业务系统采集生成的多种格式非结构化数据等.

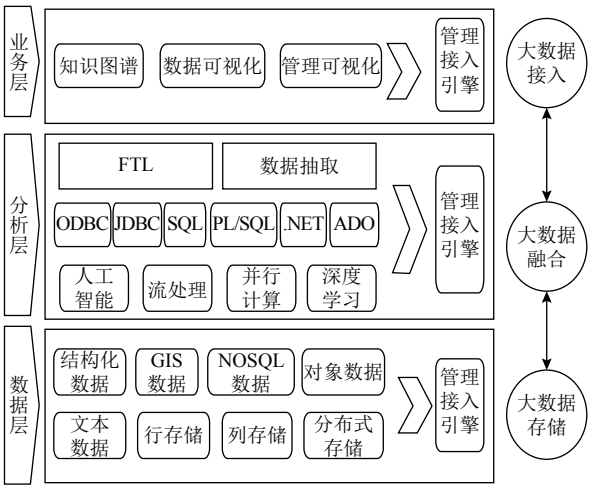

图 1 大数据管理平台架构

(1) 结构化数据的行式存储和列式存储

使用最广的数据存储方式是行式存储, 把一行数据作为一个整体来存储, 但行式存储在维护大量的索引和物化视图场景下, 在处理时间和存储空间方面成本过高. 列式存储数据库以列为单位进行数据存储, 每一列单独存放, 并由一个线程来处理, 这样既可以充分利用处理器的多核心特性, 又能够大大降低系统 I/O 开销, 因此我们采用擅长随机读操作的行式数据库与擅长条件查询的列式数据库相结合的方式, 来管理结构化数据.

(2) 非结构化数据的分布式存储和弹性扩展

非结构化数据需要分布式存储, 并且保证按需的弹性扩展功能. 平台的分布式存储充分利用 HDFS 的低成本、高容错、高吞吐特性来管理数据, 经由并行数据路径完成与 MPP 数据库服务器的数据交换, 通过弹性控制管理模块联动数据协调分发模块提供数据的弹性扩展管理, 参见图 2.

对于弹性扩展在弹性控制管理模块中采用特定语言进行描述, 通过描述中的内容进行灵活的扩展, 例如, 描述一个扩展节点, 包括硬件、软件特征和配置必须明确规定, 并以特定的方式进行表述, 再使用自动化任务解析、执行这些相关的描述文档, 从而实现相应扩展功能.

(3) 支持处理的数据类型

平台支持对常用的所有数据类型进行处理, 包括:

1) 关系数据: 支持关系数据的各种数值类型、字符类型、二进制数据类型、日期时间类型、布尔类型等.

2) 空间数据: 支持几何特征和离散特点的地理要素, 即空间对象数据, 如点、线、面、体等对象的数据组件, 以及 GIS 栅格、图层、坐标等数据存取.

3) NoSQL 数据: 支持 NoSQL 数据类型、位串类型、数组类型、复合类型等.

4) 文本数据: 支持常用的文本数据, 包括日志数据、文章数据、网页数据等.

平台对数据的管理都采用图形化界面进行操作, 例如对 NoSQL 数据的管理已实现如图 3 的界面.

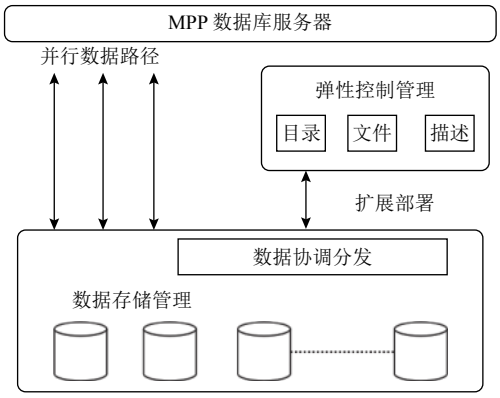

图 2 弹性扩展

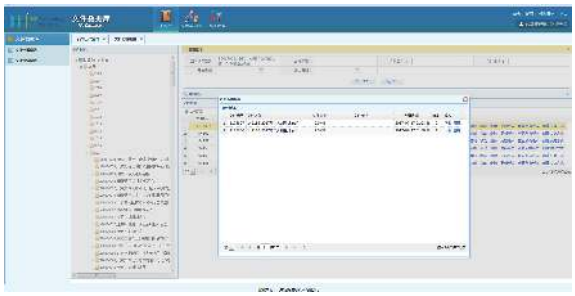

图 3 NoSQL 数据管理界面

3.1.2 分析层

分析层对数据进行融合处理, 是一种针对环保检测和评估数据的容量大、种类多、增长速度快、价值大等特征的集成技术, 包括: 流处理技术、大规模并行处理技术、机器学习技术、并行算法等.

平台通过增加并行度确保使用整个集群的资源, 而不是把任务集中在几个特定的节点上. 对于包含 Apache Spark Shuffle 的操作, 增加其并行度以确保更为充分地使用集群资源; 同时, 流处理默认将接收到的数据序列化后存储, 以减少内存的使用, 但是序列化和反序列化需要更多的处理器资源, 因此优化的序列化

方式和自定义的序列化接口可以更高效地使用处理器资源, 参见图 4.

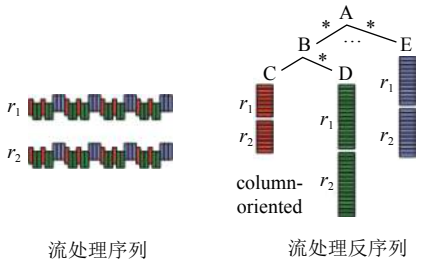

图 4 流处理序列和反序列

在流处理中, 任务之间有可能存在依赖关系, 后面的任务必须确保前面的作业执行结束后才能提交, 通常情况下分析型数据库框架能够高效地确保任务及时分发. 但是, 如果前面的任务执行的时间超出了批处理时间间隔, 那么后面的任务就无法按时提交, 这样就会进一步拖延接下来的任务, 造成后续任务的阻塞, 因此分析层会设置一个合理的批处理间隔以确保作业能够在这个批处理间隔内结束; 同样, 当批处理间隔非常小 (小于 500 毫秒) 时, 提交和分发任务的延迟就变得不可接受了, 通过经验对比, 我们采用 Spark 的 Standalone 和 Coarse-grained Mesos 模式减少因任务提交和分发所带来的延迟.

对于数据的底层模型设计, 因需要进行基于多维模型的交叉分析来有效发现问题, 所以数据的维度越丰富所能实现的交叉也越丰富和灵活; 但相应的, 如果要尽可能地丰富各维度的交叉分析, 对基层模型的要求也就越高. 因此, 我们引用数据立方体来实现模型设计, 参见图 5.

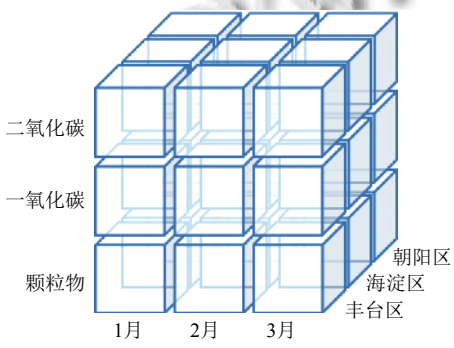

图 5 数据立方体示例

用数据立方体来拓展数据细节有两种方向, 一类是纵深拓展, 也就是基于一个维度的细分, 例如一个月细分到每一天, 一条记录将会被拓展成 30 条; 另一类是横向拓展、多个维度的交叉, 就像立方体中添加了空气污染物维和区域维. 这样存储的数据就从原本单一的时间维度扩展成了时间、污染物和区域三个维度, 也就是三维立方体所能展现的形式, 而且维度可以继续扩展, 四个、五个直到数十个, 理论上都是可行的. 以三个维度进行举例: 对于数据存储而言, 横向的拓展与纵深拓展的影响是一样的, 记录数都是以倍乘的方式增长, 假设有 20 个污染物大类, 再加上十六个区, 那么经过纵深和横向拓展之后, 原先每月的 1 条记录就变成了: $1 \times 30 \times 20 \times 16 = 9600$ (条).

在功能实现方面, 经过数据的多维分析后, 平台在数据准备区进行 ETL 处理, 数据经过抽取、转换后加载到数据仓库中, 分析完主题和数据元后建立数据模型 (概念模型、逻辑模型、物理模型) 并形成事实表和纬度表, 然后通过粒度分析将历史记录先抽取整合, 最后再根据决策者可能用到的数据集合分解成若干记录, 同时利用 OLAP 工具技术进行数据的分析导出, 以供给业务层进行数据可视化处理.

3.1.3 业务层

在业务层, 系统关注将分析层提供的数据进行可视化展现, 其中的重点就是使用知识图谱. 知识图谱基于图的数据结构, 由节点和边组成, 每个节点表示现实世界中存在的具有多种属性的“实体”, 每条边为实体与实体之间的“关系”. 知识图谱把所有不同种类的信息连接在一起而得到一个关系网络, 提供了从“关系”的角度去分析问题的能力, 是关系的最有效的表示方式[8].

基于知识图谱, 我们也尝试提供数据智能搜索服务. 智能搜索的功能类似于知识图谱在互联网搜索引擎上的应用, 也就是说, 对于每一个搜索的关键词, 我们可以通过知识图谱来返回更丰富, 更全面的信息. 比如搜索某个监测点的污染情况, 我们的智能搜索引擎可以返回与这个监测点相关的所有类型的污染记录, 包括水污染、大气污染、土壤污染等, 并同时返回区域涉及的建设项目信息、污染物排放标准等环境保护相关信息, 参见图 6.

另外, 通过可视化技术把复杂的信息以非常直观的方式呈现出来, 参见图 7, 使得我们对隐藏信息的情况也一目了然. 数据可视化是指以柱状图、饼状图、线型图等图形方式展示数据, 让决策者更高效地了解

业务的重要信息和细节层次. 大量实践表明, 人通过图形获取信息的速度比通过阅读文字获取信息的速度要快很多, 因此通过可视化展现配合门户服务, 帮助环保局管理人员实现高效、系统的数据识别和决策.

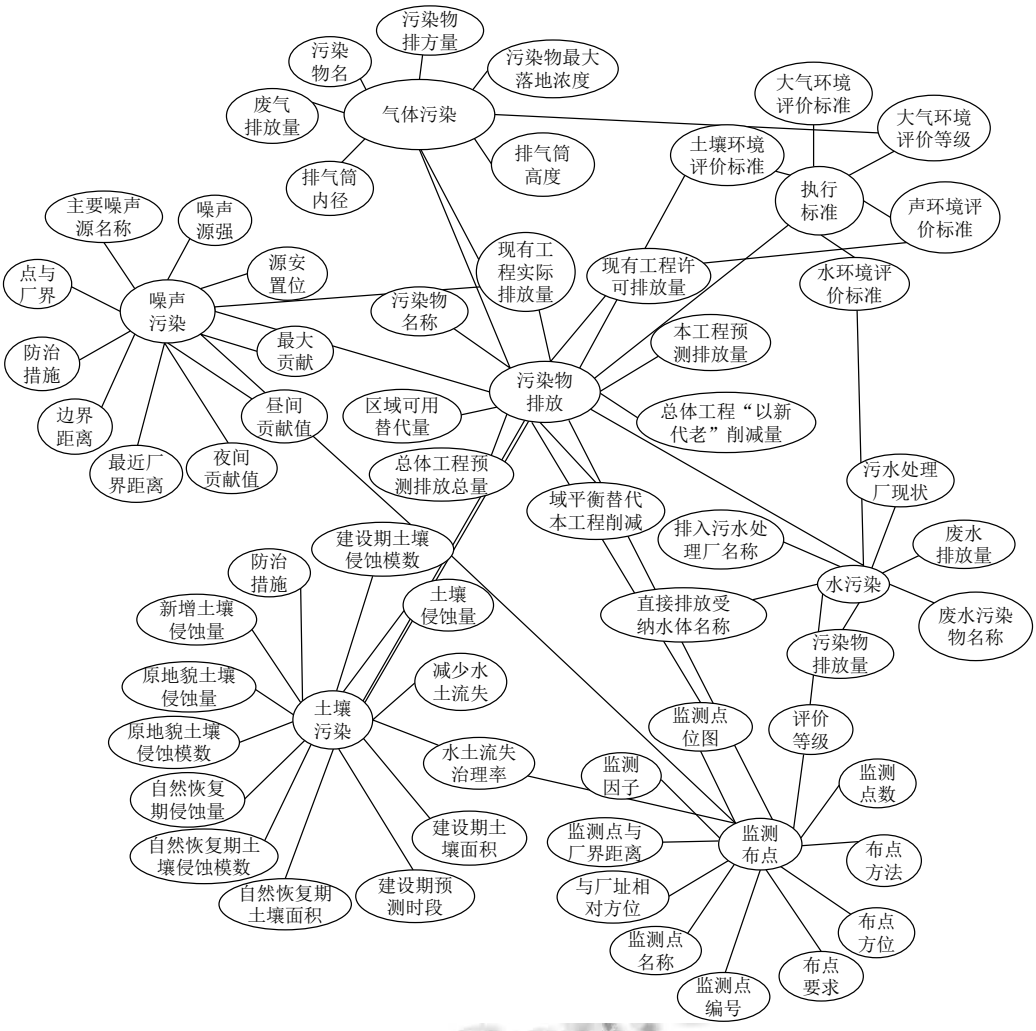

图 6 知识图谱关联

图 7 数据可视化展现示例

3.2 平台物理架构

云基础架构使得计算、存储、网络等可以通过资源池化而按需获得, 我们重点关注的是这些资源的整合以及基于此的动态变化管理策略, 形成一个有机的、可灵活调度和扩展的资源池, 面向大数据管理平台实现自动化的部署、监控、管理和运维.

参见图 8, 我们采用典型的云基础架构融合部署方案. 例如, 通过虚拟防火墙与虚拟机之间的融合, 可以实现虚拟防火墙对虚拟机的感知、关联, 确保虚拟机迁移、新增或减少时, 防火墙策略也能够自动关联. 此外, 虚拟机与负载均衡设备形成联动, 即在业务突发时, 自动按需增加相应数量的虚拟机, 与负载均衡联动实现业务负载分担; 同时, 当业务量减小时, 可以自动减少相应数量的虚拟机, 节省资源. 不仅有效解决虚拟化

环境中面临的负载突变问题, 而且大大提升了业务响应的效率和智能化. 再有, 云基础架构通过虚拟化技术与管理层的融合, 提升了 IT 系统的可靠性. 例如, 虚拟化平台可与网络管理、计算管理、存储管理联动, 当设备出现故障影响虚拟机业务时, 可自动迁移虚拟机, 保障业务正常访问; 对于设备正常、操作系统正常、但某个业务系统无法访问的情况, 虚拟化平台还可以与应用管理联动, 探测应用系统的状态, 例如 Web、应用、数据库等响应速度, 当某个应用无法正常提供访问时, 自动重启虚拟机, 恢复业务正常访问.

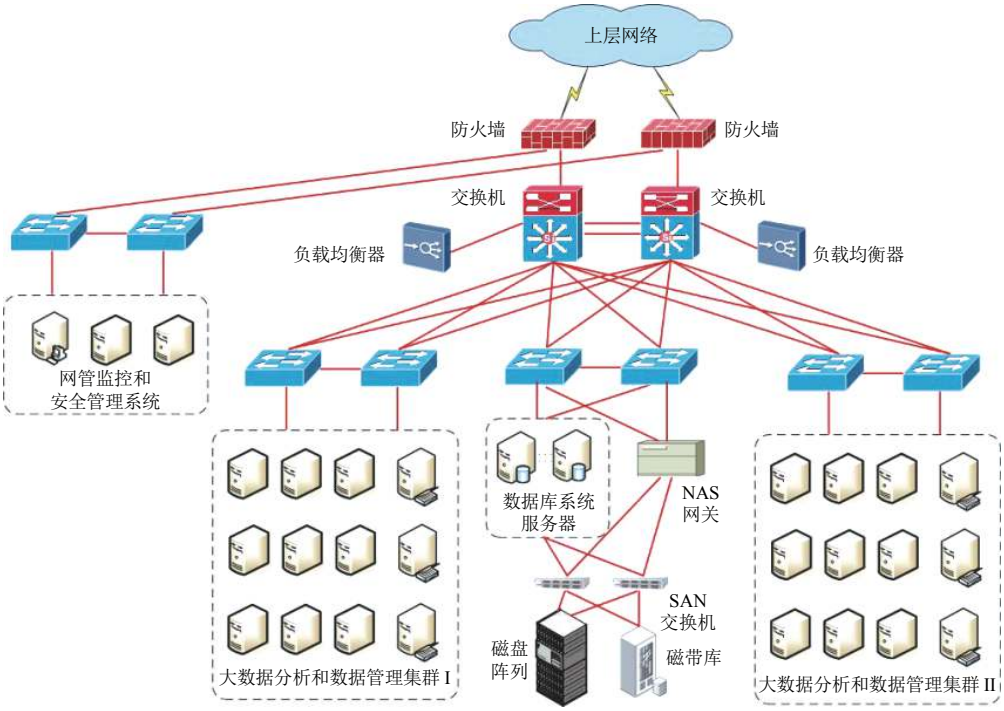

图 8 云基础架构融合部署

4 结语

本文对环境评估大数据管理平台涉及的关键技术和平台逻辑架构、物理架构设计进行阐述, 该平台是行业数据和数据库技术相结合的系统工程, 以大数据技术为支撑, 通过弹性扩展、流处理、数据湖、并行处理、机器学习等技术为手段, 不断结合环境监测与评估数据的需求分析调整技术方法, 实现环境监测和软件工程的软着陆, 为开展生态环境综合决策、环境监管和公共服务提供基础数据支撑, 为生态环境管理和决策提供服务.

参考文献

1 环境保护部办公厅. 关于印发《生态环境大数据建设总体方案》的通知. http://www.cac.gov.cn/2016-03/18/c_1118376330.htm. [2016-03-08]

2 Yang CW, Huang QY, Li ZL, *et al*. Big data and cloud computing: Innovation opportunities and challenges. International Journal of Digital Earth, 2017, 10(1): 13–53. [doi: 10.1080/17538947.2016.1239771]

3 陈付梅, 韩德志, 毕坤, 等. 大数据环境下的分布式数据流处理关键技术探析. 计算机应用, 2017, 37(3): 620–627.

4 Wikimedia Foundation, Inc. Data lake. https://en.wikipedia.org/wiki/Data_lake. [2018-07-16]

5 林荣智. 并行数据库技术分析与展望. 信息通信, 2016, (12): 200–201. [doi: 10.3969/j.issn.1673-1131.2016.12.095]

6 林志, 茆云霞. 深度学习技术在环保督查工作中的应用研究. 信息通信, 2017, (11): 80–82. [doi: 10.3969/j.issn.1673-1131.2017.11.035]

7 林伟声. 深度学习技术在信息系统数据分析中的应用. 电脑与电信, 2017, (6): 51–53.

8 Sivarajah U, Kamal MM, Irani Z, *et al*. Critical analysis of big data challenges and analytical methods. Journal of Business Research, 2017, 70: 263–286. [doi: 10.1016/j.jbusres.2016.08.001]