²63921 Department of the PLA, Beijing 100094, China

Abstract: Aiming at the problems of data management, sharing and data right confirmation, this paper proposes a multi-party, encrypted, collaborative and controlled sandbox data collaborative development environment. It studies the sandbox development and operation environment construction, destruction, cloud service orchestration and various operation resource scheduling methods. And the specific data resources and algorithm models required in the business system are dynamically loaded into a safe, controllable and virtual operation environment. The environment supports visual operation, supports different operating systems, supports on-demand loading of models and data within permissions, and ensures that data resources in the sandbox are safe and controllable. At the same time, this paper establishes a set of data resource contribution evaluation model in the application development environment to promote the sustainable sharing and sharing of data.

*通信作者: 杜文杰, wenjiedu@126.com

Keywords: Data Collaboration; Application Environment; Secure and Controllable; Sandbox

1 引言

当前,随着云计算、大数据等技术的发展,数据的规模、活性以及对数据的统一运用能力逐渐成为国家层面竞争力的体现。数据产品研发过程中需要用到各种类型、各种形式的模型和数据资源,而各行业沉淀的数据资源、算法模型都是相关领域具有核心价值的战略资产,但这些数据、模型都分散于不同单位、不同组织甚至不同部门中,外单位无法轻易获取。另一方面,由于数据资源以及模型算法容易拷贝的特点,使得保证独有数据资源在共享、流转和处理过程中不被泄露,成为了数据产品多方合作研发的另一痛点[1-3]。

在多方协同合作开展数据产品研发过程中,目前无法保证数据资源的绝对安全,现阶段各方最担心的是交出数据资源之后怎样保证自己数据资源的安全[4, 5]。为了解决上述问题,一是需要一个安全、受控的运行环境;二是运行环境中的数据不可被以任何形式带走;三是运行后的结果应能体现出自己的数据资源的实际使用情况,包括调用次数、用量、占比等等。

综上所述,构建安全可控的支持多方数据资源交换和数据确权的协同开发与应用的生态环境,支撑多方协同开展数据产品开发与共享应用,是数据产品研发过程中亟需解决的问题。本文通过构建一种“开箱即用”的数据产品协同开发环境,支持数据产品由沙箱生产后,依据数据产品使用方的需要,将数据产品直接在服务器端进行在线运行调用,降低数据产品使用的“门槛”。在对数据资源严格保护的前提下,实现各类资源的快速获取和整合利用,又能保证各类资源的私密性与安全性。

2 相关工作

数据协同开发环境的理论基础是访问控制技术[6],通过为用户提供系统资源最大限度共享的基础上,对用户的访问权限进行管理,防止对数据信息的越权篡改和滥用。目前,全球数据协同开发环境技术研究的竞争也日趋激烈,国外针对数据沙箱的研究成果如下。

Google 分析沙箱 Analytics [7, 8]通过提供按需/随时可用的环境实现数据产品的协同开发,该环境使得

数据分析师无需大量的项目经验就可以快速深入并处理大量数据并为解决方案提供原型。

IBM Netezza 1000 [9, 10]是一个独立的数据集市,该环境由大量并行中央处理器、高端内存、大容量存储和 I/O 容量组成。通常将数据仓库中的数据实验和生产数据库环境分开,能够动态的调节、使用数据仓库的容量。IBM Smart Analytics System 是企业数据仓库中的逻辑分区的一个示例,该逻辑分区可以被用作数据沙箱平台。

DataWalk [11]沙箱可以轻松地进行实验,在可用数据和想探索解决的问题的情况下进行数据产品开发。无需锁定解决方案和一组特定的数据源、规则、分析或方法,而是提供了轻松测试,调整和不断改进分析的灵活性。

中国针对数据协同开发环境的沙箱研究主要在数据资源共享的层面。代表性的研究成果包括华为的 AnyOffice 安全沙箱[12]和 MesaTEE 安全计算平台[13]。

AnyOffice 可以提供安全沙箱能力,通过安全沙箱实现个人和企业数据的隔离[14, 15],对企业数据采用高强度加密手段防护,能做到即使数据被非法获取也无法获取明文信息。该沙箱同时给上层企业应用提供透明的安全文件加解密接口,上层应用通过调用接口,可以从加密沙箱中读取数据或者向加密沙箱中写入数据。

MesaTEE 安全计算平台是强安全、高性能、易扩展的芯片级数据安全计算解决方案。协助机构之间解决数据合作过程中数据安全和隐私问题,打破数据孤岛,实现“数据可用不可见”的目标。

上述研究成果解决了数据产品研发过程的“可用不可见”问题,但由于实际应用中数据产权受到严格保护,资源无法轻易共享,会面临着资源利用率低、购置成本过高、资源质量与新鲜度难以保证等问题。本文在上述研究的基础上提出的数据协同开发环境,可以有效解决数据产品无法协同开发的问题,构建了“开箱即用”的数据产品在线运行环境。

3 总体设计

协同开发环境从数据产品开发的全周期维度重新定义标准化的数据产品开发方式，破除数据产品开发的协同壁垒，定义一种“沙箱化、协同化”的数据产品开发方式。数据协同开发环境为数据、算法模型产品研发

发任务提供标准、安全可控、高效便捷的协同开发环境，总体架构如下图 1 所示。通过云服务编排引擎以及多种云服务资源调度算法，按照任务需求方的指令进行协同开发环境的构建、使用以及销毁等操作，同时，通过日志记录相关操作，根据贡献评价模型，对数据产品进行确权。

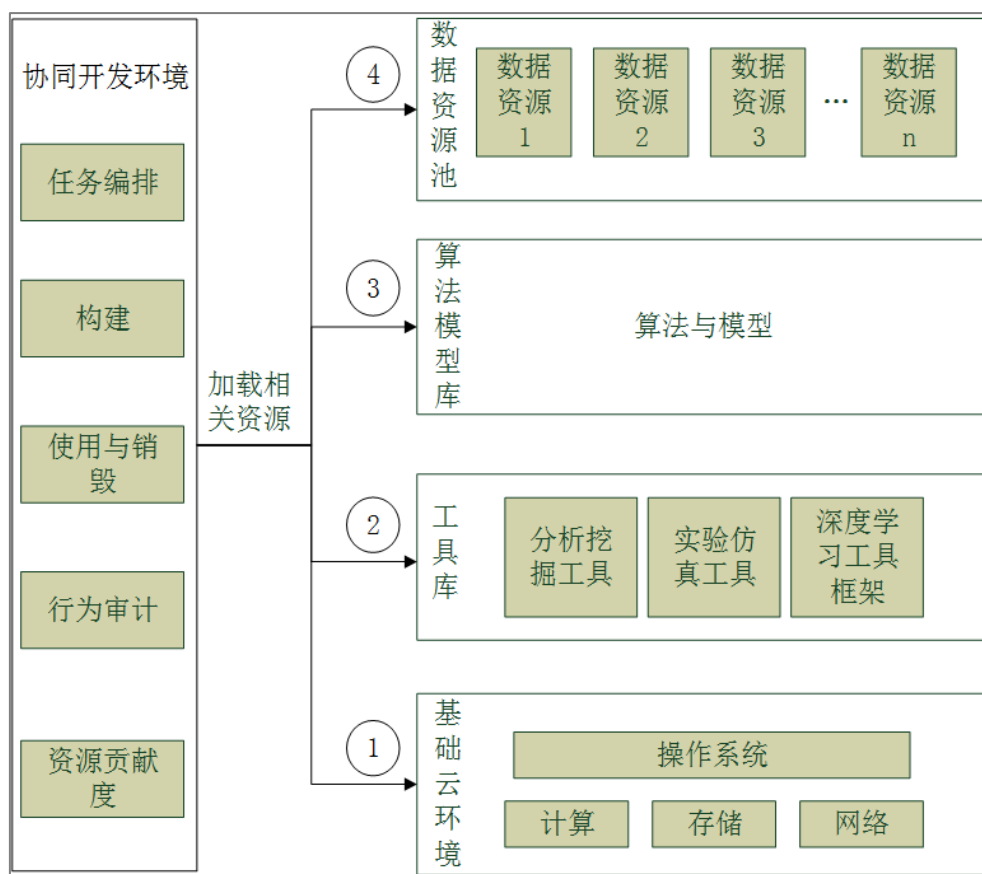


图 1 协同开发环境总体架构

4 功能模块

4.1 任务编排

协同开发环境任务编排模块基于云服务实现任务的快速编排和交付，驱动各类型数据服务、数据库服务和中间件服务的交互过程。任务编排模块由基于消息服务实现任务的分发和远程执行，远程执行器只需要实现标准交互协议即可简单接入，具备跨语言，跨架构，跨平台的特点。

任务编排模块由消息发布流程引擎、消息中间件和执行器组成。消息发布引擎发布的消息经由消息中间件进行缓存，之后由执行器完成指令下发、文件操作和函数服务等工作。在协同开发环境任务编排过程中，提供同步执行器和异步执行器两种任务编排方式，保证关键资源的使用效率，减少由于数据传输带来的资源消耗以及不安全因素。

执行器分为两类，同步执行器执行同步任务，协同开发环境任务的创建、销毁、取消和回滚等流程都是通过同步任务进行触发，时序图如下所示。

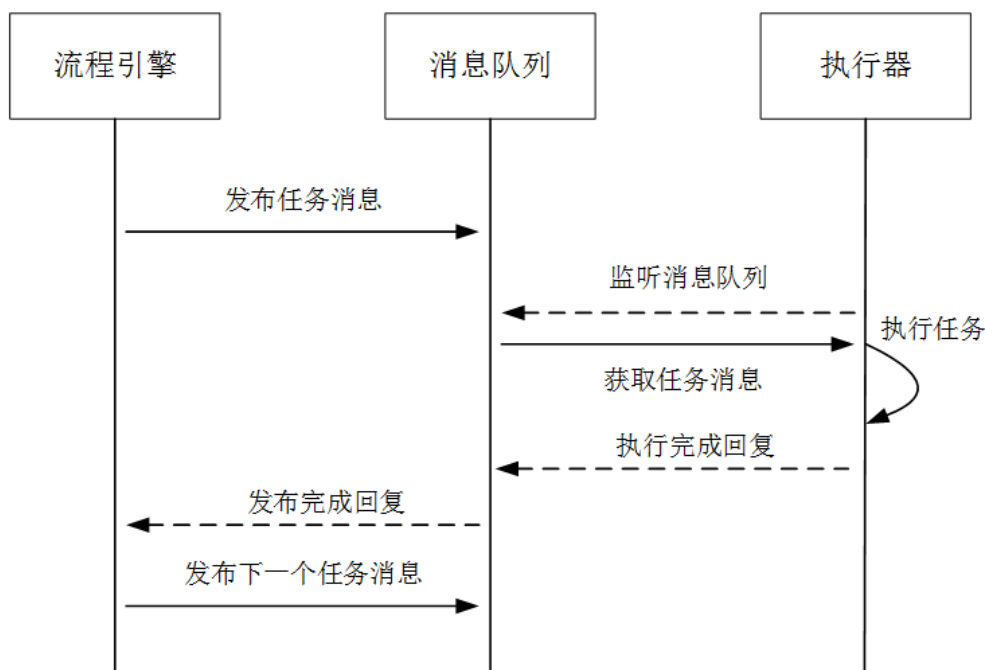


图2 同步执行器调用方式时序图

异步执行器用来执行耗时较长的远程、异步任务。只有当驱动器监视器发布执行完成回复时，才发布下一条任务消息，时序图如下所示。

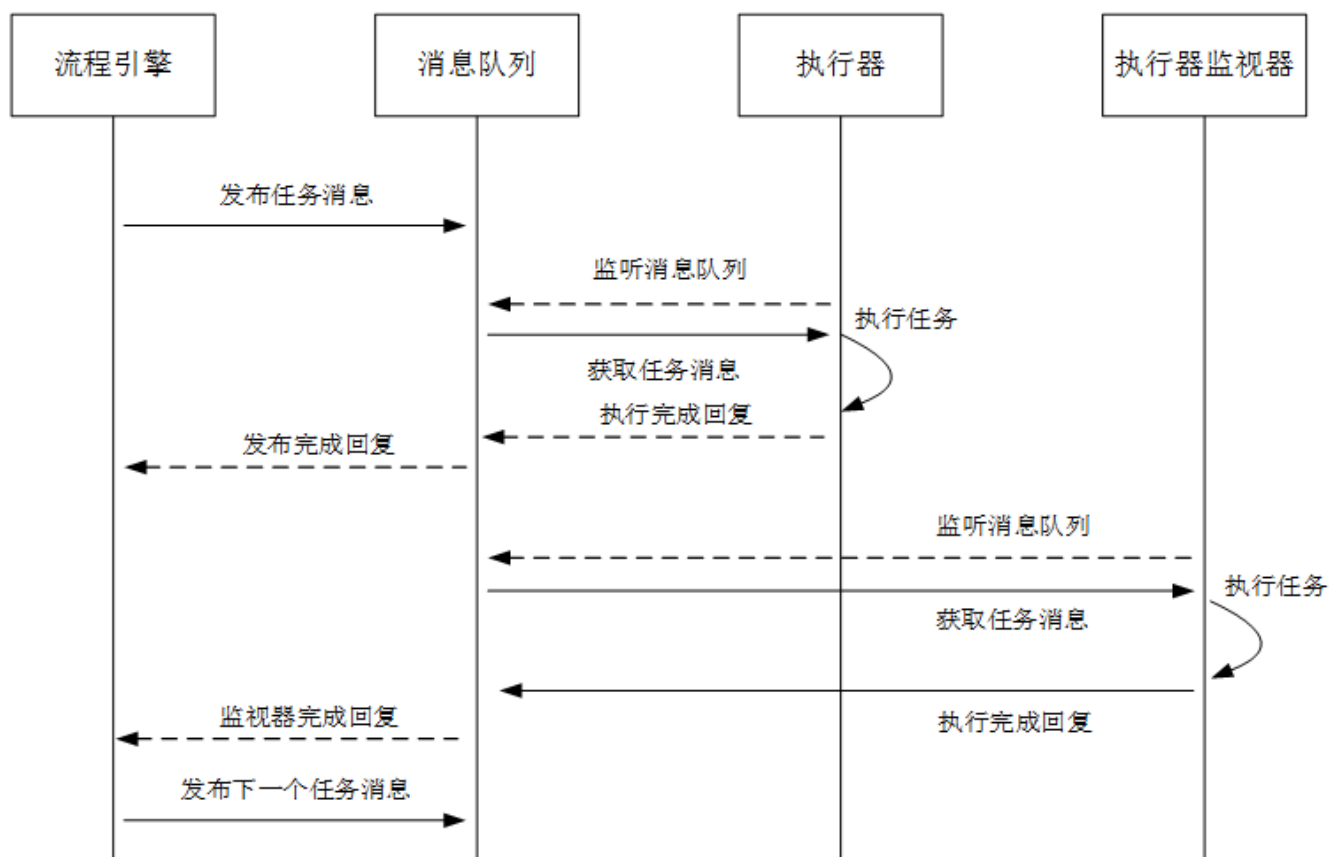


图3 异步执行器调用方式时序图

4.2 开发环境构建

协同开发环境构建模块主要研究面向多方、加密、受控的沙箱式应用平台环境构建技术，将应用任务中需要的具体资源动态加载并创建一个安全的、虚拟的云平台运行环境进行数据产品的集中开发利用。同时实现在沙箱环境下进行跨域协同应用开发，有效解决多方资源安全加密协同应用的问题，保证各类资源的快速获取和整合利用。支持根据数据资源列表，自动化部署需要的工具软件、开发环境等，以及自动加载需要用到的模型、算法、资源等内容，形成算力与算料均满足数据资源任务要求的协同开发环境。

协同开发环境构建过程能够为用户提供诸如指令下发、软件安装、运维流程、IT 策略以及应用的自动化构建及运维。如下图 4 所示，用户 B 可以将应用、服务、软件先做成模板，将步骤繁琐、脚本复杂的应用服务通过客户端同步到协同开发环境。用户 A 不需要每次部署应用都重复执行复杂的步骤，只要在界面轻松地选择方案，通过下载的方式就可以轻松快捷部署出定制好的应用、服务、软件，而且可以避免出错，可以无数次重复使用。

用户自动化部署声明每个应用程序应包含不同的

版本，每个版本可以对应多种部署的架构，每种架构对应一份部署方案。通过部署方案，声明应用部署行为中的资源操作，指令操作，文件操作，监控配置等行为，通过流程引擎进行自动化的驱动。

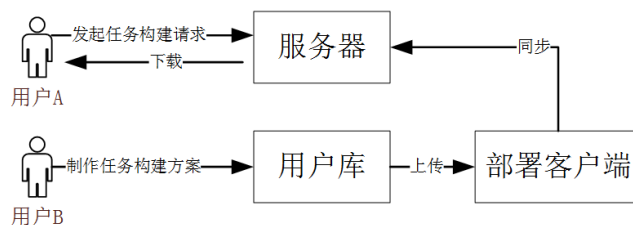


图 4 用户服务构建流程图

4.3 开发环境使用与销毁

协同开发环境使用过程中提供“分布式远程调用+集中本地式调用”两种数据资源调用模式。在数据资源调用过程中，两种调用模式可以综合考虑应用场景、网络状况、时延要求、及时性要求、数据交互量等要求，选择相应的数据资源调用操作，保证关键资源的使用效率，减少由于数据传输带来的资源消耗以及不安全因素。协同开发环境数据使用、销毁和结果导出的流程如下图 5 所示。

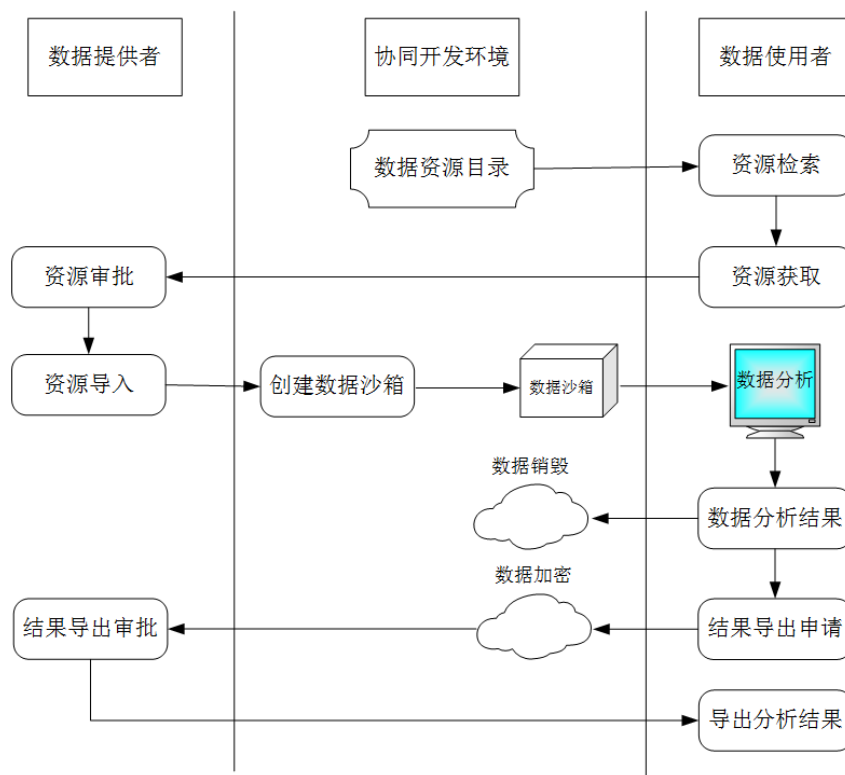


图 5 协同开发环境使用、销毁与导出架构图

协同开发环境为数据使用者提供基于数据资源目录的数据检索方式，经过数据提供者审批后，导入数据资源并创建数据沙箱。操作系统、数据资源、软件资源经过任务编排和自动化部署后，数据使用者只需通过终端电脑的浏览器即可直接连接到协同开发环境。协同开发运行环境中的各类资源开箱即用，用户可对数据资源进行解密，利用平台内的各类软件资源，包括模型、算法、仿真软件、分析软件等，对数据资源进行分析应用。

数据使用者对数据分析应用完毕后，可以选中指定任务点击销毁，进入销毁流程。确保数据产品研发任务运行完毕后，将完全“销毁”，除需要拷出的运行结果可经审批后拷出到指定加密存储区域外，保证任务所使用的一切模型、规则、算法、数据等内容一并被删除。通过与任务脱钩，保证数据资源可用不可见，不留存任何资源应用任务的数据与记录，严格保证资源的安全性。

针对数据使用者需要导出的数据分析成果，协同开发环境会加密保存至资源共享平台中，无法直接导出。数据使用者可向协同开发环境提出分析结果导出申请，经数据提供者审批同意后才能导出成果，避免信息泄露。数据使用者发起成果导出申请后，数据提供者可查看到使用者的信息以及使用记录，包括使用了资源的哪些部分，使用了哪些其他资源等，确定导出不会造成敏感信息泄露后即可同意申请。

4.4 行为日志采集与存储

协同开发环境行为审计通过识别并记录全部操作记录，确保多源异构的数据、模型、算法、工具在安全可控保障下使用。行为审计过程通过日志采集实现，主要针对协同开发环境数据资源管理、协同开发环境任务编排、构建、使用和销毁等全流程、全生命周期的实时行为进行日志采集。如下图 6 所示，协同开发环境将采集的数据实时发送至分布式消息队列，消息队列起到数据缓存和“削峰平谷”的作用，以减少因瞬时日志产生过多产生的传输压力。之后通过分布式消息队列将采集的实时日志定期持久化至分布式并行数据库，并基于协同开发环境的行为日志实现日志审计、日志分析、确权量化等功能。

协同开发环境行为审计通过识别并记录全部操作记录，确保多源异构的数据、模型、算法、工具在安

全可控保障下使用。行为审计过程通过日志采集实现，主要针对协同开发环境数据资源管理、协同开发环境任务编排、构建、使用和销毁等全流程、全生命周期的实时行为进行日志采集。如下图 6 所示，协同开发环境将采集的数据实时发送至分布式消息队列，消息队列起到数据缓存和“削峰平谷”的作用，以减少因瞬时日志产生过多产生的传输压力。之后通过分布式消息队列将采集的实时日志定期持久化至分布式并行数据库，并基于协同开发环境的行为日志实现日志审计、日志分析、确权量化等功能。

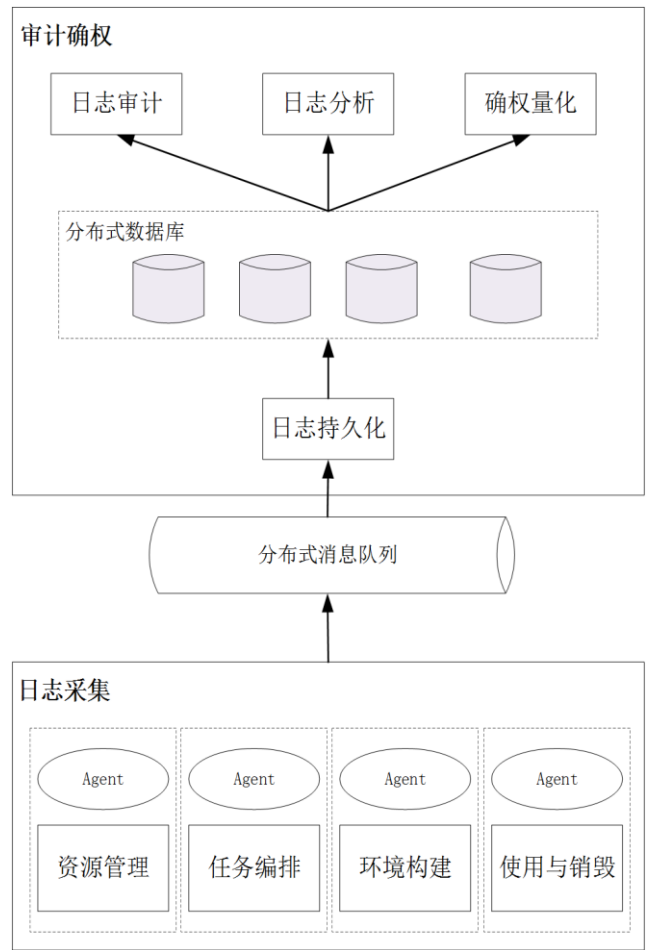


图 6 协同开发环境行为审计与确权

协同开发环境行为日志使用日志 agent 的方式采集，日志 agent 部署在协同开发环境内部和数据提供者、数据使用者的终端节点。对于协同开发环境内部和各终端节点的日志采集方式，通过对数据、算法、软件的操作日志进行文件监听实现。操作系统在接收到日志文件消息后，将产生的各类型日志包装成消息（msg），该消

息结构包含消息头和行为日志的描述信息，然后将它投递到相应的进程中，当该进程收到该消息时会查看该msg结构中的消息头，并进行响应向消息队列分发。

协同开发环境行为日志存储分为两部分，一是实时采集的日志使用分布式消息队列，二是历史日志持久化存储，使用分布式并行数据库。分布式消息队列存储时，协同开发环境通过分布式消息队列提供实时日志数据的接入和缓存，保障日志数据的实时流入，并解耦日志数据的生产和处理，提升日志数据的稳定性、可靠性。分布式并行数据库存储时，通过日志持久化模块进行历史审计日志持久化，并存入高性能列式分布式数据库管理系统。这类数据库吞吐量大，查询速度快，稳定性更高，便于历史审计日志的持久化保存，有效保证历史日志的安全性与可靠性。

4.5 数据资源贡献评价

协同开发环境数据资源贡献评价结合数据各质量维度综合分析数据资源，从数据准确性、完整性、一致性、平衡性、时效性、相关性等维度，构建权威、标准的数据共享评价体系。数据使用者在数据资源使用完毕后对资源进行评价，为其他用户使用数据提供参考。评价过程中采取维度赋权模式，将数据资源各维度质量得分进行加权求和，得到最终的数据资源贡献评价分数步骤如下：

(1) 构建主观权重向量

对于 m 个评级维度，根据数据使用者评价方法对各个维度重要性给出标度值，得到各维度主观权重向量 A ，定义如下。

$$A=[A_1,A_2,...,A_m] \quad (1)$$

(2) 构建资源评价得分矩阵

针对 n 个数据源， m 个维度进行评价，构建资源

评价得分矩阵 $R=(r_{ij})_{n \times m}$ ，其中 r_{ij} 为第 i 个数据源的第 j 个维度的评价得分，定义如下。

$$R=\begin{bmatrix} r_{11}r_{12}...r_{1m} \\ r_{21}r_{22}...r_{2m} \\ \cdots \\ r_{n1}r_{n2}...r_{nm} \end{bmatrix} \quad (2)$$

(3) 评价得分矩阵归一

通过对数据使用者的评价得分矩阵进行分析，将矩阵 R 中每一个维度质量分值 r_{ij} 乘以该维度的主管权重 A_j ，并对矩阵 R 进行归一化处理，公式如下：

$$r_{ij}=\frac{r_{ij}}{\sum_{j=1}^m r_{ij}} \quad (3)$$

(4) 维度权重

通过各数据源在第 j 个维度下的熵值 X_j ，公式如下。

$$x_j=-\sum_{i=1}^n r_{ij} \ln r_{ij} \quad (4)$$

则第 j 个维度权重公式如下。

$$w_j=\frac{x_j}{\sum_{j=1}^m x_j} \quad (5)$$

(5) 各来源数据评价分数

依次相加贡献矩阵第 i 个数据源的维度权重，获得各来源数据贡献评价分数如下。

$$Score+=w_j * R_{ij} \quad (6)$$

表 1 数据资源贡献算法

输入：由 n 个数据源， m 个维度得分矩阵 $R_{n \times m}$ ，质量维度主观权重向量 A
输出：多源数据贡献评分结果 $Score_n$
1: 按权重循环构建贡献评分矩阵
2: 归一贡献评分矩阵 公式(3)
3: 计算 j 维度下熵值 公式(4)
4: 计算各质量维度的客观权重向量 公式(5)
5: 计算各个来源数据贡献分数 公式(6)

通过上述数据资源贡献评价算法，构建覆盖数据准确性、完整性、一致性、平衡性、时效性、相关性等维

度的数据评价评估运营指标体系，推动数据共享可持续发展。全部资源评价结果均采用安全存储技术进行存储，

保证资源评结果的私密性与安全性。通过资源服务评价系统，促进数据提供者进一步提升资源质量，也为数据使用者提供一定程度上的选择参考。

5 典型场景实验和评估

本文在理论研究的基础上，开发了数据协同开发环境的原型系统，并在实验室环境下搭建了典型应用场景，实现协同开发环境的任务编排、构建、使用与销毁、行为审计和贡献度评价的基本功能，达到了数

据产品开箱即用的运行和使用目标。

5.1 实验环境搭建

协同开发环境原型系统基础运行环境支持主流 CPU 架构、支持多种虚拟化技术；面向用户提供多编程语言、多网关协议和多算法类型的支持。协同开发环境原型系统运行的实验室环境技术指标如下表 1 所示。

表 2 协同开发环境原型系统技术指标

序号	基本指标	详细指标
1	虚拟化技术	hyper-V、Xen、Kvm、ESXi、PowerVM、LXC、Docker 等虚拟化技术
2	编程语言	C、C++、JAVA、Python、matlab 等编程语言
3	基础运行环境	ARM、X86、MIPS 主流 CPU 架构
4	网关协议	HTTP、WebSocket、TCP、UDP、OpenAPI、WebService 等服务协议
5	算法	回归、分类、聚类、神经网络等算法

5.2 实例验证

为了验证协同开环境的功能和效果，在实验室环境下构建基于协同开发环境得数据资源共享交换场景，该场景由三台桌面计算机和一台服务器组成，如图 7 所示。三台桌面计算机为 Windows 运行环境，模拟数据共享交

换的参与方，既可以是数据提供方，也可以是数据使用者；服务器由国产飞腾 2500 处理器搭载银河麒麟运行，模拟协同开发环境的数据沙箱，为参与方提供标准的调用接口，实现协同开发环境的任务编排、环境构建、资源使用与销毁、行为审计和资源贡献评价等功能。

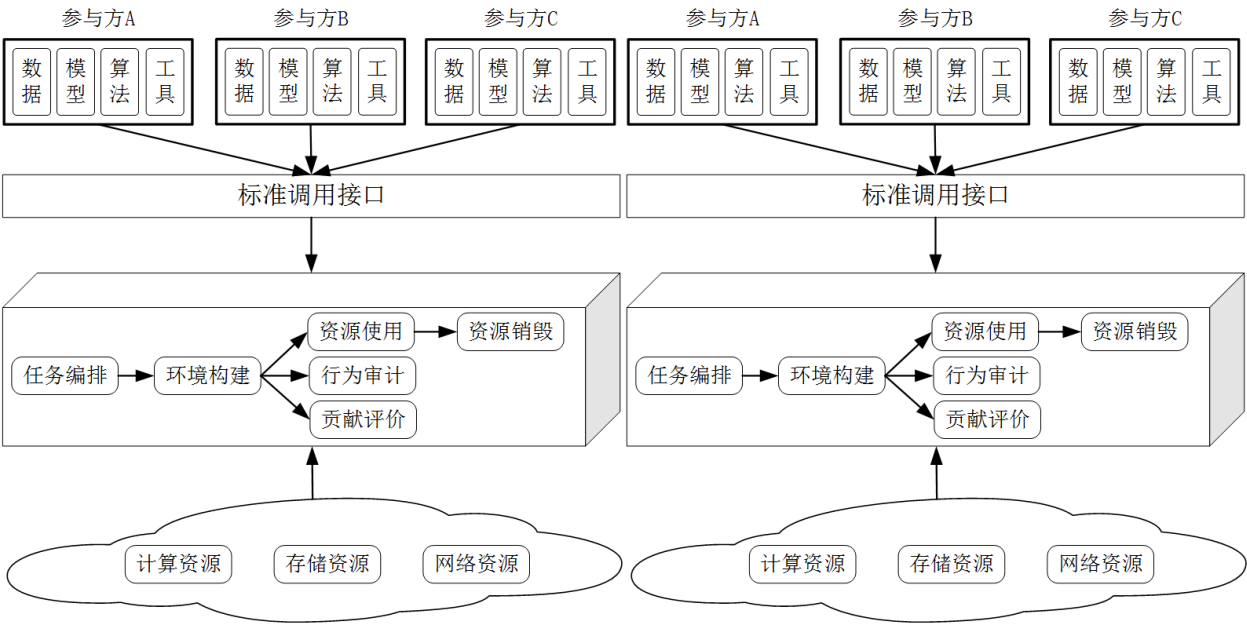


图 7 数据资源共享交换场景

在该场景下，主要验证参与方 A 作为数据提供者，参与方 B、C 作为数据使用者的数据资源共享交换过程，

验证流程如下图 8 所示。其中 B 完成任务编排、环境构建、资源使用与销毁等功能，C 完成任务编排、环境

构建、资源使用与销毁、资源导出等功能。在数据资源共享过程中,由协同开发环境对 B、C 发起行为监控,确保数据资源安全可靠的流转。在数据使用完毕后,由服务器向 B、C 发起数据贡献评价打分,最终根据本文提供算法,将数据资源贡献评价反馈给 A。

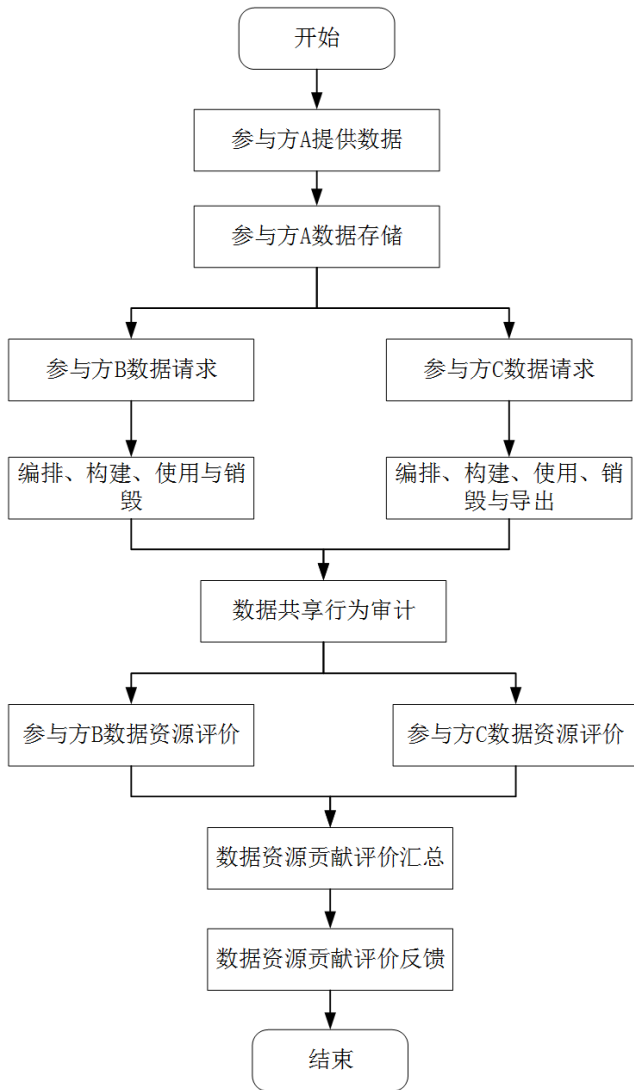


图8 数据资源共享交换流程

5.3 实验评估

经实验验证,该原型系统运行符合数据协同开发环境的设计预期。通过“统一部署开箱即用”的方式,将多方的数据资源通过加密的方式接入沙箱,快速构建一个安全的、隔离的、可信的、受保护的“沙箱式”数据资源存储与协同开发执行环境。协同开环环境中涉及参与方的任务(流程)、数据、算法逻辑、模型、工具等资源被投入多方安全存储执行环境,保证该环

境中的一切资源不可获取和篡改,即“可用不可见”。在数据产品开发任务时限和沙箱环境内,共同拥有各方数据的使用权,保证参与数据产品开发的各方确权清晰,数据产品开发任务完成后销毁多方安全环境内部一切资源,严格保护数据安全。

6 结论

本文针对数据资源共享困难和数据资源确权困难的问题,依据数据沙箱的设计特点,在现有中国外数据沙箱成果的基础上,提出了一种安全可控的数据协同开发环境。详细设计了协同开发环境的任务编排、构建、使用与销毁、行为审计,贡献度评价等模块,并开发了原型系统并在实验室环境下进行了验证。

参考文献

- [1] 陈欣华. 浏览器安全机制研究 [J]. 中国管理信息化, 2020, 23(19): 186-187.
- [2] 张晓兵, 王妍, 吕道健, 田亮, 孟繁盛. 信创体系下的浏览器安全可信架构研究 [J]. 信息安全研究, 2021, 7(04): 328-334.
- [3] 王丹娜, 李加赞: 聚焦安全可信数字化底座 支撑落实关键信息基础设施保护制度 [J]. 中国信息安全, 2021(11): 22-25.
- [4] 鲁振, 胡坚升, 李名扬. 一种多架构应用软件开发及运行方法研究与实现 [J]. 信息技术与网络安全, 2020, 39(09): 28-33. <https://doi.org/10.19358/j.issn.2096-5133.2020.09.006>
- [5] 任望, 王永涛, 程岩. “新基建”新安全 新思考 [J]. 中国信息安全, 2020(05): 46-47.
- [6] 赵旭, 陈丹敏, 颜学雄, 王清贤. 沙箱技术研究综述 [J]. 中原工学院学报, 2014, 25(04): 1-5.
- [7] Plaza B. Google Analytics for measuring website performance [J]. Tourism Management, 2011, 32(3): 477-481.
- [8] Turner S J. Website statistics 2.0: Using Google Analytics to measure library website effectiveness [J]. Technical Services Quarterly, 2010, 27(3): 261-278.
- [9] Ferguson M. Architecting a big data platform for analytics [J]. A Whitepaper prepared for IBM, 2012, 30.
- [10] Martin D, Ivanova I, Mueller R, et al. Demonstrating near real-time analytics with IBM DB2 analytics accelerator [J]. Datenbanksysteme Business, Technologie und Web (BTW) 2042, 2013.

- [11] Cogger S. Doing the Data Walk [J]. The Science Teacher, 2015, 82(2): 43.
- [12] 华为技术有限公司. AnyOffice V200R005C00 安全沙箱技术白皮书 [R/OL].
- [13] 赵波, 袁安琪, 安杨. SGX 在可信计算中的应用分析 [J]. 网络与信息安全学报, 7(6): 126-142.
- [14] 林晓拔, 沙箱隔离安全技术的研究思想 [J]. 计算机工程与设计, 2018, 4(39): 94.
- [15] 王辉, 王凌云, 吴震, 面向沙箱隔离技术安全性分析的研究 [J]. 科技世界, 2008, 41(9): 145-149.