关联大数据管理技术: 挑战、对策与实践*

沈志宏[1]　姚　畅[2]　侯艳飞[1]　吴林寰[3]　李跃鹏[1]
[1](中国科学院计算机网络信息中心　北京 100190)
[2](国家自然科学基金委员会　北京 100085)
[3](中国科学院微生物研究所　北京 100101)

摘要: 【**目的**】分析关联大数据的概念、内涵与特征, 针对关联大数据管理的技术挑战, 探讨关联大数据管理技术的对策和解决思路。【**方法**】结合 NoSQL 数据管理技术、分布式图计算技术、大数据流水线技术等给出应对挑战的思路, 并基于此思路形成大规模图数据仓库加工系统 gETL。【**结果**】该方法和系统在 NSFC-KBMS 和 WDCM 项目中得到了应用, 实现了大规模知识型数据和生物数据的有效管理, 满足了多元化的数据管理需求。【**局限**】需要结合应用的情况, 进一步完善方法与系统。【**结论**】通过采用 NoSQL 数据存储技术、分布式图计算技术、大数据流水线技术以及 gETL 系统, 可以很好地解决关联大数据的管理问题。

关键词: 关联数据　知识图谱　大数据　关联大数据
分类号: TP393
DOI: 10.11925/infotech.2096-3467.2017.1341

1　关联大数据的概念、内涵与特征

1.1　相关概念

关联数据(Linked Data)的概念由 Berners-Lee 于 2006 年提出[1], 其原理是用一种轻型的、可利用分布数据集及其自主内容格式、基于标准的知识表示与检索协议、可逐步扩展的机制实现可动态关联的知识对象网络, 并支持在此基础上的知识组织和知识发现[2]。2007 年 5 月, W3C 的关联开放数据(Linking Open Data, LOD)运动正式启动, 旨在推动将 Web 上的开放数据源以资源描述框架(Resource Description Framework, RDF)的方式发布, 同时生成数据源之间的 RDF 链接, 以供关联数据浏览器、搜索引擎以及更高级的应用程序使用。由此关联数据的理念逐渐深入到各个领域, 关联数据的应用得以蓬勃发展。

2008 年, Nature 出版专刊 BigData[3], 分析了大量快速涌现的数据给数据分析处理带来的巨大挑战。大数据(Big Data)指传统的数据处理应用软件难以处理的庞大和复杂的数据集[4]。近年来, 数据爆炸式增长催生着大数据时代的来临。一方面, 互联网、移动互联网、物联网、车联网、GPS、生命科学、医学影像、安全监控、金融、电信等领域产生着巨大的数据; 另一方面, 随着越来越多的诸如 500 米口径球面射电望远镜(FAST)、中国散裂中子源(CSNS)等大科学装置的建设和重大科学实验的开展, 以及无所不在的科学传感器和传感器网络广泛应用于天空、陆地和海洋, 对自然环境进行全方位的探测、监测, 源源不断产生的科学数据将科学研究快速推进到一个前所未有的大数据时代[5]。

随着越来越多的数据以关联数据的方式发布, 关联大数据的概念为人提起。Hu 等介绍了一种针对“Big Linked Data”的存储方案, 允许在保持语义特性的同时还能实现大规模数据的存储[6]。Hitzler 等在讨论“关联数据、大数据与第四范式”时认为, 关联数据已经毫

通讯作者: 沈志宏, ORCID: 0000-0002-0113-0478, E-mail: bluejoe@cnic.cn。

*本文系国家重点研发计划云计算和大数据专项“科学大数据管理系统”(项目编号: 2016YFB1000605)和中国科学院计算机网络信息中心与国家自然科学基金委员会合作项目“国家自然科学基金大数据知识管理服务平台”(项目编号: GC-FG4161781)的研究成果之一。

无争议地成为大数据版图中的一部分[7]。文献[8]也认为: 大数据和关联数据将成为未来 Web 基础设施一个必不可少的部分, 大量的数据将变得可用、互联和可标识。Robak 等介绍了如何将大数据和关联数据的概念应用到供应链管理当中的实践[9]。刘炜等认为由于数据量越来越大, 越来越多的关联数据应用系统不得不考虑采用大数据解决方案[10]。由于大数据方案对于海量、高速发展的数据具有很好的管理能力, 因而被用来管理关联数据是一个必然的选择。这种采用大数据技术方案建立的关联数据应用, 可称之为“大”关联数据应用。

本文认为, 关联大数据(Big Linked Data)即体量很大(如: RDF 资源数目达到亿级, RDF 三元组数目达到十亿级)、无法采用传统的方法和系统(如: Jena、Virtuoso[11]、D2R[12]、Silk[13]等)进行管理的关联数据。

1.2 关联大数据的内涵

关联大数据综合了学术和工程实践的丰富内涵。本文认为关联大数据具有两个形态, 即外在的开放数据形态和内在的语义网络形态。外在的开放数据形态关注其开放可获取性, 在技术特征上一般表现由 HTTP、URI 和 SPARQL 所构成的 RDF 数据开放协议和服务; 内在的语义网络形态则关注关联大数据的语义网络结构, 表现为由 RDF 资源和 RDF Link 构成的庞大的关系网络。

与语义网络很相关的另一个概念即知识图谱(Knowledge Graph), 作为当前的研究热点, 知识图谱将互联网的信息表达为更接近人类认知世界的形式, 提供一种更好地组织、管理和理解互联网海量信息的能力[14]。很多知名的知识图谱(如: DBpedia[15]、YAGO[16]、Wikidata[17]等)就采用 RDF 模型作为知识的表达格式。有观点认为知识图谱本质上就是语义网络[18]。

语义网络形态又可以进一步分解成两个要素, 即语义要素和关系网络要素。语义要素表现为 RDF 描述框架、RDF Schema、语义推理等语义规范和内容, 关系网络表现为由实体、关系构成的图(Graph)。关联大数据的概念层次如图 1 所示。

1.3 关联大数据的特征

大数据具有显著的 4V 特征: 体量大(Volume)、增长速度快(Velocity)、多形态(Variety)、高价值(Value)[19-20]。关联大数据作为大数据版图中的一部分,

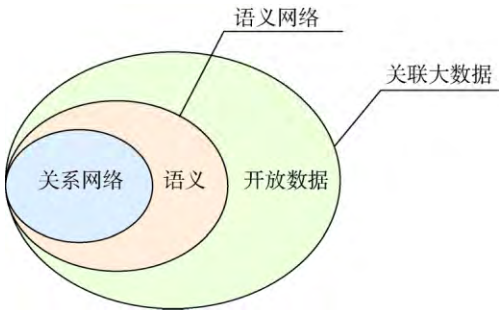

图 1 关联大数据的概念层次

同样表现出体量大和增长速度快的特征。随着数据内容的快速增加, 关联数据的资源(RDF Resource)和关联(RDF Link)往往达到千万级、十亿级乃至百十亿级的规模。以关联开放数据组织 LOD 收录的数据集为例, 2007 年共收录 12 个开放的关联数据集; 2010 年,收录 203 个数据集, 共 3.95 亿条三元组; 2012 年, 收录 295 个数据集, 共 316 亿条三元组; 截至 2017 年 2 月, 数据集的数目已达到 1 163 个, RDF 三元组达到 1 494 亿条[21]。再以世界微生物数据中心 WDCM[22]为例, 共收集了全世界 68 个国家 584 个保藏中心的微生物资源信息, 其构建的 RDF 网络包括海量的物种、蛋白质、基因、酶等信息以及丰富的相关文献信息, 从而从整体的层面对生命进行理解和研究。据统计, WDCM 的 RDF 数据三元组多达 30 亿条。

关联大数据也具有多形态的特征表现, 主要表现为语义 Schema 的多元性。从 LODStats[23]收录的 9 960 个数据集来看, 它采用的词表多达 2 593 项, 命名空间 33 761 项, 由此可以看出数据集中描述实体的 Schema 具有较大的差异性。另外, 关联大数据的汇聚数据源也呈现出多形态的特征, 如 WDCM 中除了主要的本体数据外, 很大一部分数据来源于关系型数据库表格、文献信息等非关系型数据和以文本形式存储的基因序列文件。

关联大数据所具有的价值性特征则主要表现为基于数据集乃至跨数据集的关联发现以及深层次关系挖掘的价值。比较经典的应用案例有, Dong 等开发的 Chem2Bio2RDF Dashboard 系统集成了化学、生物、药物领域的关联数据, 用以发现两个实体或概念之间的路径[24]。Vidal 等开发的 BioNav 系统能够基于本体技术有效发现药物和疾病之间的潜在关系等[25]。此外, 关联大数据所具备的关系网络形态, 导致其在实体聚

类、社区发现等方面也具有较高的挖掘价值。

2　关联大数据管理的典型任务和技术挑战

2.1　典型任务

传统的观点[26-27]认为，关联数据的应用包括三个方面：关联数据的发布(Publishing Linked Data)、关联数据的互联(Interlinking Linked Data)、关联数据的消费(Consuming Linked Data)。在关联大数据的场景下，这个观念呈现出两个方向的改变。

(1) 关联数据的互联和消费任务的内涵变得更加丰富，互联与消费之间的界限变得模糊，逐渐演化成关联数据的采集、加工、存储、挖掘、可视化整个过程。如：关联数据集成处理框架 LDIF[28]就为用户提供了丰富的数据采集、Schema 映射、标识识别(Identity Resolution)、质量评估与数据融合等功能模块。

(2) 关联大数据的消费与发布构成循环，消费系统通过采集开放数据，进行加工整合又形成了新一代的语义网络，再以开放数据的方式发布，整个过程构成“关联数据发布–关联数据消费–关联数据发布”的循环，关联大数据的两个形态也在反复切换，即“外在的开放数据形态–内在的语义网络形态–外在的开放数据形态”，如图 2 所示。

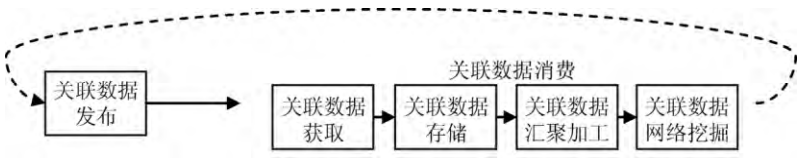

图 2　关联大数据管理的典型流程

结合关联大数据的内涵分析，关联大数据的管理任务可以分解成数据在线发布、语义查询与推理、语义封装、可扩展数据存储、多源数据汇聚加工、大规模网络智能挖掘等内容。完整的任务框架如图 3 所示。

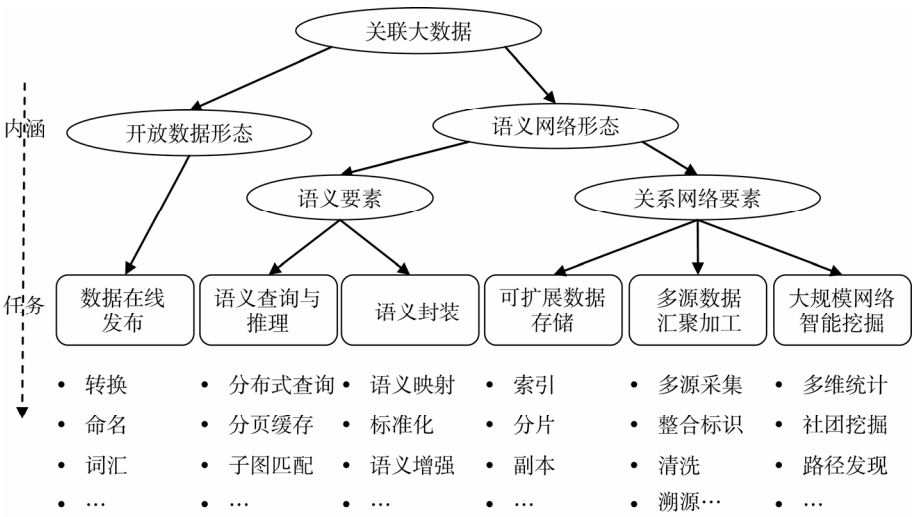

图 3　关联大数据管理的任务框架

2.2　技术挑战

传统的关联数据技术很难满足关联大数据的管理任务，在数据存储、分析计算等多个方面都存在诸多瓶颈，技术上主要面临的挑战包括：来自大体量的存储管理挑战、来自迭代式多源数据汇聚加工的挑战以及来自关联大数据多元服务需求的挑战。

(1) 来自大体量的挑战

大体量带来的首要问题就是大规模 RDF 数据的存储问题。传统的 RDF 存储管理系统，采用基于内存、文件系统和关系数据库的存储方法[29]，如：Jena TDB、

3Store、RStar、Virtuoso 等，很难满足大体量的要求。近年来，也有一些基于分布式的存储系统提出，如：RDFPeers、4store、Bigdata、YARS2、HadoopRDF 等。但这些系统针对 10 亿级甚至 100 亿级 RDF 三元组的存储管理，还存在较大的差距。有观点甚至认为由于 RDF 数据库的存储能力问题，目前只有少量的人还在用 Triple Store(三元组数据库)和 SPARQL，RDF 数据库不再成为主流[30]。

(2) 多源汇聚成为常态，数据加工成为常态

在构建关联数据语义网络时，常常需要采用开放的第三方数据源作为输入。以国家自然科学基金大数据知识管理服务平台项目(简称 NSFC-KBMS)为例，为了构建完整的成果数据库，需要集成开放的 DBLP 数据源、授权访问的 CSCD 数据以及授权访问的 Scopus 数据。这些数据源具有显著的多元异构特征，新一代的关联大数据管理系统需要支持这些多元化数据的存储需求，并提供灵活的数据汇聚模型和工具。

关联数据的加工本质上是关联数据语义网络不断演化的过程[31]，这个过程是迭代式的、极其复杂的。在数据不断快速增加的情况下，这种迭代式加工逐渐成为一种常态。然而，传统的关联数据的发布加工系统往往基于“数据量小、加工频次低”的假设，通常采用一种粗暴式的“全量批处理”方式处理 RDF 数据集。以 SILK[32]为例，SILK 允许用户制定 SILK-LSL(SILK Link Specification Language)规则文件，并藉此自动生成出不同数据集之间的实例级链接。这种模式是以指定的数据集为输入，并批量生成一批新的关联数据。其他工具，如：LinQuer(Linkage Query Writer)[33]、LIMES(Link Discovery Framework for Metric Spaces)[34]和 RDF-AI[35]都采取这种“全量批处理”模式。这种模式具有以下不足：

①对于大规模数据来说，单批次计算时间过长；

②一旦有新的数据加入，不支持增量计算，需要重新计算；

③经常在预处理、后处理阶段进行数据迁移、数据复制等操作，实际上在大规模数据的场景下，这些操作的成本很高，包括空间和时间的消耗；

④处理方法之间的组合性很差，不同处理过程之间需要大量的数据复制。

(3) 关联大数据的多元服务需求

传统的关联数据的服务形态体现为基于 Web 的 SPARQL 查询服务，即应用程序通过 HTTP 协议提交 SPARQL 查询请求，从而获取到 SPARQL 查询结果。然而，关联大数据在面向上层应用时，除了基础的 SPARQL 查询需求以外，还需要满足更多的服务需求。

①全文检索需求。作为大数据的一个重要特征，非结构数据、自由文本在关联数据的数据源中还占有较大的比重，如：成果的标题和摘要、人员的简历等，针对这些非结构化自由文本需要提供自由高效的全文检索；

②图分析需求。应用需要针对语义网络进行深层次的分析，如：最短路径计算、集中度测量(如 PageRank、特征向量集中度、亲密度、关系度、HITS)等；

③可视化浏览与交互式分析需求。即针对关联数据语义网络实现可视化挖掘分析，如：在线挖掘两个实体之间的关联路径，社区聚类展示等。

3 技术对策分析

针对以上技术挑战，目前已经有一些值得借鉴和采用的方法及大数据技术，包括 NoSQL 数据存储技术、分布式图计算技术以及大数据流水线技术。

3.1 NoSQL 数据存储

大规模的结构化、半结构化、非结构数据的产生引发了 NoSQL 运动[36]。数据库世界由最初的 SQL 垄断的局面转变成传统 SQL、NoSQL、NewSQL 分治的局面。与传统的 SQL 数据库不同，NoSQL 数据库自发明之日起，就具备着利于大数据存储的几个特性[37-38]：BASE 原则、分布式架构、横向扩展。

人们根据数据存储模型的差异性，常常将 NoSQL 数据库分成 Key-Value 数据库、列式数据库、文档数据库、图数据库。作为 NoSQL 数据库中的一个重要分支，图数据库成为关联大数据实现存储管理的最佳选择。目前典型的图数据库有 Neo4j[39]、Titan[40]、Virtuoso 等。

分布式图数据库可以解决大规模关联数据的存储问题[41-42]。然而，为了应对大数据问题，往往需要多种数据模型的混搭，而非一种单一的图数据库模型。如：H2RDF[43]作为一个大规模 RDF 查询系统，在 RDF 数据存储方面就采用了列式数据库 HBase 作为存储介质。由于 RDF 是采用三元组表示数据，而 HBase 是面向列存储稀疏的数据库，因此需要将 RDF 采用面向列的方式进行转换，然后存储到 HBase 上。H2RDF 的混搭架构如图 4 所示。

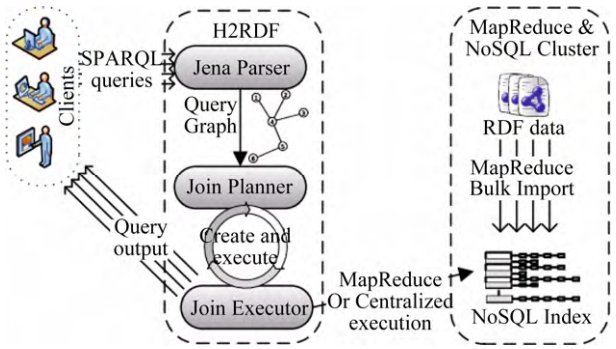

图 4　基于 HBase 存储的 RDF 管理引擎[43]

3.2　分布式图计算

随着大数据应用的深入，越来越多的图计算框架被提出，包括 GraphLab[44]、Giraph[45]、Spark GraphX[46]、Faunnus 等。这些计算框架具有较大的异构性，如：GraphLab 框架采用 Message Passing Interface (MPI)模型运行基于 HDFS 数据的复杂算法；Apache Giraph 和 Apache Hama 则采用 Bulk Synchronous Paralle(BSP)范式实现；Faunnus 项目通过用 Hadoop 运行 MapReduce 作业的方式处理 Titan 数据库中图对象；Spark GraphX 项目基于 Spark 分布式计算框架，如图 5 所示。为了实现大规模图的高效计算与处理，Spark GraphX 采用图的分布式或者并行处理方法，即将图拆分成很多的子图，然后分别对这些子图进行计算，计算时可以分别迭代进行分阶段的计算，即对图进行并行计算。

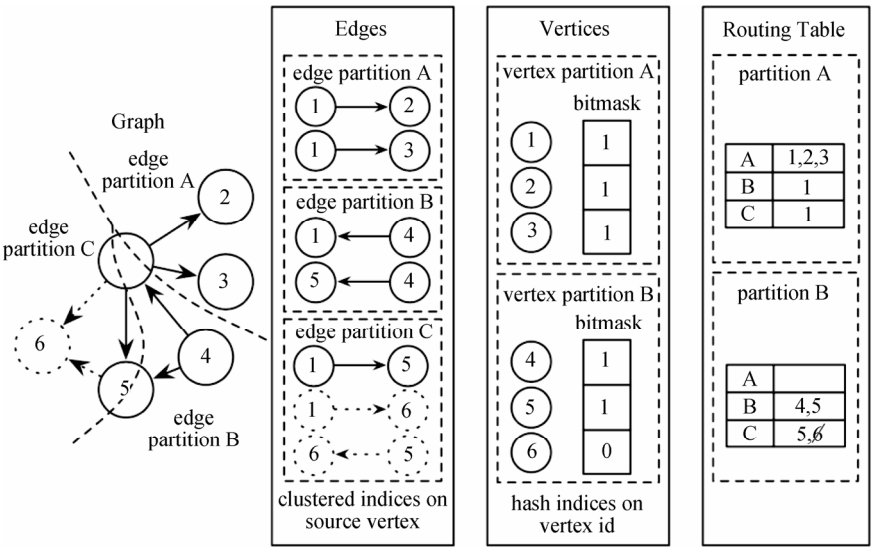

图 5　分布式图的迭代计算[46]

3.3　大数据流水线

流水线结构(Pipeline Architecture)是计算机体系结构中的术语，指在系统处理数据时，每个时钟脉冲都接受下一条处理数据的指令。其中不同的部件完成不同的任务，就像生产线流水操作一样，而不是等一个或一批产品做完，再接受下一批生产命令。每个工序完成以后，立即接受下一批生产任务，这样提高了系统处理数据的速度。大数据流水线(Pipeline)指从数据源到数据关联、数据处理及分析的一系列大数据操作[47]，如：Netflix[48]为了处理由几百个微服务系统每天产生的万亿条消息和 PB 级数据，构建大数据流水线，实现从生产者到消费平台(如 Hadoop/Elastic Search/Kafka)的大规模数据传输。

scikit-learn[49]和 GraphLab 等框架也采用流水线的概念构建系统。Spark Mllib[50]同样提供机器学习流水线机制，将机器学习的过程抽象成“数据注入–数据清洗–特征抽取–模型训练–模型验证–模型选择–模型部署”的过程。

Apache NiFi[51]是一个成熟的开源大数据流水线项目，基于其工作流式的编程理念，提供了强大的、可靠的、高度可配置的流水线定义和执行功能。然而，由于 Apache NiFi 采用专有的分布式计算框架和应用容器机制，导致很难实现与 Hadoop、Spark 等大数据框架的无缝集成。另外，NiFi 基于 Flow File 的溯源机制，

在处理大数据时往往具有极差的性能。

4 大规模图数据仓库加工系统 gETL

针对关联大数据管理的典型任务及流程，基于对其面临的有关挑战及技术对策的分析，笔者构建了面向大规模图数据仓库的 ETL(Extract-Transform-Load)加工系统 gETL。考虑到图数据模型比 RDF 模型具有更大的适用性，gETL 采用图模型作为基础模型，实现了大规模图数据的流程化存储、加工、挖掘管理，同时针对关联数据的语义特性和开放特性提供相应的发布和转换的功能。图 6 为 gETL 的总体架构，主要包括：由 Jena、TITAN、HBase、HIVE[52]等构成的存储环境；流水线系统；由 Solr、Kylin、GraphX 等构成的查询分析环境；SPARQL 查询、全文检索、多维统计以及网络分析挖掘等的接口服务。

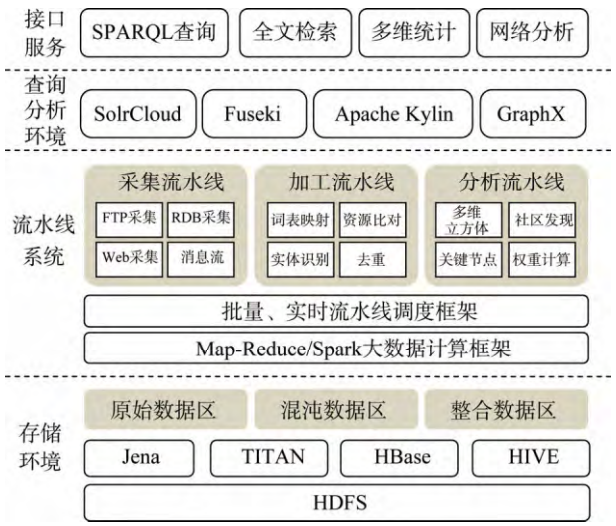

图 6 大规模图数据仓库加工系统 gETL

4.1 基于 TITAN 图引擎的 RDF 存储

为了管理平台中超大规模的属性数据及关系数据，gETL 采用大规模分布式图数据库 TITAN。TITAN 支持横向扩展，可容纳数千亿个顶点和边，并支持事务，可以支撑上千并发用户和计算复杂图形遍历。TITAN 更大的优势在于，其数据存储支持 Cassandra、HBase、BerkeleyDB，索引存储支持 ElasticSearch、Solr、Lucene。gETL 所采用的 RDF 存储系统架构如图 7 所示。

其中，TITAN 作为分布式存储、查询引擎，底层数据存储采用 HBase。面向关联数据的开放数据和语义

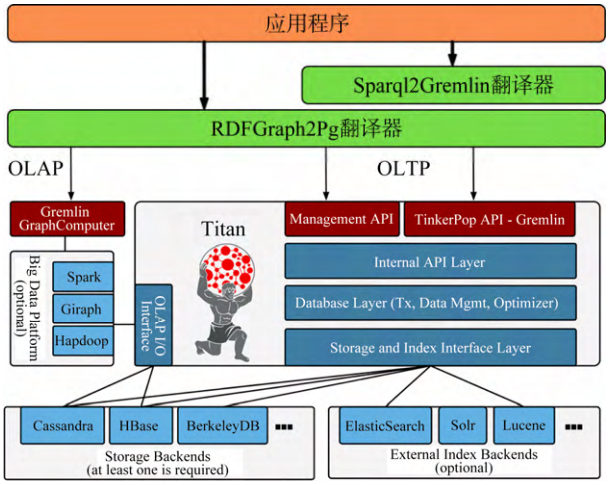

图 7 基于 TITAN 图引擎的 RDF 存储

特性，gETL 提供 RDFGraph2Pg 翻译器和 Sparql2Gremlin 翻译器。其中，RDFGraph2Pg 翻译器基于 Jena RDF I/O API 完成，主要实现 RDF 图到属性图的映射，即接受 RDF Graph(RDF 资源、属性、RDF link)作为输入，将其转换成属性图(节点、边、标签)，存储到 TITAN 中。在读取时，从 TITAN 中加载到属性图信息，结合 RDF Schema 将其转换成 RDF Graph。Sparql2Gremlin 翻译器基于 Jena ARQ 完成，主要实现 SPARQL 查询语句到 TITAN 支持的 Gremlin 语句。Gremlin[53]是 Apache TinkerPop 框架下的图遍历语言，采用一种函数式数据流语言规范，用户可以使用简洁的方式表述复杂的属性图的遍历或查询。Sparql2Gremlin 翻译器针对 Gremlin 的查询结果进行解析，将其封装成 SPARQL 查询结果返回。

4.2 流水线系统

作为 gETL 的核心部分，流水线系统基于统一的流水线软件表达模型，实现图数据的采集、存储、查询和分析等过程。一条流水线(pipeline)用以描述一个相对独立的过程，如：FTP 数据采集流水线、实体识别流水线等。如图 8 所示，流水线由多个节点组成，每个节点被称为处理器(processor)，处理器之间存在数据传输。每个处理器可具有多个输入端口(in ports)和多个输出端口(out ports)。其中，具有一个输入和一个输出的处理器被称为转换器(transform)，具有多个输入的处理器形成合流(merge)的效果，具有多个输出的处理器形成分流(fork)的操作。

与大数据采集框架 Apache Flume[54]类似，gETL 引

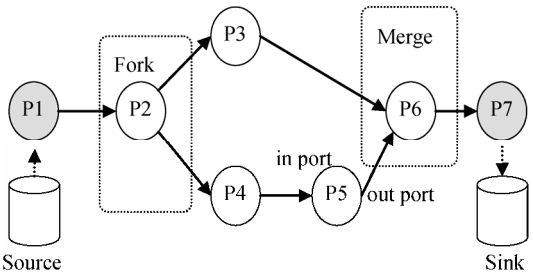

图 8　大数据流水线的抽象模型

入 Source 和 Sink 的概念。有一类处理器没有输入只有输出，它从指定的源(Source，如：关系型数据库、TCP 输入流等)读取数据；另一类处理器没有输出只有输入，它将数据写到指定的槽(Sink，如：关系型数据库、消息系统、控制台输出等)中。

在流水线系统中，流水线被抽象成多个处理器节点构成的有向无环图。流水线申请系统接收到流水线任务后，校验无误后将它编译成一个有向无环图，并将其封装成任务(job)提交给流水线任务调度系统。流水线调度系统根据调度要求(定时、实时、重复执行等)，在合适的时机执行该任务。该流程如图 9 所示。

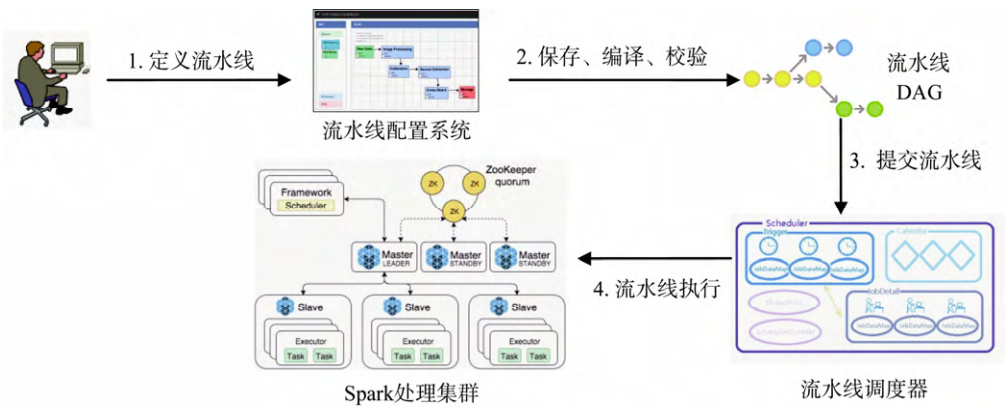

图 9　大数据流水线的执行过程

4.3　典型 RDF 流水线处理器

在数据处理的过程中，有一些针对 RDF 资源的处理任务，包括：Schema 映射、资源比对、实体识别、去重、属性 IRI(Internationalized Resource Identifiers)化等。

(1) Schema 映射。Schema 映射用以实现两个异构 RDF 资源之间的转换。如：可以将科学数据库项目中关于一个联系人的信息 csdb:Contact 映射成一个人员信息 foaf:Person 和一个机构信息 vcard:Organisation。

(2) 资源比对。资源比对用以计算两个 RDF 资源之间的相似度。如：在文献数据库中，会出现两个同名作者，为了确定他们的身份，需要考虑其他的属性(如：所在的工作单位)计算两条 RDF 资源的图形相似度。

(3) 实体识别。实体识别是根据一个 RDF 资源的属性，通过与规范记录(Authority Record)比对，获取到该资源的规范名称(URI)。与资源比对不同，实体识别往往需要借助于规范库，如根据人名规范库确定“鲁迅”和“周树人”是同一个作者。

(4) 去重。根据 RDF 资源比对结果，将两个被认为同指的资源进行合并。

(5) 属性 IRI 化。属性 IRI 化实现将 RDF 资源的属性替换成 IRI，如：将值为一段文本的 dc:subject，更换成一个指向某个分类条目的 IRI。模式映射法(根据学科分类代码创建 IRI)是一种简单实用的属性 IRI 化的方法。较复杂的属性 IRI 化方法往往还涉及资源比对等多种操作的参与。

4.4　关联数据扩展服务流水线

除 SPARQL 查询服务外，gETL 还针对其他服务要求实现了全文索引流水线，以及一系列多维统计流水线、网络分析挖掘流水线。

(1) 全文索引流水线。基于分布式搜索系统 SolrCloud，建立针对文本字段(如：人物姓名、成果标题)的全文索引，从而满足高效的全文检索要求。SolrCloud 是基于 Solr 和 Zookeeper 的分布式搜索方案，由多台服务器共同响应索引或搜索请求。

(2) 多维统计流水线。实现面向事实数据的多维度(如: 年度、地区、单位等)统计。为了满足高性能查询统计的需求, 采用 Apache Kylin 作为统计引擎。Apache Kylin[55]是由 eBay 开源的一个大数据 OLAP 框架, 在 Hadoop 上提供标准的、友好的 SQL 接口, 以及交互式的多维分析能力。

(3) 网络分析挖掘流水线。基于 Spark GraphX, 通过集成第三方图分析挖掘算法库, 实现常见的网络分析功能, 包括: 中心度计算、关键节点发现、社区发现、节点聚类等。

4.5 应用情况

关联大数据在 NSFC-KBMS 项目中得到了有效的应用, gETL 帮助建立起庞大的项目、人员、单位和成果等海量信息的大数据仓库, 存储规模可支持到十亿级关系。同时, NSFC-KBMS 基于 gETL 建立丰富的大数据采集加工流水线, 管理任务涉及到业务数据采集、成果库采集、项目数据清洗、人员数据清洗、单位数据清洗、成果数据清洗、关系清洗、多维统计、全文索引等过程, 流水线达 12 类 93 条, 涉及处理器 972 个。图 10 简单展示出流水线的种类。

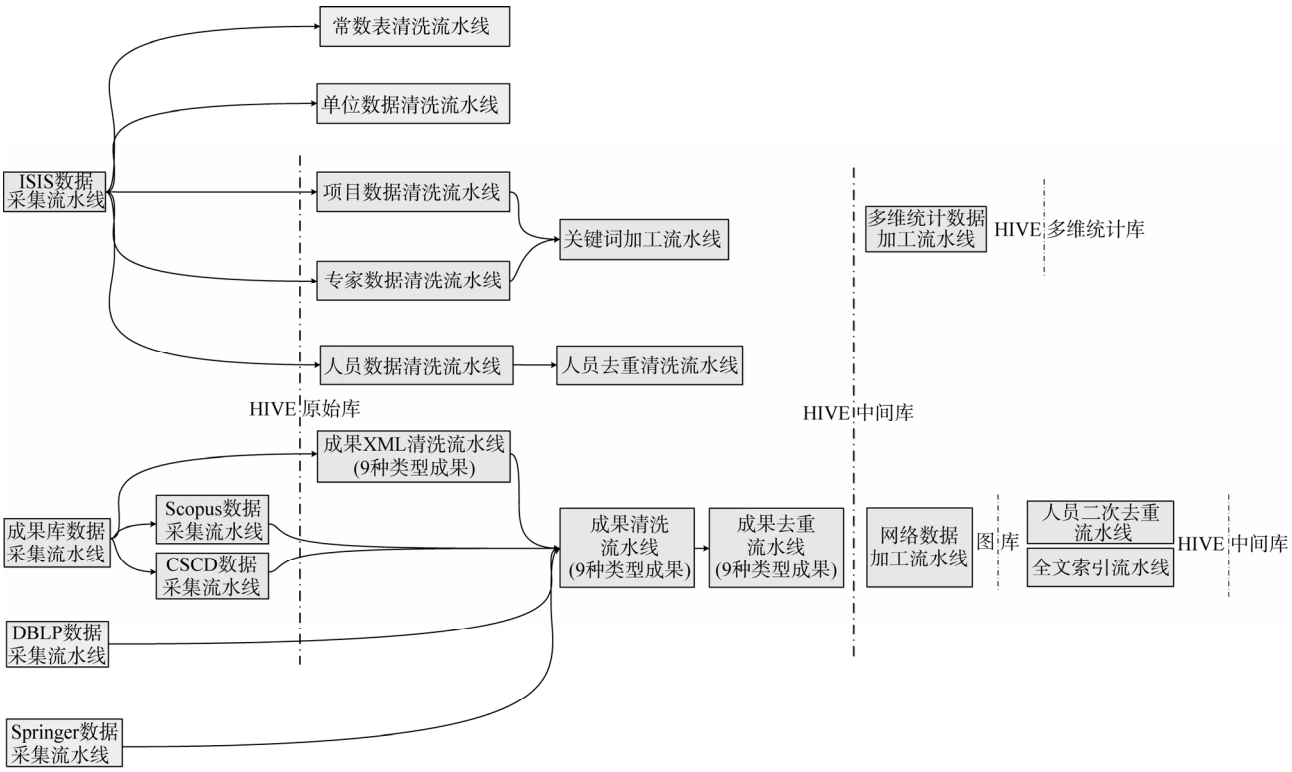

图 10 NSFC-KBMS 流水线概览

NSFC-KBMS 同时基于 gETL 建立针对申请项目事实表、获批项目事实表、人员成果事实表、项目成果事实表、获批项目参与人员事实表、获批项目参与单位事实表、人员维度表、单位维度表的多维统计流水线。从基金项目类型、项目金额、项目起止时间、人员学历、年龄、研究领域等 124 个维度建立一个大数据仓库, 并实现针对项目资助情况、结题项目成果产出、获资助科研人员、获资助科研单位的多维统计, 其中维度组合可达 31 393 种。得益于 Kylin 的高性能, 统计服务达到亚秒级响应水平。

NSFC-KBMS 还基于 gETL 建立了多条网络挖掘流水线, 一方面通过寻找专家之间关系、专家和待评审项目之间潜在的关联路径, 以期为基金项目的评审给出宝贵的参考意见; 另一方面通过发现科研人员或科研机构的合作网络, 寻找合作网络间的关键节点, 从而对关键专家、关键项目进行预测。

NSFC-KBMS 最终面向公众提供一个基于大数据网络的服务门户, 提供直观的科研社区展示、实体属性与关联展示的功能, 运行效果如图 11 所示。

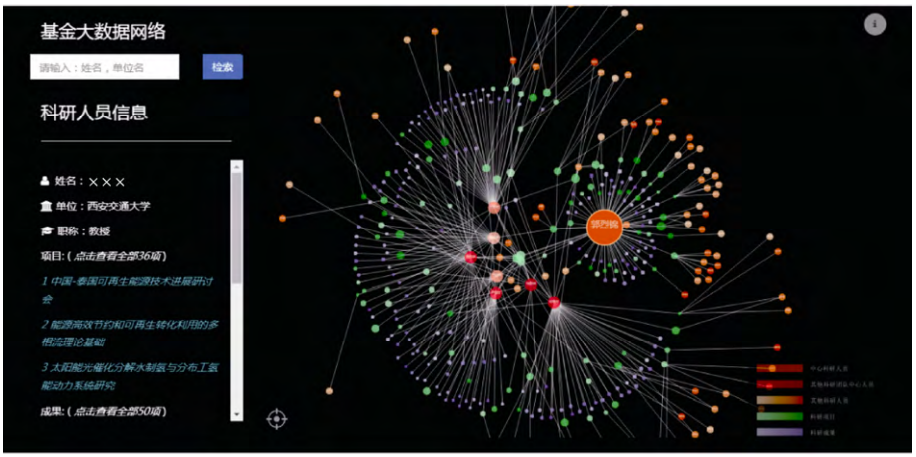

图 11　NSFC-KBMS 大数据网络服务

gETL 还成功地应用到 WDCM 项目中，目前已帮助 WDCM 实现了 30 亿 RDF 三元组的管理，提供基于 SPARQL 查询的 Web 服务应用。测试表明，对于大部分查询，gETL 的 RDF 查询能达到秒级响应[56]。下一步将继续利用 gETL 的流水线系统，实现对 WDCM 原有离散式的数据采集、汇聚、加工流程的统一管理。

5　结　语

关联大数据即体量很大、无法采用传统的管理系统(如: Jena、Virtuoso、D2R、Silk 等)进行管理的关联数据。关联大数据具有开放数据和语义网络两个显著的形态，其中语义网络同时包含语义要素和关系网络要素。关联大数据管理的任务涉及到数据存储、数据发布、数据汇聚加工以及数据挖掘统计等多个方面。面向这些任务，关联大数据的管理技术需要应对多个方面的挑战: 来自大体量的存储管理挑战、来自迭代式数据汇聚加工的处理挑战以及来自关联大数据的多元服务需求的服务挑战。

针对以上挑战，业界有一些值得借鉴和采用的方法和大数据技术，包括 NoSQL 数据存储技术、分布式图计算技术以及大数据流水线技术。基于该思路，本文介绍了大规模图数据加工系统 gETL。gETL 由存储环境、流水线系统、查询分析环境以及接口服务组成，gETL 目前成功应用在 NSFC-KBMS 和 WDCM 项目中，已实现 10 亿以上规模的生物数据和基金数据的管理与多元数据服务。

gETL 还处于研发阶段，下一步的研发计划包括: 深化应用，结合项目的需求完善流水线的框架，并扩充加工流水线算法库；进一步扩充网络分析和挖掘的流水线；完善流水线系统的容错性和错误恢复机制；进一步提高开放性，增加对计算框架 Flink[57]、Storm[58]、Apache Beam[59]等的支持。

参考文献:

[1] Berners-Lee T. Design Issues: Linked Data[EB/OL]. [2017-12-29]. https://www.w3.org/DesignIssues/LinkedData.html.

[2] 沈志宏，张晓林. 关联数据及其应用现状综述[J]. 现代图书情报技术, 2011(11): 1-9. (Shen Zhihong, Zhang Xiaolin. Linked Data and Its Applications: An Overview[J]. New Technology of Library and Information Service, 2011(11): 1-9.)

[3] BigData[J]. Nature, 2008, 455(7209): 1-136.

[4] Big Data [EB/OL]. [2017-12-29]. https://en.wikipedia.org/wiki/Big_data.

[5] 黎建辉，沈志宏，孟小峰. 科学大数据管理: 概念、技术与系统[J]. 计算机研究与发展，2017, 54(2): 235-247. (Li Jianhui, Shen Zhihong, Meng Xiaofeng. Scientific Big Data Management: Concepts, Technologies and System[J]. Journal of Computer Research and Development, 2017, 54(2): 235-247.)

[6] Hu B, Carvalho N, Laera L, et al. Towards Big Linked Data: A Large-scale, Distributed Semantic Data Storage[C]// Proceedings of the 14th International Conference on Information Integration and Web-based Applications & Services, Bali, Indonesia. New York, USA: ACM, 2012: 167-176.

[7] Hitzler P, Janowicz K. Linked Data, Big Data, and the 4th Paradigm[J]. Semantic Web, 2013, 4(3): 233-235.

[8] Big Data & Linked Data[EB/OL]. [2017-06-08]. http://www.

semantic-web.at/big-data-linked-data.

[9] Robak S, Franczyk B, Robak M. Applying Big Data and Linked Data Concepts in Supply Chains Management[C]// Proceedings of the 2013 Federated Conference on Computer Science and Information Systems (FedCSIS). IEEE, 2013: 1215-1221.

[10] 刘炜，夏翠娟，张春景. 大数据与关联数据：正在到来的数据技术革命[J]. 现代图书情报技术，2013(4): 2-9. (Liu Wei, Xia Cuijuan, Zhang Chunjing. Big Data and Linked Data: The Emerging Data Technology for the Future of Librarianship[J]. New Technology of Library and Information Service, 2013(4): 2-9.

[11] Erling O, Mikhailov I. Virtuoso: RDF Support in a Native RDBMS[A]//Semantic Web Information Management[M]. Springer, Berlin, Heidelberg, 2010: 501-519.

[12] Bizer C, Cyganiak R. D2R Server-Publishing Relational Databases on the Semantic Web[C]//Proceedings of the 5th International Semantic Web Conference. 2006.

[13] Volz J, Bizer C, Gaedke M, et al. Silk – A Link Discovery Framework for the Web of Data[C]//Proceedings of the 2nd Workshop about Linked Data on the Web. 2009.

[14] 李涓子，侯磊. 知识图谱研究综述[J].山西大学学报：自然科学版, 2017, 40(3): 454-459. (Li Juanzi, Hou Lei. Overview of Knowledge Graph[J]. Journal of Shanxi University: Natural Science Edition, 2017, 40(3): 454-459.)

[15] Auer S, Bizer C, Kobilarov G, et al. DBpedia: A Nucleus for a Web of Open Data[A]// The Semantic Web[M]. Springer, Berlin, Heidelberg, 2007.

[16] Suchanek F M, Kasneci G, Weikum G. YAGO: A Large Ontology from Wikipedia and Wordnet[J]. Web Semantics: Science, Services and Agents on the World Wide Web, 2008, 6(3): 203-217.

[17] Vrandečić D, Krötzsch M. Wikidata: A Free Collaborative Knowledgebase[J]. Communications of the ACM, 2014, 57(10): 78-85.

[18] 知识图谱的应用[EB/OL]. [2017-10-02]. http://www.36dsj.com/archives/37763. (Application of Knowledge Graph [EB/OL]. [2017-10-02]. http://www.36dsj.com/archives/37763.)

[19] Barwick H. The ‘Four Vs’ of Big Data. Implementing Information Infrastructure Symposium [EB/OL]. [2012-10-02]. http://www.computerworld.com.au/article/396198/iiis_four_vs_big_data/.

[20] IBM. What is Big Data? [EB/OL]. [2012-10-02]. http://www-01.ibm.com/software/data/bigdata/.

[21] Cyganiak R, Jentzsch A, Abele A, McCrae J. Linking Open Data Cloud Diagram [EB/OL]. [2016-12-02]. http://lod-cloud.net.

[22] Wu L, Sun Q, Desmeth P, et al. World Data Centre for Microorganisms: An Information Infrastructure to Explore and Utilize Preserved Microbial Strains Worldwide[J]. Nucleic Acids Research, 2017, 45(D1): D611-D618.

[23] Auer S, Demter J, Martin M, et al. Lodstats - An Extensible Framework for High-performance Dataset Analytics[A]// Knowledge Engineering and Knowledge Management[M]. Springer Berlin Heidelberg, 2012: 353-362.

[24] Dong X, Ding Y, Wang H, et al. Chem2Bio2RDF Dashboard: Ranking Semantic Associations in Systems Chemical Biology Space[C]// Proceedings of the 19th World Wide Web Conference on the Future of the Web in Collaboratice Science(FWCS), Raleigh, NC, USA. 2010.

[25] Vidal M E, Raschid L, Márquez N, et al. BioNav: An Ontology-Based Framework to Discover Semantic Links in the Cloud of Linked Data[A]// The Semantic Web: Research and Applications[M]. Springer, Berlin, Heidelberg, 2010.

[26] Hausenblas M. Linked Data Applications[R/OL]. Digital Enterprise Research Institute(DERI), 2009. http://pdfs.semanticscholar.org/07ec/a4bc1b06dd4bde598f834d95996cf24538cd.pdf.

[27] 夏翠娟，刘炜.关联数据的消费技术及实现[J].大学图书馆学报，2013, 31(3): 29-37. (Xia Cuijuan, Liu Wei. Technologies and Implementation of Consuming Linked Data[J]. Journal of Academic Libraries, 2013, 31(3): 29-37.)

[28] Slater T, Bouton C, Huang E S. Beyond Data Integration[J]. Drug Discovery Today, 2008, 13(13-14): 584-589.

[29] 何少鹏，黎建辉，沈志宏，等. 大规模的 RDF 数据存储技术综述[J]. 网络新媒体技术，2013, 2(1): 8-16.(He Shaopeng, Li Jianhui, Shen Zhihong, et al. Overview of the Storage Technology for Large-scale RDF Data[J]. Microcomputer Applications, 2013, 2(1): 8-16.)

[30] 从语义网到知识图谱——语义技术工程化的回顾与反思[EB/OL]. [2016-12-02]. https://zhuanlan.zhihu.com/p/22811120. (From Semantic Web to Knowledge Graph——Review of the Engineering of Semantic Technology[EB/OL]. [2016-12-02]. https://zhuanlan.zhihu.com/p/22811120.)

[31] 沈志宏，黎建辉，张晓林. 面向 LOD 的关联发现过程的定位、目标与复杂性分析[J]. 中国图书馆学报，2013, 39(6): 101-108. (Shen Zhihong, Li Jianhui, Zhang Xiaolin. Insights into Link Discovery Process for Linked Open Data: Positioning, Goals and Complexity[J]. Journal of Library Science in China, 2013, 39(6): 101-108.)

[32] Hassanzadeh O, Lim L, Kementsietsidis, et al. A Declarative Framework for Semantic Link Discovery over Relational Data[C] // Proceedings of the 18th World Wide Web Conference (WWW2009). 2009: 1101-1102.

[33] Ngomo A C N, Auer S. LIMES: A Time-efficient Approach for Large-scale Link Discovery on the Web of Data[C]// Proceedings of the 22nd International Joint Conference on Artificial Intelligence. 2011: 2312-2317.

[34] Hassanzadeh O. Publishing Relational Databases as Linked Data [EB/OL]. [2016-12-02]. http://www.cs.utoronto.ca/~oktie/slides/publishing-relational-databases-as-linked-data.pdf.

[35] Scharffe F, Liu Y, Zhou C. RDF-AI: An Architecture for RDF Datasets Matching, Fusion and Interlink[C]//Proceedings of the IJCAI 2009 Workshop on Identity, Reference, and Knowledge Representation (IR-KR). 2009.

[36] Cattell R. Scalable SQL and NoSQL Data Stores[J]. ACM SIGMOD Record, 2010, 39(4): 12-27.

[37] Wang G, Tang J. The NoSQL Principles and Basic Application of Cassandra Model[C]// Proceedings of the 2012 International Conference on Computer Science & Service System (CSSS). 2012: 1332-1335

[38] Brewer E. CAP Twelve Years Later: How the "Rules" Have Changed[J]. Computer, 2012, 45(2): 23-29.

[39] Webber J. A Programmatic Introduction to Neo4j[C]// Proceedings of the 3rd Annual Conference on Systems, Programming, and Applications: Software for Humanity. ACM, 2012: 217-218.

[40] Jouili S, Vansteenberghe V. An Empirical Comparison of Graph Databases[C]// Proceedings of the 2013 International Conference on Social Computing (SocialCom). IEEE, 2013: 708-715.

[41] Abreu D D, Flores A, Palma G, et al. Choosing Between Graph Databases and RDF Engines for Consuming and Mining Linked Data[C]// Proceedings of the 4th International Conference on Consuming Linked Data. 2013.

[42] Hernández D, Hogan A, Riveros C, et al. Querying Wikidata: Comparing SPARQL, Relational and Graph Databases[C]// Proceedings of the 15th International Semantic Web Conference. Springer International Publishing, 2016.

[43] Papailiou N, Konstantinou I, Tsoumakos D, et al. H2RDF: Adaptive Query Processing on RDF Data in the Cloud[C]// Proceedings of the 21st International Conference on World Wide Web. ACM, 2012: 397-400.

[44] Low Y, Gonzalez J, Kyrola A, et al. Distributed GraphLab: A Framework for Machine Learning and Data Mining in the Cloud[J]. Proceedings of the VLDB Endowment, 2012, 5(8): 716-727.

[45] Avery C. Giraph: Large-scale Graph Processing Infrastructure on Hadoop[C]//Proceedings of the Hadoop Summit. 2011.

[46] Xin R S, Gonzalez J E, Franklin M J, et al. Graphx: A Resilient Distributed Graph System on Spark[C]//Proceedings of the 1st International Workshop on Graph Data Management Experiences and Systems. ACM, 2013: 2.

[47] Koitzsch K. Data Pipelines and How to Construct Them[A]// Pro Hadoop Data Analytics[M]. Apress, 2017: 77-90.

[48] Yi X, Liu F, Liu J, et al. Building a Network Highway for Big Data: Architecture and Challenges[J]. IEEE Network, 2014, 28(4): 5-13.

[49] Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: Machine learning in Python[J]. Journal of Machine Learning Research, 2011, 12: 2825-2830.

[50] Meng X R, Bradley J, Yavuz B, et al. Mllib: Machine Learning in Apache Spark[J]. Journal of Machine Learning Research, 2016, 17(1): 1235-1241.

[51] Apache NiFi. An Easy to Use, Powerful, and Reliable System to Process and Distribute Data[EB/OL]. [2016-12-02]. http://nifi.apache.org/.

[52] Thusoo A, Sarma J S, Jain N, et al. Hive-A Petabyte Scale Data Warehouse Using Hadoop[C]//Proceedings of the 26th International Conference on Data Engineering(ICDE). IEEE, 2010: 996-1005.

[53] Avram A. Gremlin, A Language for Working with Graphs [EB/OL]. [2016-12-02]. https://www.infoq.com/news/2010/01/Gremlin.

[54] Wang C, Rayan I A, Schwan K. Faster, Larger, Easier: Reining Real-time Big Data Processing in Cloud[C]// Proceedings of the Posters and Demo Track. ACM, 2012.

[55] Ranawade S V, Navale S, Dhamal A, et al. Online Analytical Processing on Hadoop Using Apache Kylin [EB/OL]. [2016-12-02]. http://www.ijais.org/archives/volume12/number2/ranawade-2017-ijais-451682.pdf.

[56] Li L, Shen Z H, Li J H, et al. A Resilient Index Graph for Querying Large Biological Scientific Data[C]//Proceedings of the 2017 IEEE International Congress on Big Data (BigData Congress). 2017: 435-443.

[57] Carbone P, Katsifodimos A, Ewen S, et al. Apache Flink: Stream and Batch Processing in a Single Engine[J]. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 2015, 36(4): 28-38.

[58] Jones M. Process Real-time Big Data with Twitter Storm

[EB/OL]. [2016-12-02]. https://www.ibm.com/developerworks/library/os-twitterstorm/index.html.

[59] Apache Beam: An Advanced Unified Programming Model [EB/OL]. [2016-12-02]. https: //beam.apache.org.

作者贡献声明:

沈志宏: 提出论文总体思路, 确定论文总体写作框架, 撰写论文;
姚畅: 参与论文总体思路讨论, 重点完成 NSFC-KBMS 项目背景、需求、应用方案与效果的撰写;
侯艳飞: 提出论文修改意见, 重点参与文章第 1 节和第 4 节内容的调整和改进;
吴林寰: 参与论文总体思路讨论, 重点完成 WDCM 项目背景、需求、应用方案与效果的撰写;
李跃鹏: 参与论文总体思路讨论, 完成参考文献的校核与修改。

利益冲突声明:

所有作者声明不存在利益冲突关系。

收稿日期: 2017-12-12
收修改稿日期: 2018-01-07

Big Linked Data Management: Challenges, Solutions and Practices

Shen Zhihong[1] Yao Chang[2] Hou Yanfei[1] Wu Linhuan[3] Li Yuepeng[1]
[1](Computer Network Information Center, Chinese Academy of Sciences, Beijing 100190, China)
[2](National Natural Science Foundation, Beijing 100085, China)
[3](Institute of Microbiology, Chinese Academy of Sciences, Beijing 100101, China)

Abstract: [**Objective**] This article analyzed the concept, connotation and characteristics of the big linked data, aiming to explore possible solutions for technical challenges facing its management. [**Methods**] We proposed a new model based on NoSQL data management, distributed graph computing and big data pipeline technologies, which designed and develop gETL, a large-scale graph data warehouse processing system. [**Results**] The proposed system was used in NSFC-KBMS and WDCM projects, which effectively manages large-scale knowledge-data and biological data. [**Limitations**] The proposed system could be improved with new applications. [**Conclusions**] The NoSQL data storage, distributed graph computing, and big data pipeline technologies, as well as the gETL system, help us address the challenges facing linked big data management.

Keywords: Linked Data Knowledge Graph Big Data Big Linked Data