计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2017.20160939
54(2):258-266,2017

高能物理大数据挑战与海量事例特征索引技术研究

程耀东[1] 张 潇[2] 王培建[2] 查 礼[3] 侯 迪[2] 齐 勇[2] 马 灿[4]
[1](中国科学院高能物理研究所 北京 100049)
[2](西安交通大学计算机科学与技术系 西安 710049)
[3](中国科学院计算技术研究所 北京 100190)
[4](中国科学院信息工程研究所 北京 100093)
(chyd@ihep.ac.cn)

Data Management Challenges and Event Index Technologies in High Energy Physics

Cheng Yaodong[1], Zhang Xiao[2], Wang Peijian[2], Zha Li[3], Hou Di[2], Qi Yong[2], and Ma Can[4]
[1](*Institute of High Energy Physics, Chinese Academy of Sciences, Beijing* 100049)
[2](*Department of Computer Science and Technology, Xi'an Jiaotong University, Xi'an* 710049)
[3](*Institute of Computing Technology, Chinese Academy of Sciences, Beijing* 100190)
[4](*Institute of Information Engineering, Chinese Academy of Sciences, Beijing* 100193)

Abstract Nowadays, more and more scientific data has been produced by new generation high energy physics facilities. The scale of the data can be achieved to PB or EB level even by one experiment, which brings big challenges to data management technologies such as data acquisition, storage, transmission, sharing, analyzing and processing. Event is the basic data unit of high energy physics, and one large high energy physics experiment can produce trillions of events. The traditional high energy physical data processing technology adopts file as a basic data management unit, and each file contains thousands of events. The benefit of file-based method is to simplify the complexity of data management system. However, one physical analysis task is only interested in very few events, which leads to some problems including transferring too much redundant data, I/O bottleneck and low efficiency of data processing. To solve these problems, this paper proposes an event-oriented high energy physical data management method, which focuses on high efficiency indexing technology of massive events. In this method, event data is still stored in ROOT file while a large amount of events are indexed by some specified properties and stored in NoSQL database. Finally, experimental test results show the feasibility of the method, and optimized HBase system can meet the requirements of event index.

Key words high energy physics; data management; event index; HBase; query optimization

摘 要 新一代高能物理实验装置的建成与运行,产生了PB乃至EB量级的数据,这对数据采集、存储、传输与共享、分析与处理等数据管理技术提出了巨大挑战.事例是高能物理实验的基本数据单元,一次大型实验即可产生万亿级的事例.传统高能物理数据处理以ROOT文件为基本存储和处理单位,每个ROOT文件可以包含数千至数亿个事例.这种基于文件的处理方式虽然降低了高能物理数据管理系统的开发难度,但物理分析仅对极少量的稀有事例感兴趣,这导致了数据传输量大、I/O瓶颈以及数据处理效率低等问题.提出一种面向事例的高能物理数据管理方法,重点研究海量事例特征高效索引技术.

收稿日期:2016-11-15;**修回日期**:2017-01-09
基金项目:国家重点研发计划项目(2016YFB1000604)
This work was supported by the National Key Research Program of China (2016YFB1000604).

在这种方法中,将物理学家感兴趣的事例的特征量抽取出来建立专门的索引,存储在 NoSQL 数据库中.为便于物理分析处理,事例的原始数据仍然存放在 ROOT 文件中.最后,通过系统验证和分析表明,基于事例特征索引进行事例筛选是可行的,优化后的 HBase 系统可以满足事例索引的需求.

关键词 高能物理;数据管理;事例索引;HBase;查询优化

中图法分类号 TP311.133

为了研究宇宙起源、天体演化、物质结构组成等科学问题,人类建造了许多大科学设施,包括欧洲大型强子对撞机(large hadron collider, LHC)、北京正负电子对撞机(Beijing electron-positron collider, BEPC)、大亚湾中微子实验、江门中微子实验(Jiangmen underground neutrino observatory, JUNO)、高海拔宇宙线实验(large high altitude air shower observatory, LHAASO)等.随着实验装置规模不断扩大和精度的提高,产生了越来越多的数据,比如 LHC 在线数据率高达每秒 1 PB,需要长期保存和处理的数据达到每年 50 PB.当前,高能物理领域总体累积的数据已经接近 1 000 PB,并且还在不断增加,全球有近万名物理学家利用这些数据进行物理研究.这样大的数据量需要超大规模的存储、计算及网络资源,大量的计算任务需要由所有高能物理合作单位共同承担.物理学家把分布于全世界的存储、计算资源整合到一起,形成一个超高性能的通用计算基础设施——WLCG(worldwide LHC computing grid)网格[1],提供大量的计算和存储资源,用于数据的处理、模拟和分析.

高能物理数据处理过程包括数据筛选、数据重建、物理模拟以及分析等.目前,高能物理的实验数据以文件为单位进行管理,每个文件包含了若干个事例.事例是基本的数据单元,指一次粒子对撞或者一次粒子间的基本相互作用产生的数据,包含了条件参数以及相关的物理量,比如光子数、带电径迹数、电子数等,一个大型高能物理实验可以产生数十亿甚至万亿级别数量的事例.另一方面,由于高能物理实验装置的规模及数据量巨大,通常一家单位难以处理全部的数据,数据由分布在全球的高能物理单位合作完成.这种分布式的、以文件为基础的存储方式,大大简化了数据管理的复杂度,在很长一段时间内促进了高能物理领域的发展.

然而,随着实验数据的飞速增长以及新技术的出现,这种传统的数据存储和处理方式也暴露出越来越多的弊端.首先,文件形式的数据虽然存储方便,但不利于数据的检索.而数据检索在高能物理的数据处理中占很大比重.因此,以文件为基础的存储,大大降低了数据处理的效率.其次,数据处理程序只能运行在存储数据的站点,所以需要提前将数据以文件的方式传输到指定的站点.这种方式难以实现计算资源的灵活调度,而文件传输到目标站点后只有其中少部分被使用,造成带宽的浪费.因此,提高数据处理效率和资源利用率是高能物理软件领域亟待解决的一个重要问题.

提高现有系统的处理效率并不是一个简单的任务,存在诸多挑战:1)文件格式的存储方式未提供有效的属性查询功能,致使事例检索效率非常低下.当物理学家检索事例时,关心的属性只有少数几个,关心的事例也通常少于原始数据的 1/100,甚至 1/1000000(如图 1 所示).但针对文件进行检索,需要访问某一范围内的所有文件,并读取每个事例的所有属性值.大量的 I/O 操作都是无用的.2)分站点存储空间不足且网络传输速度有限,这给计算任务在分站点运行提出挑战.由于分站点的规模往往远小于主站点,无法存储所有数据的完整拷贝,需要在计算时再临时复制数据到分站点.由于网络传输速度和数据量之间的矛盾,实时复制数据会造成很大的延迟以及文件系统的开销,甚至系统宕机.3)数据格式在存储和处理上具有不一致性.数据存储的方式是文件,而数据处理的单位是事例,系统需要大量的转化操作,造成极大的开销.4)已有系统极为复杂,

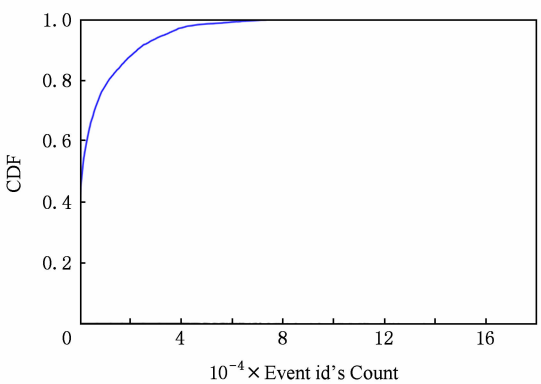


Fig. 1 CDF (cumulative distribution function) of the simulated event selection

图 1 模拟筛选返回的事例数及所占百分比

新的处理方式难以兼容. 高能物理领域针对每一个实验装置都会开发各自的离线数据处理软件系统，长期以来形成了独立的体系，系统的优化不能对这些应用软件造成太大的影响.

本文研究一个重要的问题：如何提高高能物理数据管理系统的效率. 为了应对以上挑战，本文设计了面向事例的数据管理系统，有效解决事例数据处理效率低以及分站点资源利用率低的问题. 首先设计了一个基于 NoSQL 数据库[2]的事例索引系统. 通过事例数据特征抽取，选取物理学家最感兴趣的属性作为索引，存储在数据库中，并采用倒排索引技术，提高事例数据检索的效率. 接着，针对事例数据进行缓存优化，减少数据转化和存储开销. 然后，提出事例级跨域传输方案，降低分站点处理数据的延迟. 作者在一个小型的原型系统上实现了事例索引并进行了试验验证. 实验结果表明，事例级的索引技术能够显著提高事例数据的检索效率.

本文首先介绍高能物理的数据处理流程以及面向事例管理的相关工作，然后重点介绍面向事例的科学数据管理系统设计，最后给出系统验证与分析结果，分析性能提升的原因.

1 相关背景

1.1 高能物理数据处理流程

高能物理实验主要有 3 个要素，分别是粒子源、用以观察和记录各种高能粒子的相关信息的探测器以及用于获取和处理这些信息的电子学系统. 探测器电子学系统的结构比较复杂，有时甚至需要计算机程序来控制，因此制造的技术难点也相对较高. 高能物理数据处理过程的过程如图 2 所示，主要包括数据筛选、重建、分析和模拟等部分.

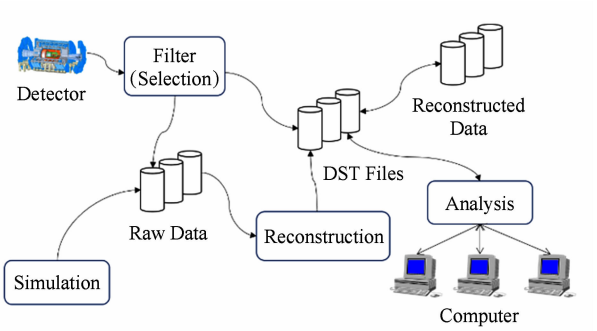


Fig. 2 Data processing workflow in high energy physics

图 2 高能物理数据处理过程示意图

粒子在加速器中经高速对撞后产生的信息会被传感器捕获，交由在线系统进行筛选，之后传送到离线系统中，以文件的形式保存在磁盘系统中，并在磁带库中进行永久保存，同时在随机读写存储系统中进行缓存，这些数据叫做原始数据(Raw Data)，由事例(Event)组成. 之后这些原始数据会借助对撞和取数时的参数进行重建，并被赋予真实的物理意义. 重建之后的数据保存为 ROOT 格式[3]的 DST 文件，由于软件升级等原因，重建的过程可能进行多次. 不同条件下产生的重建数据被标记为不同的版本以便区分，如 655，664 等. 最后，物理学家利用数据分析框架对重建后的数据做深入的分析，并生成用于绘制图表的结果 ROOT 文件. 在数据分析阶段，不同的物理学家会根据自己的需要或建立的模型设计并实现一组数据分析/筛选算法，算法通常由顶点拟合、径迹筛选、4C 拟合、5C 拟合等多个部分构成. 数据分析过程往往涉及大量的数据读取工作，但最终满足条件的事例数通常仅占整体数据量的 1‰左右，之后高能物理学家还会对这些挑选出来的数据做进一步的分析和计算. 有时物理学家为验证分析结果正确与否，还要使用蒙特卡罗模拟软件产生与原始数据规模相当的模拟数据，之后对这些数据进行重建，最后再对重建后的数据进行相同的分析，用以与之前的分析结果进行比对.

当前，高能物理实验以文件为单位进行数据管理与计算. 由于事例之间的无关性和独立性，高能物理往往把一系列的事例组成一个文件，多个文件可以在多个机器上同时处理，而不需要相互通信. 因此，高能物理计算的特点是高吞吐率的数据并发. 基于这些特点，目前高能物理领域普遍采用集群计算系统以及计算和存储分离的模式，典型的系统结构如图 3 所示：

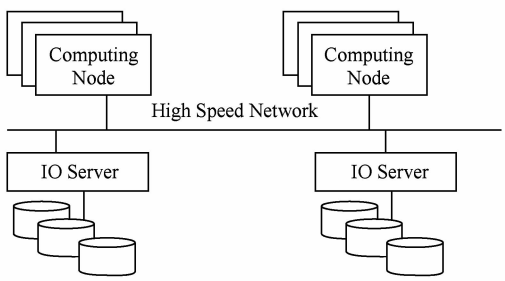


Fig. 3 Typical architecture of HEP computing system

图 3 高能物理计算系统典型结构

海量的高能物理实验数据存储在 I/O 服务器中，计算节点通过高速网络从 I/O 服务器中获取数据. 多个 I/O 服务器通过分布式存储系统来管理，比如 Lustre，GPFS，EOS 等. 计算节点通过批处理作业系统来管理，比如 HTCondor，LSF，PBS 等.

1.2 面向事例管理的相关工作

为了能够快速得到物理学者感兴趣的数据集,最初的方案是把索引信息存储到高能物理领域中常用的 ROOT 文件格式中,称为索引文件.实际上,真正的事例数据仍然存储在数据文件中.这样,物理学家在进行研究时,先读取索引文件,筛选出感兴趣的事例 ID,最终通过事例 ID 在数据文件中提取感兴趣的事例.中国科学院高能物理研究所的刘北江从事例数据中抽取一系列的特征量(称为 TAG),存储在单独的 ROOT 文件中,形成索引文件.用户筛选时首先在该索引文件中查找,减少遍历数据文件的次数,应用于北京谱仪 BESIII 的实验中[4].澳大利亚墨尔本大学的 Bloomfield 等人将特定筛选条件选取的事例位置信息存储在 ROOT 文件中,应用到日本的 BELLE2 实验中[5].这样,用户在做分析时直接免去预筛选过程,但是这种方法只能适用于固定模式的筛选,无法满足用户个性化的需求.

采用文件存储索引的方式在管理、共享和访问性能方面难以满足更大规模的实验数据,因此在有些大型高能物理实验中采用关系型数据库来存储事例索引.物理学者在进行事例筛选时,通过数据库查询语句获得符合条件的事例 ID,最后再从原始数据文件中提取出事例.比如,欧洲核子中心 CERN 的 Goosens 以及美国阿贡国家实验室的 Cranshaw 等人采用 Oracle 数据库存储 ATLAS 实验(超环面仪器)的索引信息,通过水平分区、纵向分区等数据库优化等技术手段实现了 10 亿级别的事例索引 TAGDB[6].

随着事例数量不断增加,近年来有很多研究人员提出采用 NoSQL 来存储索引信息.西班牙的 Sánchez 等人采用 HBase 构建了 ATLAS 实验(超环面仪器)事例索引数据库 EventIndex[7].中国科学院高能物理研究所的孙功星等人提出将事例数据存储在 HBase 中并建立特征事例索引,以加快数据分析过程[8].

值得指出的是,21 世纪初曾经一段时间,有相关人员提出将高能物理全部的实验数据存储到面向对象的数据库中,比如美国斯坦福直线加速器中心 SLAC 的 Becla 将 BaBar 的实验数据完全存入到 Objectivity/DB 中[9].欧洲核子中心 CERN 的 Düllmann 提出将 LHC 的海量数据存储到 Objectivity/DB 中[10].但是,这些方案最终没有成功实施,目前高能物理的数据存储仍然采用文件管理的方式.

从以上的分析可以看出,完全将事例数据存储到关系型、面向对象或者 NoSQL 中的方案,在高能物理领域还没有得到广泛的应用.将数据文件存储与特征索引结合起来是一个可行的方案,但是目前的工作都是针对某个实验或者解决具体问题开展,缺乏通用、可扩展和全面的解决方案及系统.

2 面向事例的科学数据管理系统设计

结合高能物理海量数据管理需求和研究现状,本文采用 ROOT 文件存储和 NoSQL 事例特征索引融合的管理架构,提出一个通用、可扩展的方案,设计面向事例的科学数据管理系统,实现高能物理海量数据的高效管理、快速处理以及远程访问等.

该系统主要包括 4 个部分:事例特征抽取、事例索引数据库、面向事例的缓存、面向事例的传输,其架构如图 4 所示.在传统的高能物理计算环境中,高能物理数据处理软件,比如 BOSS(BESIII Offline Software System)[11],SNiPER(Software for Non-collider Physics Experiments)[12],LodeStar(LHAASO Offline Data Processing Software Framework)等,直接访问实验数据的存储系统,比如 Lustre,GPFS,EOS 等分布式文件系统.在本文的设计中,面向事例的科学数据管理系统位于高能物理数据处理软件与实验数据存储系统之间,提供事例级的海量数据管理.同时,数据处理软件仍然可以使用原有的方式,直接访问数据存储系统,从而保证了系统的兼容性.

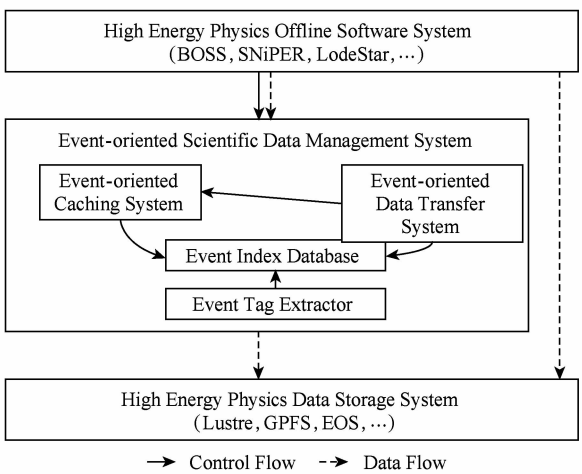


Fig. 4 Architecture of event-oriented scientific data management system

图 4 面向事例的科学数据管理系统结构

采用面向事例的科学数据管理系统以后,事例特征抽取模块会扫描实验数据存储系统,从中抽取

出物理学家过滤事例的特征量,并保存到事例索引数据库中.事例特征数据库记录了事例的特征属性以及事例的存储位置和偏移量,并且将特征属性值编码至 NoSQL 数据库的主键中,同时提供事例查询接口.高能物理数据处理软件在做分析时,首先通过筛选条件查询事例索引数据库,得到感兴趣的事例集合.接着,调用事例缓存的接口.如果该事例已经在缓存系统中,就会直接给数据处理软件返回该事例数据.如果数据处理软件运行在远程站点,当需要某个事例时,系统还会触发面向事例的传输系统将事例数据从网络上实时传输给数据处理软件.

2.1 事例特征抽取

由于原有物理数据被封装在 ROOT 等高能物理处理框架的数据对象中,数据在相关物理软件外对用户是不可见.所以为了能够快速查找相关物理事例,需要预先提取事例粒度的相关特征数据,以供后续的查找.

事例特征抽取模块负责识别不同实验、不同格式的数据文件,并从中抽取出对于数据处理有意义的特征变量,比如 BESIII(北京谱仪)实验中,包含了运行号、事例号、总径迹数、总带电径迹数、总不带电径迹数、好的光子数、好的正负带点径迹数、好的正负介子数、好的正负 k-介子数、好的正负质子数、可见光能量定义等 16 个特征属性.识别出这些特征属性后,事例抽取模块根据用户定义,将其中的特征存储到事例索引数据库中.

事例特征抽取模块基于 ROOT 框架实现,与具体的实验无关.为了保证系统的通用性,该模块定义了一个规范的接口.每个高能物理实验通过配置接口定义文件即可实现相应的事例特征抽取功能,可以指定需要将哪些特征属性存储到事例特征数据库中.高能物理中存在多种不同的文件格式,该模块会分类识别,主要包括 AOD(Analysis Object Data),重建数据摘要信息,用于物理分析);ESD(Event Summary Object),全部的重建输出数据;EVNT(蒙特卡罗模拟产生的事例);RDO(Raw Data Objects),原始数据及其产生原始数据的条件信息.一般情况下,不需要对原始数据 Raw Data 建立索引.原始数据是探测器产生的字节流,其中的事例信息可以从重建后的 AOD 或者 ESD 中获得.

2.2 事例索引数据库及查询条件归并

在提取了物理事件级别的特征后,我们还要能够有效地组织并索引千亿甚至万亿级别的事例数据,达到能够在现有文件中快速提出物理事例集合的目的.

高能物理实验中的事例数量庞大,单个大型实验可以达到百亿甚至万亿级别.事例的属性从几十到几百个不等.高能物理数据处理并发访问量非常高,大型集群和网格计算的并发任务量可达到十万级别.这要求事例索引数据库具有非常好的可扩展性和性能.基于以上的需求,本系统采用基于 HBase 集群[13]来构建是索引数据库.

首先,由于 HBase 中主键的构建采用了按字典序排序的索引结构,并且通常缓存在内存中[14],因而具备很好的查询效率.所以我们在 HBase 中将前一步中提取到的事例界别的属性名及其具体值编码到了 HBase 的 Rowkey 中,以支持使用在主键上进行二分查找.此外,利用提取后的事件特征数据进行查询条件的归并,满足相同查询条件的事例集合会被归并到 HBase 的一条记录中,可以使得满足同一条件的所有事例信息可以在一次查询中返回.HBase 的事例特征索引构建如图 5 所示:

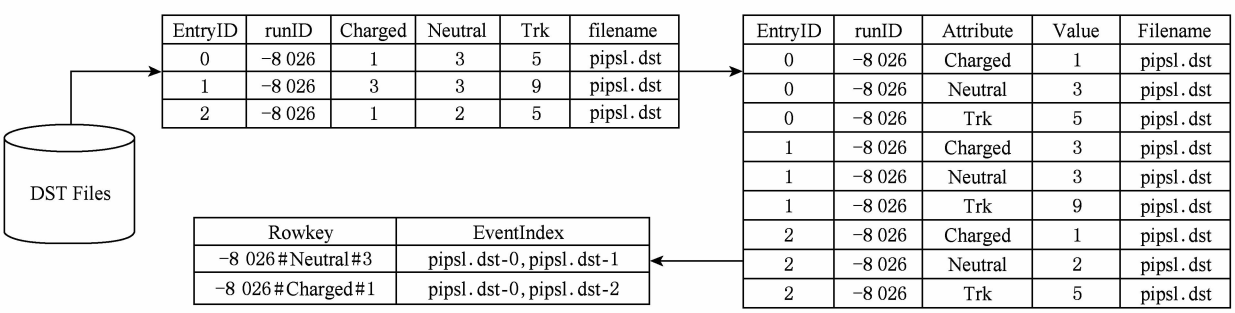


Fig. 5 Building event index in HBase

图 5 在 HBase 中构建事例特征索引示意图

2.3 面向事例的缓存

物理学者感兴趣的数据集通常会呈现出一定的访问模式.为了能够减少重复查询中消耗的 I/O 资源,系统需要将现有的查询热区缓存起来.

一次高能物理数据分析过程中,仅仅对某些稀有事例感兴趣,而这些稀有事例分布在不同的 ROOT 文件中.所以,物理分析过程中,仅仅读取文件的一小部分数据,针对文件的预读和缓存等存储系统优化方法难以发挥作用.中国科学院高能物理研究所对 BESIII 实验数据分析过程的文件访问模式分析发现,大部分的文件读连续请求的大小分布在 256 KB~4 MB 之间,每两个连续请求之间都有 offset,65%的 offset 绝对值分布在 1~4 MB 之间,也就是说文件的读访问方式为大记录块的跳读.如果打开文件系统的预读选项,会读取大量无效数据,导致性能急剧下降[15].

为此,本系统引入了面向事例的缓存.系统记录事例数据的访问频次,将高访频度的事例数据缓存到 SSD 以及内存中,从而减少索引和事例数据文件之间的 I/O 开销.面向事例的缓存模块检测到需要缓存的事例后,将该事例进行序列化存储.当数据处理软件调用接口获取事例时,面向事例的缓存模块再将存储在 SSD 及内存的事例进行反序列化,以 ROOT 的对象直接返回,而不需要再从底层存储系统中读出.

2.4 面向事例的数据传输

高能物理领域广泛采用分布式计算,将计算任务分布到全球合作站点上运行.欧洲大型强子对撞机产生海量数据便是由 WLCG(Worldwide LHC Computing Grid)负责存储和处理的.WLCG 采用了三级站点的网格形式[16],主要分为 Tier0,Tier1 和 Tier2.Tier0 主要负责获取并保存对撞机产生的原始数据,同将其发送给多个 Tier1 站点作为副本进行保存;Tier1 主要负责对原始数据进行重建以及一些后续的处理工作;Tier2 主要负责产生模拟数据和物理分析等工作.在 WLCG 的 Tier 结构中,数据并不是完全复制到所有的站点中,因此计算任务会被调度到存储数据的地方.如果某个站点需要分析感兴趣的数据,需要提前进行数据订阅,将数据预先传输到指定的站点.CMS(紧凑 μ 子线圈)实验使用 PhEDEx 系统[17]实现 WLCG 站点之间传输数据.

不同于 WLCG 预先传输文件,面向事例的数据传输系统仅传输物理分析程序所感兴趣的事例,所需数据量大幅降低,随着网络带宽不断提升,将可以支持计算任务实时传输数据.数据传输系统的结构如图 6 所示:

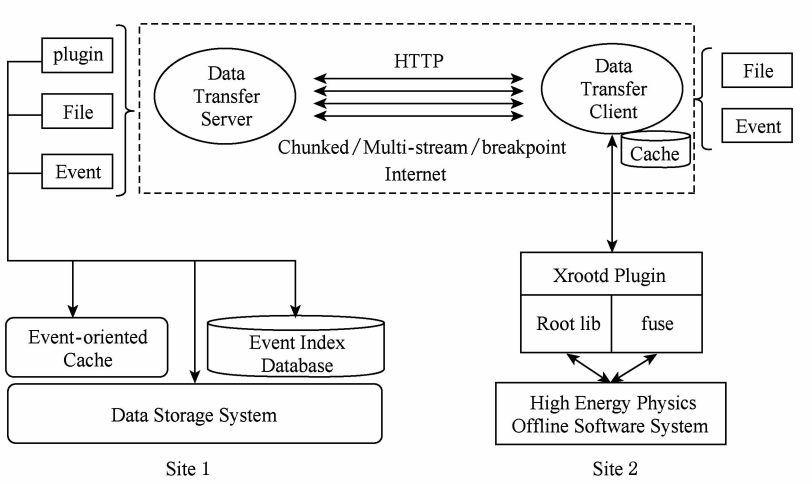


Fig. 6 Architecture of event-oriented data transfer system

图 6 面向事例的数据传输系统结构

数据传输系统由数据传输服务器和数据传输客户端 2 部分构成,分别运行在 2 个不同的站点.运行在远程站点的高能物理数据处理软件在做物理分析时不用考虑数据是否在本地站点,它可以通过 ROOT 框架或者本地文件系统接口来访问所需要的事例数据.首先,数据处理软件调用时事例索引数据库获得事例索引信息,然后通过数据传输客户端向数据传输服务器发送事例请求.数据传输服务器从数据存储系统或者事例缓存中将事例数据序列化传输到客户端,然后客户端再将事例反序列化以 ROOT 对象的方式返回给数据处理软件.如果数据处理软件以文件系统接口调用,数据传输系统仅传输所需要的数据块,以减少传输量.为了提升数据访问性能,在数据传输客户端也设置了基于事例和数据块的缓存系统.数据传输基于 HTTP 协议,支持分块、多流及断点续传等功能.

3 系统验证与分析

3.1 实验条件

本文在 Hadoop 2.6.2 平台上建立了验证系统,采用 4 台服务器构建 Hadoop 集群,其中 1 台主节点,3 台数据节点. 硬件选用曙光 A620 服务器,每台服务器配备 2 颗 AMD Operon 6320 服务器、64 GB 内存、1 块 1 TB 7200 RPM SAS 硬盘. 节点之间采用千兆以太网互联,操作系统为 Ubuntu 14.04.

实验过程中选用了北京谱仪 BESIII 的真实运行数据,共包含 384 个 DST 文件、1 400 万个事例. 基于以上数据我们构建了事例索引数据库. 事例索引中包含 7 个特征量,即:entry(事例文件内编号),runNo(运行号),eventID(事例实验全局编号),totalCharged(总的带电粒子数),totalNeutral(总的中性粒子数),totalTrks(总的径迹数),以及原始的 DST 文件名.

3.2 实验结果分析

为验证事例索引数据库的有效性,实验开展了如下工作:1)模拟用户查询;2)关系型数据库查询效率;3)对比未归并查询条件与经过查询条件归并的 HBase 查询效率;4)验证在不同试验中条件归并效果.

1) 模拟用户查询

实验中,首先指定 RunNo,然后再选择属性值,模拟用户真实的事例查询模式,并使用蒙特卡洛方法随机数产生查询条件,用于模拟用户的查询. 在测试的数据中,所有 DST 的文件中共包含了 1 400 万个事例,查询返回理论上限为 1 400 万条. 实验中产生了 1 000 条模拟查询条件,其中在 1 400 万事例数据中有效的单值查询的条件和对应事例数量的累计分布图如图 1 所示,有 77%的查询返回低于 1 万条数据(少于千万分之一). 这说明了事例筛选是有效的,可以大大降低用户遍历原始数据文件的开销.

2) 关系型数据库查询效率

关系型数据支持多个索引,能够灵活支持结构化数据查询,因此也是构建事例特征索引数据库的一个选项. 本实验将事例索引存放到 MySQL 数据库中,做 1 000 次模拟查询,查询时间大部分集中在 200～500 ms 之间. 实验结果如图 7 所示.

3) HBase 查询效率

上面的实验中对 MySQL 的各个字段都增加了索引. 对比传统 RDBMS 的实验结果,查询时间有了一定的降低,但是查询效率提升并不明显. 此外,如果直接采用未优化的 HBase,由于 Schema 固定,难以支持灵活的半结构化数据查询,而且对于新增加的数据,需要更新相关的索引,对于大规模的应用及后期扩展仍然存在问题.

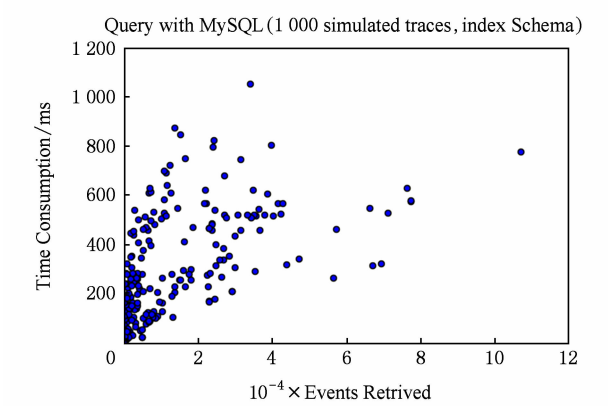


Fig. 7 Number of returned event and time consumed in simulated selection using MySQL

图 7 MySQL 上模拟筛选返回的事例数及所用时间

实验中采用了流式处理的 HBase 以及新的 Schema,归并了查询条件以及对应的事件结果. 实验结果显示线性扩展性较好. 而且,由于支持半结构化数据,也不用更新相关索引,容纳条目数多. 对于查询优化,虽然不支持在重建或者模拟时直接加入数据,但是对于性能的提升极为明显. 在实际应用中可以与重建或者模拟程序接口,实现数据的自动化增加与索引构建. 实验结果如图 8 所示:

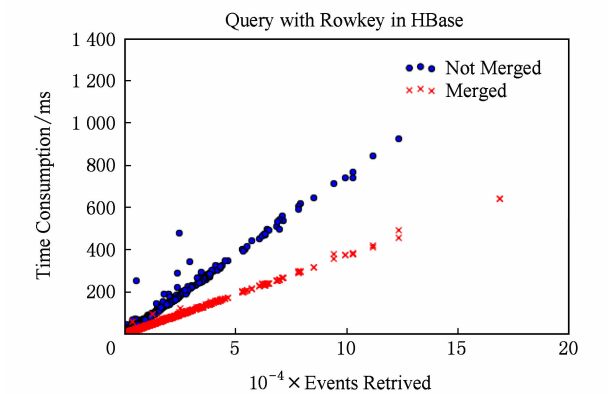


Fig. 8 Number of returned event and time consumed in simulated selection using HBase and new Schema

图 8 经过查询优化后 HBase 上模拟筛选返回的事例数及所用时间

4) 条件归并效果

由图 8 结果可以看出查询条件的归并对于性能

带来了很大的提升,主要原因是条件归并使得 HBase 中的条目数量大大降低.整体范围内看,物理实验的 1 400 万个事例数据,由于一个属性需要切分成为单独的一个条目,所以在未归并查询条件前在 HBase 中共有 4200 万条.进行查询条件归并后,仅剩 5 564 条.具体压缩比的效果与实验用例的关系图 9 所示:

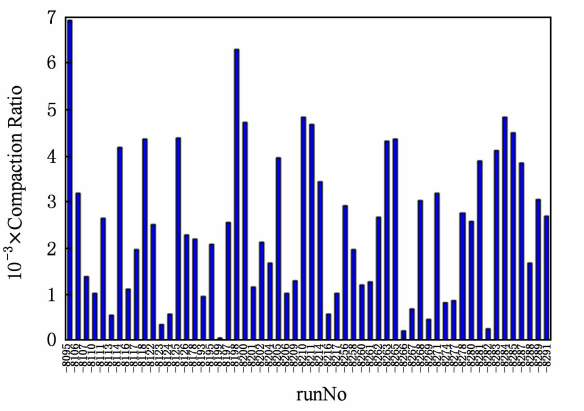


Fig. 9 The relationship between compaction ratio and runNo

图 9 压缩比的效果与实验用例的关系

其中,按照查询条件进行归并压缩的压缩率根据实验的不同而不同,平均能够到达 2 486,中位数为 2 402,75%的实验用例能够保证压缩率在 1 049 以上.

4 结 论

针对高能物理海量数据以及万亿级事例管理带来的挑战,本文在现有数据管理的基础上,提出采用基于文件存储和 NoSQL 事例特征索引融合的管理架构,设计一套面向事例的科学数据管理系统,与原有的文件级管理方式相比,可以实现高效的事例索引、筛选与快速处理,提高数据分析的效率.同时,由于具备面向事例的细化管理,可以克服原有数据访问局部性差的问题,实现事例级的缓存,提高数据 I/O 性能.基于事例特征索引数据库和面向事例的数据传输系统,可以支持计算任务实时访问远程站点数据,而不需要预先将所有文件传输到目标站点,使得分布式计算调度更加灵活,有利于提高资源利用率.验证系统选用了真实的 1 400 万高能物理实验事例数据和相关特征变量,实验结果说明基于特征查询和筛选具有可行性,经过查询优化的 HBase 系统具有非常好的查询性能和可扩展性.下一步工作,我们将针对更多的高能物理实验和更多的事例特征变量,开展大规模的研究和测试,最终实现万亿级别的事例索引和快速查询,提高数据处理效率,更好地支撑高能物理领域的科学发现活动.

参 考 文 献

[1] Girone M, Shiers J. WLCG operations and the first prolonged LHC run [C] //Proc of the 18th Int Conf on Computing in High Energy and Nuclear Physics. Taipei: Journal of Physics Conference Series, 2011, 331: 072014

[2] Han Jing, E Haihong, Le Guan, et al. Survey on NoSQL database [C] //Proc of the 6th Int Conf on Pervasive Computing and Applications. Piscataway, NJ: IEEE, 2011: 363-366

[3] Rademakers F, Brun R. ROOT: An object-oriented data analysis framework [J]. Nuclear Instruments & Methods in Physics Research, 1998, 389(1/2): 81-86

[4] Liu Beijiang. High performance computing activities in hadron spectroscopy at BESIII [C] //Proc of the 15th Int Workshop on Advanced Computing and Analysis Techniques in Physics. Beijing: Journal of Physics Conference Series, 2014, 523: 012008

[5] Bloomfield T, Sevior M. Index files for Belle II-very small skim containers [C] //Proc of the 22nd Int Conf on Computing in High Energy and Nuclear Physics. San Francisco: IOP Publishing, 2016

[6] Cranshaw J, Goosens L, Malon D, et al. Building a scalable event-level metadata service for ATLAS [C] //Proc of the 16th Int Conf on Computing in High Energy and Nuclear Physics. Victoria, Canada: Journal of Physics Conference Series, 2008, 119: 072012

[7] Sánchez J, FernándezCasaní A, Gonzalez de la Hoz S, et al. Distributed data collection for the ATLAS eventIndex [C] //Proc of the 21st Int Conf on Computing in High Energy and Nuclear Physics. Okinawa: Journal of Physics Conference Series, 2015, 664: 042046

[8] Lei Xiaofeng, Li Qiang, Sun Gongxing. HBase-based storage and analysis platform for high energy physics data [J]. Computer Engineering, 2015, 41(6): 49-55 (in Chinese)
(雷晓凤, 李强, 孙功星. 基于 HBase 的高能物理数据存储及分析平台[J]. 计算机工程, 2015, 41(6): 49-55)

[9] Becla J. Improving performance of object oriented databases, BaBar case studies [C] //Proc of the 11th Int Conf in High Energy and Nuclear Physics. Padova, Italy: InSPIRE, 2001: 410-413

[10] Düllmann D. Petabyte databases [C] //Proc of Int Conf on Management of Data. New York: ACM, 1999: 506-507

[11] Li W, Liu H, Deng Z, et al. The offline software for the BESIII experiment [C] //Proc of the 15th Int Conf in High Energy and Nuclear Physics. Mumbai, India: TIFR, 2006: 225-229

[12] Zou J H, Huang X T, Li W D, et al. SNiPER: An offline software framework for non-collider physics experiments [C] //Proc of the 21st Int Conf on Computing in High Energy and Nuclear Physics. Okinawa, Japan: Journal of Physics Conference Series, 2015, 664: 072053

[13] Vora M N. Hadoop-HBase for large-scale data [C] //Proc of Int Conf on Computer Science and Network Technology. Piscataway, NJ: IEEE, 2011: 601-605

[14] Chang F, Dean J, Ghemawat S, et al. Bigtable: A distributed storage system for structured data [J]. ACM Trans on Computer Systems, 2008: 26(2): 15-28

[15] Cheng Yaodong, Wang Lu, Huang Qiulan, et al. Design and optimization of storage system in HEP computing environment [J]. Computer Science, 2015, 42(1):54-58 (in Chinese)
(程耀东, 汪璐, 黄秋兰, 等. 高能物理计算环境中存储系统的设计与优化[J]. 计算机科学, 2015, 42(1): 54-58)

[16] Bonacorsi D, Ferrari T. WLCG service challenges and tiered architecture in the LHC era [G] //IFAE 2006: Proc of Italian Meeting on High Energy Physics. Berlin: Springer, 2007: 365-368

[17] Rehn J, Barrass T, Bonacorsi D, et al. PhEDEx data service [C] //Proc of the 17th Int Conf in High Energy and Nuclear Physics. Prague, Czech Republic: Journal of Physics Conference Series, 2010, 219: 062010

Cheng Yaodong, born in 1977. PhD and associate professor at the Institute of High Energy Physics, Chinese Academy of Sciences. His main research interests include distributed storage system, cloud computing and big data technologies.

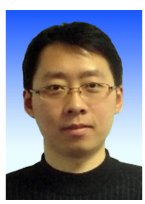
Zhang Xiao, born in 1991. PhD candidate of Xi'an Jiaotong University. His main research interests include cloud computing, anomaly detection and big data.

Wang Peijian, born in 1984. PhD, assistant professor of Xi'an Jiaotong University. Member of CCF. His main research interests include cloud computing and big data.

Zha Li, born in 1974. PhD. Associate professor at the Institute of Computing Technology, Chinese Academy of Sciences. Member of CCF. His main research interests include distributed system, big data management system and resource management in data center.

Hou Di, born in 1960. Associate professor of Xi'an Jiaotong University. Member of CCF. His main research interests include database, middleware, big data theory and technology.

Qi Yong, born in 1957. PhD and professor of Xi'an Jiaotong University. His main research interests include operating systems, distributed systems, cloud computing, big data system and system security.



Ma Can, born in 1984. Senior engineer of the Institute of Information Engineering, Chinese Academy of Sciences. His main research interests include cloud computing and big data.