BES离线刻度常数管理系统的研究*

张晓梅[1)] 马秋梅 毛泽普
(中国科学院高能物理研究所 北京 100039)

摘要 回顾了BESⅡ离线数据分析中离线刻度常数数据的文件管理方式;提出了利用数据库进行离线刻度常数进行管理的方法,详细描述了在LINUX操作系统下的实现过程,并讨论了用数据库管理BESⅢ离线刻度常数的可行性、优点以及需要进一步研究的问题.

关键词 北京谱仪 离线数据处理 刻度常数 数据库

1 引言

计算机工业的高速发展,数据管理技术已经从人工管理阶段,文件系统阶段发展到目前盛行的数据库管理阶段. 利用数据库管理数据,一方面可以实现记录为单位的数据共享,数据具有了更高的独立性,从而可以减少数据的冗余度,降低应用程序的研制和维护费用. 另一方面,数据库还提供了数据保护和并发控制的功能,保证了多用户操作下数据的完整性和一致性. 数据库管理系统为用户提供了方便的数据处理接口,也是数据库高速发展的一个重要因素. 北京谱仪(BES)离线数据管理基本上还处在文件系统管理阶段. 刻度常数文件的产生、存储与应用一直是利用ACSII文件管理系统加上人工进行管理. 这种方法管理复杂,耗费人力,还容易因为人为的因素出错;并且数据长时间的保存、查询、恢复以及安全性更是难以保证. 鉴于以上原因,对BES离线处理中的常数进行数据库管理成为急待解决的事情.

2 北京谱仪及其离线系统

BES是工作在北京正负电子对撞机(BEPC)上的大型通用磁谱仪[1],用于2.8—5.6GeV能区的高能物理研究. 北京谱仪的基本结构由顶点探测器(VC)、主漂移室(MDC)、簇射计数器桶部和端盖部分(BSC和ESC)、飞行时间计数器桶部和端盖部分(TOF)、亮度监测器(LUM)与μ子计数器(MU)以及束流管、磁铁、触发判选与电子学等子系统组成[2]. BES的数据处理流程可分为在线取数、离线数据处理和物理分析3个主要部分[3].

离线数据处理主要包括离线刻度、数据重建、和蒙特卡洛数据的产生和模拟,分别由DRUNK和SOBER两个主程序框架实现.

离线刻度过程是用Bhabha和双μ事例分别对各个子探测器作离线刻度,得到各个子探测器的离线刻度常数的过程,该过程对物理分析中最终获得的事例数据的准确度起着非常关键的作用. 本文主要是对离线刻度过程所产生的离线刻度常数的管理展开讨论.

3 BESⅡ离线刻度常数的文件管理方式

BESⅡ的离线刻度常数的文件管理方式基本上采用目录管理和索引管理. 首先,按照BES软件版本将离线刻度常数文件分类到不同目录下;其次,在同一个目录下存放多个索引文件(master文件),这些索引文件按照不同数据的序列号(RUN号)顺序进行命名,应用时,根据相应的索引文件对刻度常数文件进行索引查找.

2003-10-09收稿,2003-12-16收修改稿
* 国家自然科学基金(19991480)资助
1) E-mail:zhangxm@mail.ihep.ac.cn

BESⅡ离线刻度常数文件的管理和应用主要包括常数数据文件的产生,索引文件的产生和修改,以及刻度常数文件的读取3个过程. 图1和图2分别为产生和读取BES离线刻度常数文件示意图. 下面分别对这3个过程作介绍.

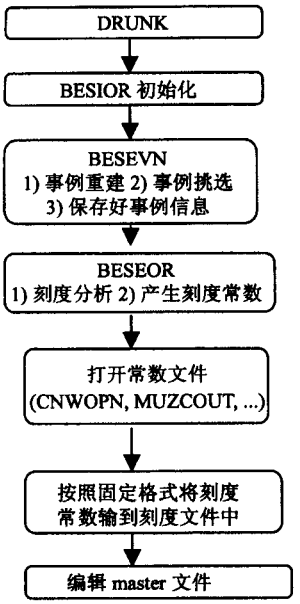

图1 BES刻度数据的产生流程

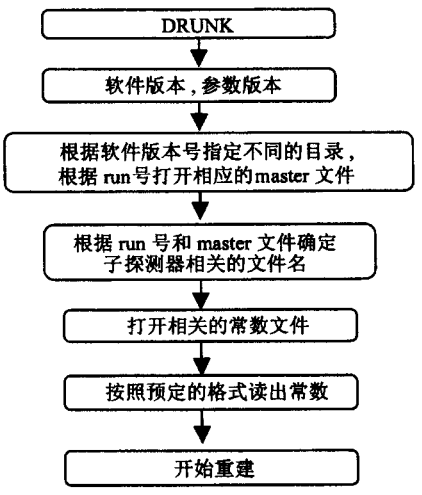

图2 BES刻度数据的读取流程

BESⅡ的离线刻度是通过BESⅡ主程序框架DRUNK驱动的. 首先利用BES用户接口程序BESIOR进行初始化,然后在BESEVN中对事例作循环,即重建事例和事例判选,并将判选得到的好事例保存起来,为刻度做准备. 最后在BESEOR中利用BESEVN保存的好事例信息作刻度,得到刻度常数,并产生相应格式的刻度常数文件. 刻度常数文件以相关的Run号和与子探测器相关的8个字符命名,文件后缀为.offcons.

索引文件的产生和修改是经过FORTRAN程序和人工操作完成. 索引文件实际是BES各个子探测器的刻度常数文件名的列表文件. 索引文件名为besoff.runconsX,其中X表示该索引文件所管理的RUN的范围;每个索引文件管理1000个RUN的刻度常数文件,即每个索引文件包含1000行. 每行的行号隐含了相应的RUN号;每行由120个字符组成,并划分成12组,每组表示一个子探测器的刻度常数名. 所以索引文件的号数、行号和每组的字符将所要操作的run号对应的各个子探测器的刻度常数文件名惟一的确定下来.

填写索引文件的过程是:1)产生相关数据(一定RUN号)的各个探测器的刻度常数文件,2)填写与数据相关的索引文件的相关行的各个探测器的刻度常数文件名,3)将修改好的索引文件和各个探测器的刻度常数文件移植到与BES软件版本相关的目录. 在BES运行期间,每天平均可取数达到20个run以上,因此每天都会有新的刻度常数文件产生,即每天都要对索引文件进行很多次更新.

在进行BES事例重建时,首先根据数据的序列号(RUN号)寻找相关的索引文件,从索引文件中与RUN号相关的行得到各个探测器的刻度常数文件名,然后分别打开各个刻度常数文并读取刻度常数.

上述过程,可以注意到文件管理方式的不足之处:

(1) 由于不同run号的数据的各个探测器的刻度常数文件完全由master文件确定,相同run号数据的不同的软件版本号下的各个探测器的刻度常数文件由不同子目录中的master文件确定,所以master文件的管理尤其重要,而且该工作大部分由人工完成. 由此可见这种管理办法的"记忆性"是很差的,或者由于计算机的故障,或者由于人为的失误,造成错误的几率是很大的.

(2) 由于master文件的创建和编辑等由人工直接参与,所以造成人力资源的浪费并工作效率低下.

(3) 应用程序直接依赖文件的数据格式和语言,灵活性很差,更新和维护都很困难.

(4) 数据存放方式复杂,管理很难做到一目了然,很难满足数据处理中的一些特殊要求,比如对某些数据在不同软件版本下的刻度常数的查询,或者同一软件版本下不同的刻度次数的刻度常数的和查询和重复.

如果采用数据库技术就可以很好解决上述问题.

4 BES 离线刻度常数的数据库系统设计和实现过程

一个数据库系统的 3 大基本组件:数据库,数据库管理系统(DBMS)和应用程序[4]. 它们之间的关系如图 3 所示. 应用程序是通过 DBMS 间接的访问数据库文件,用户不需要关心数据的物理存储格式,这样使得应用程序非常简单清晰. BES 离线刻度常数的数据库管理系统的设计需要 3 个方面:数据库管理系统的选取,数据库的设计以及应用程序对数据库应用编程接口(API)的调用.

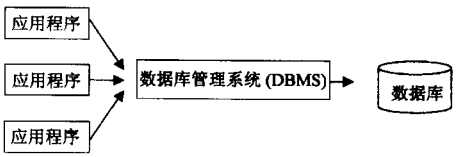

图 3　数据库,数据库管理系统(DBMS)和应用程序之间的关系图

4.1 数据库管理系统的选取

目前广泛应用的 DBMS 系统很多,主要分为两类:商业产品和开放源代码产品[5]. 商业产品有 oracle, sysbase, DB2 等. 开放源代码的产品有 mysql, postgresql, msql 等. 由于刻度参数数据没有复杂的相互关系,所需的数据库管理功能相对简单,实现的操作系统平台为 LINUX,因此从价格性能比考虑,首选开放源代码产品. Postgresql 结合了 mysql 和 linux 下 oracle8i 的优点,支持复杂的数据类型(如数组类型,大对象),支持复杂查询(如嵌套查询),支持事务处理,支持多种编程语言,以及完善的文档系统[6]. 这使得最终选择了 postgresql 作为数据库管理系统.

4.2 BES 离线刻度常数数据库设计

在 BES 刻度常数数据系统中,经过不同的刻度方法对 BES 的 5 个子探测器分别进行离线刻度后,得到 8 组功能不同的刻度常数,它们分别是:vc, mdchot, mdct0, tof, de/dx, esc, bsc, muon. 组之间相对独立,几乎没有相互关系;重建程序对每组的数据调用也是独立的. 因此为每组分别设计一个单独的表,分别对应于上述 8 组刻度常数.

接着根据表的查询情况定义表的结构. 由于 BES 事例重建程序分别需要根据刻度产生时的实验类型、刻度版本、以及 BES 软件的版本、刻度进行的时间,所适用的 run 号范围对表中的数据进行查询. 因此表必须考虑的基本属性包括:实验类型、run 号区间的开始值、结束值、刻度版本、软件版本和刻度时间,它们分别对应表的属性名为 exptyp, runfrm, runto, vercal, versft, createtime. 由于这 8 组刻度数据都是多维数据,因此在存储数据时避免过多的重复而采用数组类型,postgresql 数据库是支持数组类型的. 下面为定义表 muon 的 sql 语句,其中 azal, bzal, czal 都是表示为一维数组的 3 组刻度参数,muhead 为刻度参数的头文件说明,均由刻度过程给出.

```
create table muon (
  id int4 primary key,
  exptyp int4 NOT NULL,
  runfrm int4 NOT NULL,
  runto int4 NOT NULL,
  versft int4 NOT NULL,
  vercal int4 NOT NULL,
  createtime timestamp default now(),
  muzhead int[10],
  azcal double precision[1512],
  bzcal double precision[1512],
  czcal double precision[1512])
```

其他表的定义也与此类似,不再赘述.

4.3 BES 离线刻度常数数据库系统应用程序的实现过程

BES 离线刻度常数数据库系统应用程序主要包括两个过程:装库过程和读库过程. 图 4、图 5 显示了 BES 离线刻度程序中利用数据库写入和读出接口程序存储和读取数据的过程. 从图中可以看到这里 BES 离线刻度程序摆脱了文件管理方式下直接与物理文件打交道的繁琐易错的过程,只需要简单的调用两个数据库接口程序就能完成对数据的存储和读取. 这两个数据库的接口程序分别如下:

(1) Mu2db 接口程序功能是将 muon 探测器的刻度常数转入数据库中,其程序参数分别对应与 muon 表的各属性一一对应.

mu2db (pexpn, prunfrom, prunto, pversft, pvercal, header, azcal, bzcal, czcal)

(2) Db2mu 接口程序功能是根据提供的选择条件从库中取出所需要的 muon 刻度常数数据. 程序参数的前 4 项为输入的选择条件,后 4 个变量存放取出的刻度参数数据.

db2mu (pexpn, prunnum, pversft, pvercal, mu-

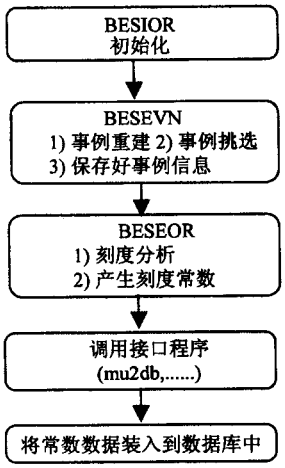

图4 BES刻度数据的入库流程

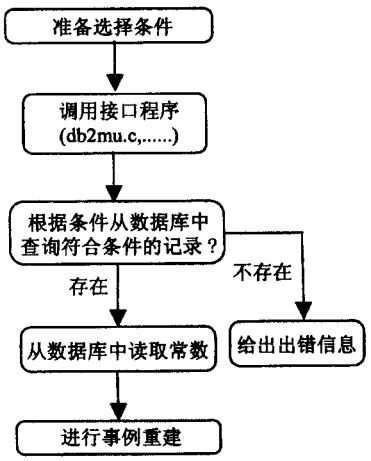

图5 BES刻度数据的读库流程

head, azcal, bzcal, czcal)

下面讨论这两个数据库接口程序的实现. 其实现主要有两个方面的问题需要解决:数据库编程接口(API)选择以及FORTRAN程序和C,C++程序兼容性问题. 数据库编程接口使用C API. FORTRAN程序的函数和C程序的函数的兼容性可以通过选择合适的编译器来解决. C程序中函数的函数名末尾加上一个下划线,其函数就可以安全在FORTRAN程序中所调用,这是编译器的一个约定. 因此C语言实现的接口函数分别如下:

mu2db_(int * pexpn, int * prunfrom, int * prunto, int * pversft, int * pvercal,

int header[10], float azcal[1512], float bzcal[1512], float czcal[1512])

该写入接口程序实现主要是利用CAPI执行SQL语句INSERT INTO.

db2mu_(int * pexpn, int * prunnum, int * pversft, int * pvercal, int muhead[10],

float azcal[1512], float bzcal[1512], float czcal[1512])

该读出接口程序实现主要是利用C API执行SQL语句SELECT.

5 BES刻度常数数据库系统运行和检测过程

1) 数据传输的正确性检查

数据的正确性检查主要有两个方面:入库和出库的一致性以及C程序和FORTRAN程序之间参数传递的一致性. 两个过程都采用差值法,即对经过过程前的数据和经过过程后的数据进行求差值,如果数据是准确,所有的差值应该等零,这种方法对于检测大批量的数据的正确比较有效,一目了然. 对所有输入输出环节都用差值法进行了检查,结果都全为零,从而验证了数据传输部分的正确性.

2) 重建结果的正确性检测

在确保了C函数和FORTRAN函数参数传递以及数据库输入输出之间的正确性之后,开始检验经过重建过程后结果的正确性. 用小批数据对各个子探测器进行刻度,产生刻度参数分别采用文件和数据库方式分别管理常数数据,而后比较这两种方式管理下的重建结果. 我们进行了一个run的重建,采用的是Run24050,把重建生成的TRKLST数据进行比较. 如每个事例产生的径迹数、每条径迹的各种参量(如动量、能量)、TOF的时间信息、μ的判选信息等. 典型的两个参量的比较如图6所示(径迹的动量和径迹数分布). 左面的图是原来文件管理方式下的重建结果,右边的图是数据库管理方式下的重建结果. 从图中可以看出,利用文件管理和利用数据库管理的刻度常数重建的结果是完全一致的,这个结果可以保证刻度常数的管理可以从文件管理方式安全地过渡到数据库管理方式.

3) 数据的存取效率检测检查

数据的存取效率也是评价数据库管理方式的一个重要指标. 在验证了数据正确性之后,很有必要对数据库的存取效率做检验. 这次使用大批量的数据,即刻度1000个run(run24100—25100)的数据,同样对数据进行文件方式和数据库方式分别进行管理,在保证运行环境相同的情况下,运行结果表明除去人工参与的时间,用数据库管理常数文件进行事

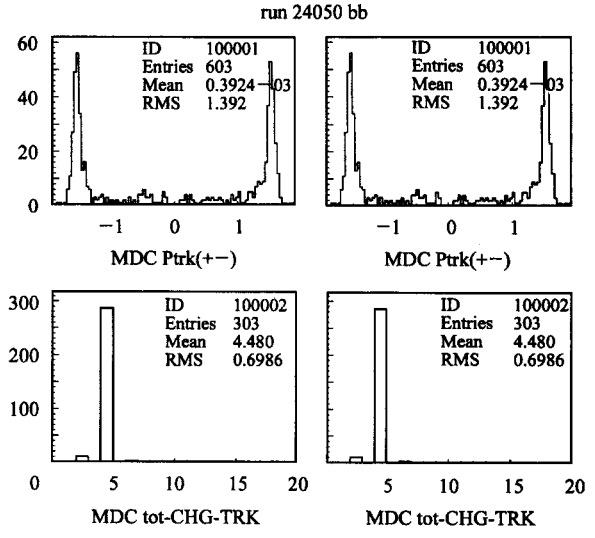

图 6　重建结果检测

例重建的机器运行的时间与文件管理方式下的机器运行时间基本相当.

综合以上检测,考虑到文件管理方式下需要大量人工参与管理数据的因素,可以证明数据库管理方式下的工作效率和正确率明显高于文件管理方式.

6　讨论

经过这次实验,可以证明数据库方式管理 BES 离线数据是切实可行的. 但是,本次实验的测试环境比较局限,仅限于在单一数据库,本地机器上,因此对于更加复杂和广泛的应用还需作进一步研究和尝试. 为了以后 BESⅢ中更有效的利用数据库来管理刻库数据,还需要做以下几个方面的努力:

(1) 充分利用网络,将数据库的应用网络化. 建立数据库应用体系的三层模型,使得用户可以直接通过网页可以对数据库进行查询,修改等操作.

(2) 考虑到以后的不断积累的海量数据,可以研究采用分布式数据库方式来解决.

(3) 为了使得对数据库的选择具有灵活性,可以引入中间驱动程序如 ODBC(Open Data Base Connectivity),从而使得应用程序和具体的数据库管理系统之间保持相对独立性,即改变数据库管理系统时,不需要更改你的应用程序.

(4) 在保证多用户操作下,数据库安全性考虑.

参考文献(References)

1 ZHENG Zhi-Peng, ZHU Yong-Sheng. Beijing Spectrometer and e^+e^- Physics. Nanning: Science and Technology Publishing House of Guangxi, 1998. 34—53(in Chinese)
(郑志鹏,朱永生. 北京谱仪正负电子物理. 南宁:广西科学技术出版社,1998. 34—58)

2 DING Hui-Liang et al (BES Collaboration). HEP&NP, 1992, **16**(9): 769—789(in Chinese)
(丁慧良等(BES 合作组). 高能物理与核物理,1992,**16**(9): 769—789)

3 WANG Tai-Jie et al. Nuclear Electronics & Detection Technology, 1990, **10**(6):343(in Chinese)
(王泰杰等. 核电子学与探测技术,1990,**10**(6):343)

4 YU Pang-Xiang, SHEN Jin-Fa. Principles of DataBase System. Publishing House of Tsinghua University, 1988(in Chinese)
(俞盘祥,沈金发. 数据库系统原理. 清华大学出版社,1988)

5 Michele Petrovsky et al. M. Linux Database Bible. Publishing House of Electronics Industry, 2002(in Chinese)
(耿岳,赵友兵译. Linux 数据库宝典. 电子工业出版社,2002)

6 Barry Stinson. PostgresSQL Essential Reference. Publishing House of Renmin postage, 2002(in Chinese)
(云巅工作室译. PostgreSQL 必备参考手册. 人民邮电出版社,2002)

Research on the BES Offline Calibration Data Management System*

ZHANG Xiao-Mei[1)]　MA Qiu-Mei　MAO Ze-Pu

(Institute of High Energy Physics, CAS, Beijing 100039, China)

Abstract　The old file management system for BES offline calibration is reviewed. A new method of using database to manage BES offline calibration constant data is presented, and its whole implementation in Linux is described in detail. The feasibility, advantages and further improvements are also discussed.

Key words　BES, offline data management, calibration constant, database

Received 9 October 2003, Revised 16 December 2003

* Supported by NSFC(19991480)

1) E-mail:zhangxm@mail.ihep.ac.cn