(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 107451012 A
(43)申请公布日 2017.12.08

(21)申请号 201710482647.6

(22)申请日 2014.07.04

(62)分案原申请数据
201410317676.3 2014.07.04

(71)申请人 华为技术有限公司
地址 518129 广东省深圳市龙岗区坂田华为总部办公楼

(72)发明人 夏命榛 史云龙

(51)Int.Cl.
G06F 11/14(2006.01)

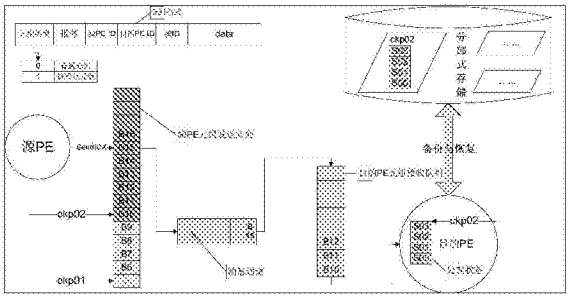
权利要求书3页 说明书15页 附图7页

(54)发明名称

一种数据备份方法及流计算系统

(57)摘要

本发明提供一种数据备份方法及流计算系统，该方法包括：目的PE从自身的接收队列中读取第一检查点元组，若判断所述第一检查点元组的批号与当前批号相同，且与所述第一检查点元组具有相同批号的所有元组都已处理完毕，则将自身的状态数据备份至所述流计算系统的分布式存储器中。本发明提供的数据备份方法和流计算系统，采用异步备份的方式，使得数据备份不受PE之间数据传递时延的影响，同时通过设置元组的批次，通过批号的比较，使得同一批号的所有元组到齐之后再进行状态备份，保证了数据备份的一致性。



CN 107451012 A

1.一种流计算系统中的数据备份方法，所述流计算系统包括多个执行单元PE，用于对待处理的元组进行处理，所述多个执行单元包括：源PE和目的PE；其特征在于，所述数据备份方法包括：

所述目的PE接收所述源PE发送的多个元组并加入自身的接收队列，所述多个元组中的每个元组都携带有表示该元组批次的批号；所述多个元组包括多个普通元组和多个检查点元组，不同的检查点元组具有不同的批号，所述接收队列中的两个检查点元组之间间隔有多个具有相同的批号的普通元组，且每个检查点元组的批号与其相邻的前一个普通元组的批号相同；

所述目的PE从所述接收队列中读取第一检查点元组，所述第一检查点元组指示所述目的PE进行状态数据备份；

所述目的PE判断与所述第一检查点元组具有相同批号的所有普通元组是否都已处理完毕；

若与所述第一检查点元组具有相同批号的所有普通元组都已处理完毕，则所述目的PE将自身的状态数据进行备份。

2.根据权利要求1所述的数据备份方法，其特征在于，还包括：

所述目的PE从所述接收队列中读取第二检查点元组，所述第二检查点元组指示所述目的PE进行状态数据恢复；

所述目的PE加载自身备份的状态数据，并基于所述备份的状态数据进行状态恢复。

3.根据权利要求1所述的数据备份方法，其特征在于，所述目的PE中保存有检查点状态信息，所述检查点状态信息包括：当前批号以及元组到齐标记；所述当前批号指示所述目的PE当前处理的元组的批号；

所述目的PE判断与所述第一检查点元组具有相同批号的所有普通元组是否都已处理完毕包括：

所述目的PE根据所述检查点状态信息中的所述当前批号和所述元组到齐标记确定与所述第一检查点元组具有相同批号的所有普通元组是否都已处理完毕。

4.根据权利要求1所述的数据备份方法，其特征在于，还包括：

所述目的PE从所述接收队列中读取第三检查点元组，所述第三检查点元组指示所述目的PE进行状态数据清理；

所述目的PE清理自身备份的状态数据。

5.根据权利要求4所述的数据备份方法，其特征在于，还包括：所述目的PE清理所述检查点状态信息。

6.根据权利要求5所述的数据备份方法，其特征在于，所述清理所述检查点状态信息包括：

将所述检查点状态信息中的所述当前批号加1，并将所述元组到齐标记清零。

7.根据权利要求1-6任一项所述的数据备份方法，其特征在于，所述流计算系统还包括：分布式存储器；

所述目的PE将自身的状态数据进行备份包括：

所述目的PE通过调用第一接口将当前的状态数据缓存到本地内存；

所述目的PE通过第二接口调用备份与恢复模块，以使所述备份与恢复模块启动备份线

程，将所述本地内存中的状态数据备份至所述分布式存储器。

8.根据权利要求7所述的数据备份方法，其特征在于，所述目的PE加载自身备份的状态数据，并基于所述备份的状态数据进行状态恢复包括：

所述目的PE从所述分布式存储器中加载所述目的PE最近一次备份的状态数据，并基于所述最近一次备份的状态数据进行状态恢复。

9.根据权利要求3-6任一项所述的数据备份方法，其特征在于，还包括：

所述目的PE从所述接收队列中读取第一普通元组；

如果所述第一普通元组的批号等于所述当前批号，则对所述第一普通元组进行处理；

如果所述第一普通元组的批号大于所述当前批号，则将所述第一普通元组加入缓存队列，并更新所述检查点状态信息中的元组到齐标记。

10.一种流计算系统中的目的执行单元，所述流计算系统包括源执行单元和所述目的执行单元；其特征在于，所述目的执行单元包括：

接收队列，用于缓存所述源PE发送的多个元组，所述多个元组中的每个元组都携带有表示该元组批次的批号；所述多个元组包括多个普通元组和多个检查点元组，不同的检查点元组具有不同的批号，所述接收队列中的两个检查点元组之间间隔有多个具有相同的批号的普通元组，且每个检查点元组的批号与其相邻的前一个普通元组的批号相同；

业务数据处理模块，用于从所述接收队列中读取元组并对读取到的元组进行处理；

备份与恢复模块，用于当所述业务数据处理模块读取到的元组为指示所述目的执行单元进行状态数据备份的第一检查点元组时，判断与所述第一检查点元组具有相同批号的所有普通元组是否都已处理完毕；若与所述第一检查点元组具有相同批号的所有普通元组都已处理完毕，则将所述目的执行单元的状态数据进行备份。

11.根据权利要求10所述的目的执行单元，其特征在于，所述备份与恢复模块，还用于当所述业务数据处理模块读取到的元组为指示所述目的执行单元进行状态数据恢复的第二检查点元组时，加载自身备份的状态数据，并基于所述备份的状态数据进行状态恢复。

12.根据权利要求10所述的目的执行单元，其特征在于，所述目的执行单元中保存有检查点状态信息，所述检查点状态信息包括：当前批号以及元组到齐标记；所述当前批号指示所述业务数据处理模块当前处理的元组的批号；

所述备份与恢复模块具体根据所述检查点状态信息中的所述当前批号和所述元组到齐标记确定与所述第一检查点元组具有相同批号的所有普通元组是否都已处理完毕。

13.根据权利要求10所述的目的执行单元，其特征在于，所述备份与恢复模块还用于，当所述业务数据处理模块读取到用于指示所述目的执行单元进行状态数据清理的第三检查点元组时，清理自身备份的状态数据。

14.根据权利要求10-13任一项所述的目的执行单元，其特征在于，所述备份与恢复模块具体用于：

通过调用第一接口将所述目的执行单元当前的状态数据缓存到本地内存；

启动备份线程，以使所述备份线程将所述本地内存中的状态数据备份至所述流计算系统的分布式存储器。

15.根据权利要求14所述的目的执行单元，其特征在于，所述备份与恢复模块具体用于：

从所述分布式存储器中加载自身最近一次备份的状态数据，并基于所述最近一次备份的状态数据进行状态恢复。

16.根据权利要求12至15任一项所述的目的执行单元，其特征在于，所述业务数据处理模块具体用于：当从所述接收队列中读取的第一普通元组的批号等于所述当前批号时，则对所述第一普通元组进行处理；当所述第一普通元组的批号大于所述当前批号时，则将所述第一普通元组缓存，并更新所述检查点状态信息中的元组到齐标记。

17.一种流计算系统，其特征在于，包括：多个执行单元PE，用于对待处理的元组进行处理，所述多个执行单元包括：源PE和目的PE；其中，

所述源PE，用于将自身的发送队列中缓存的元组发送给所述目的PE；

所述目的PE，用于接收所述源PE发送的所述多个元组并加入自身的接收队列，所述多个元组中的每个元组都携带有表示该元组批次的批号；所述多个元组包括多个普通元组和多个检查点元组，不同的检查点元组具有不同的批号，所述接收队列中的两个检查点元组之间间隔有多个具有相同的批号的普通元组，且每个检查点元组的批号与其相邻的前一个普通元组的批号相同；从所述接收队列中读取第一检查点元组，所述第一检查点元组指示所述目的PE进行状态数据备份；若确定与所述第一检查点元组具有相同批号的所有普通元组都已处理完毕，则将自身的状态数据进行备份。

18.根据权利要求17所述的流计算系统，其特征在于，

所述源PE还用于，接收用于状态数据备份的第一检查点命令，根据所述第一检查点命令生成所述第一检查点元组，将生成的检查点元组加入所述源PE的发送队列。

19.根据权利要求17所述的流计算系统，其特征在于，

所述源PE还用于，接收用于数据恢复的第二检查点命令，根据所述第二检查点命令生成用于指示所述目的PE进行状态数据恢复的第二检查点元组，将生成的第二检查点元组加入所述源PE的发送队列，将所述发送队列中缓存的所述第二检查点元组发送至所述目的PE；

所述目的PE还用于，从所述接收队列中读取所述第二检查点元组，根据所述第二检查点元组加载自身备份的状态数据，并基于所述状态数据进行状态恢复。

20.根据权利要求17所述的流计算系统，其特征在于，

所述源PE还用于，接收用于状态数据清理的第三检查点命令，根据所述第三检查点命令生成第三检查点元组，将生成的第三检查点元组加入所述源PE的发送队列，将所述发送队列中缓存的所述第三检查点元组通过消息通道发送至所述目的PE；

所述目的PE还用于，从所述接收队列中读取所述第三检查点元组，并根据所述第三检查点元组清理自身备份的状态数据。

21.根据权利要求17-20任一项所述的流计算系统，其特征在于，所述流计算系统还包括：分布式存储器；

所述目的PE具体用于，通过调用第一接口将自身当前的状态数据缓存到本地内存；通过第二接口调用备份与恢复模块，以使所述备份与恢复模块启动备份恢复线程，将所述本地内存中的状态数据备份至所述分布式存储器。

22.根据权利要求21所述的流计算系统，其特征在于，所述目的PE具体用于，从所述分布式存储器中加载自身备份的状态数据。

一种数据备份方法及流计算系统

技术领域

[0001] 本发明涉及计算机技术领域，尤其涉及一种数据备份方法以及一种流计算系统。

背景技术

[0002] 近年来，数据密集型业务已经得到越来越广泛的应用，这些业务的实例包括金融服务、网络监控、电信数据管理、传感检测等等。数据密集型业务产生的数据具有数据量大、快速、时变的特点，流计算系统在接收流数据时就开始对其进行处理，以保证实时性。如图1所示，流计算系统通常包括一个主控节点(Master)和多个工作节点(worker)，主控节点主要负责调度和管理各个工作节点，而工作节点是承载实际的数据处理操作的逻辑实体，工作节点具体通过调用若干个执行单元(PE，Process Element)来对数据进行处理，PE是业务逻辑的物理执行单元。

[0003] 可以看出，流计算系统实质上是一个分布式集群系统，因此系统出现异常的概率较高，流计算系统发生故障可能会导致业务中断或状态数据丢失，为了保证流计算系统的可靠性，现有技术通常采用多节点备份机制，如图2所示，周期性的将每个工作节点中的PE的状态数据以及业务数据备份至其他工作节点的内存，当某个工作节点出现故障，则迁移到备份的工作节点继续进行数据处理。

[0004] 由于流计算系统是分布式数据处理系统，工作节点中的每个PE可能会处理多条数据流中的数据，同时，同一数据可能会同时被不同的PE处理，流计算系统中数据处理的并发性和无序性，以及PE之间数据传递的时延，会导致采用现有技术这种整体同步备份的方式，数据备份的一致性得不到保证。

发明内容

[0005] 本发明实施例提供一种数据备份方法及流计算系统，用以保证分布式流计算系统中数据备份的一致性。

[0006] 第一方面，本发明实施例提供了一种数据备份方法，应用于流计算系统中，所述流计算系统包括多个工作节点，所述多个工作节点通过调用多个执行单元PE来对待处理的元组进行处理，所述多个执行单元包括：源PE和目的PE；所述源PE将自身的发送队列中缓存的元组发送到所述目的PE的接收队列中，所述目的PE读取自身的接收队列中的元组并进行处理；所述源PE的发送队列中缓存的元组包括普通元组和检查点元组，且每个元组携带有用于表示该元组批次的批号；其中，检查点元组用于指示PE进行数据备份或数据恢复；所述源PE的发送队列中不同的检查点元组具有不同的批号，处于相邻的两个检查点元组之间的普通元组具有相同的批号，且每个检查点元组的批号与其相邻的一个普通元组的批号相同；所述数据备份方法包括：

[0007] 所述目的PE从自身的接收队列中读取第一检查点元组，所述第一检查点元组用于指示所述目的PE进行状态数据备份；

[0008] 所述目的PE判断所述第一检查点元组的批号与当前批号是否相同，以及与所述第

一检查点元组具有相同批号的所有普通元组是否都已处理完毕;所述当前批号为所述目的PE当前处理的普通元组的批号;

[0009] 若所述第一检查点元组的批号与所述当前批号相同,且与所述第一检查点元组具有相同批号的所有普通元组都已处理完毕,则所述目的PE将自身的状态数据进行备份;其中,所述目的PE的状态数据包括所述目的PE在处理所述普通元组过程中产生的数据。

[0010] 在第一方面的第一种可能的实现方式中,所述数据备份方法还包括:

[0011] 所述目的PE从自身的接收队列中读取第二检查点元组,所述第二检查点元组用于指示所述目的PE进行状态数据恢复;

[0012] 所述目的PE加载自身备份的状态数据,并基于所述备份的状态数据进行状态恢复和数据回放。

[0013] 结合第一方面,或者第一方面第一种可能的实现方式,在第二种可能的实现方式中,所述目的PE中保存有检查点状态信息,所述检查点状态信息包括:所述当前批号以及元组到齐标记;

[0014] 所述目的PE判断所述第一检查点元组的批号与当前批号是否相同,以及与所述第一检查点元组具有相同批号的所有普通元组是否都已到齐,包括:

[0015] 所述目的PE比较所述第一检查点元组的批号与所述检查点状态信息中包含的当前批号是否相等,以及根据所述检查点状态信息中的元组到齐标记确定与所述第一检查点元组具有相同批号的所有普通元组是否都已处理完毕。

[0016] 结合第一方面第二种可能的实现方式,在第三种可能的实现方式中,还包括:

[0017] 所述目的PE从自身的接收队列中读取第三检查点元组,所述第三检查点元组用于指示所述目的PE进行状态数据清理;

[0018] 所述目的PE清理自身备份的状态数据,并清理所述检查点状态信息。

[0019] 结合第一方面第三种可能的实现方式,在第四种可能的实现方式中,所述清理所述检查点状态信息包括:

[0020] 将所述检查点状态信息中的当前批号加1,并将元组到齐标记清零。

[0021] 结合第一方面,或者第一方面第一至第四种任意一种可能的实现方式,在第五种可能的实现方式中,所述流计算系统还包括:分布式存储器;所述目的PE将自身的状态数据进行备份,包括:

[0022] 所述目的PE通过调用第一接口将当前的状态数据缓存到本地内存;

[0023] 所述目的PE通过第二接口调用备份与恢复模块,以使所述备份与恢复模块启动备份线程,将所述本地内存中的状态数据备份至所述分布式存储器。

[0024] 结合第一方面第五种可能的实现方式,在第六种可能的实现方式中,所述目的PE加载自身备份的状态数据,并基于所述备份的状态数据进行状态恢复和数据回放,包括:

[0025] 所述目的PE从所述分布式存储器中加载自身最近一次备份的状态数据,并基于所述最近一次备份的状态数据进行状态恢复和数据回放。

[0026] 结合第一方面第二到第六种中任意一种可能的实现方式,在第七种可能的实现方式中,还包括:

[0027] 目的PE从自身的接收队列中读取普通元组;

[0028] 将该普通元组的批号与当前批号进行比较,如果该元组的批号等于当前批号,则

对该普通元组进行处理。

[0029] 结合第一方面第七种可能的实现方式，在第八种可能的实现方式中，还包括：

[0030] 如果该普通元组的批号小于当前批号，则丢弃该普通元组，并从所述接收队列中读取下一个元组。

[0031] 结合第一方面第七种可能的实现方式以及第八种可能的实现方式中的任意一种可能的实现方式，在第九种可能的实现方式中，还包括：

[0032] 如果该普通元组的批号大于当前批号，则将所述普通元组加入缓存队列，并更新所述检查点状态信息中的元组到齐标记。

[0033] 第二方面，本发明实施例提供了一种数据备份方法，应用于流计算系统中，所述流计算系统包括多个工作节点，所述多个工作节点通过调用多个执行单元PE来对待处理的元组进行处理，所述多个执行单元包括：源PE和目的PE；所述源PE将自身的发送队列中缓存的元组发送到所述目的PE的接收队列中，所述目的PE读取自身的接收队列中的元组并进行处理；所述源PE的发送队列中缓存的元组包括普通元组和检查点元组，且每个元组携带有用于表示该元组批次的批号；其中，检查点元组用于指示PE进行数据备份或数据恢复；所述源PE的发送队列中不同的检查点元组具有不同的批号，处于相邻的两个检查点元组之间的普通元组具有相同的批号，且每个检查点元组的批号与其相邻的一个普通元组的批号相同；所述数据备份方法包括：

[0034] 源PE接收用于状态数据备份的第一检查点命令；

[0035] 源PE根据所述第一检查点命令生成第一检查点元组，并将生成的检查点元组加入所述源PE的发送队列；

[0036] 所述源PE将所述发送队列中缓存的所述第一检查点元组发送至目的PE的接收队列，以使所述目的PE从所述接收队列中读取所述第一检查点元组之后，若判断所述第一检查点元组的批号与当前批号相同，且与所述第一检查点元组具有相同批号的所有普通元组均已处理完毕时，将所述目的PE当前的状态数据进行备份；其中，所述目的PE的状态数据包括所述目的PE在处理所述普通元组过程中产生的数据。

[0037] 在第二方面的第一种可能的实现方式中，所述流计算系统还包括：用于管理所述多个工作节点的主控节点；所述多个工作节点包括检查点PE所处的工作节点；所述源PE接收用于状态数据备份的第一检查点命令，包括：

[0038] 所述源PE接收所述流计算系统的主控节点或者所述检查点PE发送的第一检查点命令。

[0039] 第三方面，本发明实施例提供了一种流计算系统中的目的执行单元，所述流计算系统包括源执行单元和所述目的执行单元；所述源执行单元用于将自身的发送队列中缓存的元组发送到所述目的执行单元的接收队列，所述源执行单元的发送队列中缓存的元组包括普通元组和检查点元组，且每个元组携带有用于表示该元组批次的批号；其中，检查点元组用于指示执行单元进行数据备份或数据恢复；所述源执行单元的发送队列中不同的检查点元组具有不同的批号，处于相邻的两个检查点元组之间的普通元组具有相同的批号，且每个检查点元组的批号与其相邻的一个普通元组的批号相同；

[0040] 所述目的执行单元包括：业务数据处理模块，用于从所述目的执行单元的接收队列中读取元组并对读取到的元组进行处理；

[0041] 备份与恢复模块，用于当所述业务数据处理模块读取到的元组为用于指示所述目的执行单元进行状态数据备份的第一检查点元组时，判断所述第一检查点元组的批号与当前批号是否相同，以及与所述第一检查点元组具有相同批号的所有普通元组是否都已处理完毕；所述当前批号为所述业务数据处理模块当前处理的普通元组的批号；若所述第一检查点元组的批号与所述当前批号相同，且与所述第一检查点元组具有相同批号的所有普通元组都已处理完毕，则将所述目的执行单元的状态数据进行备份；其中，所述目的执行单元的状态数据包括所述业务数据处理模块在处理元组的过程中产生的数据。

[0042] 在第三方面的第一种可能的实现方式中，所述备份与恢复模块，还用于当所述业务数据处理模块读取到的元组为用于指示所述目的执行单元进行状态数据恢复的第一检查点元组时，加载自身备份的状态数据，并基于所述备份的状态数据进行状态恢复和数据回放。

[0043] 结合第三方面，或者第三方面第一种可能的实现方式，在第二种可能的实现方式中，所述目的PE中保存有检查点状态信息，所述检查点状态信息包括：所述当前批号以及元组到齐标记；

[0044] 在判断所述第一检查点元组的批号与当前批号是否相同，以及与所述第一检查点元组具有相同批号的所有普通元组是否都已到齐的方面，所述备份与恢复模块具体用于：

[0045] 比较所述第一检查点元组的批号与所述检查点状态信息中包含的当前批号是否相等，以及根据所述检查点状态信息中的元组到齐标记确定与所述第一检查点元组具有相同批号的所有普通元组是否都已处理完毕。

[0046] 结合第三方面，或者第三方面第一至第二种任意一种可能的实现方式，在第三种可能的实现方式中，所述流计算系统还包括：分布式存储器；在将所述目的执行单元的状态数据进行备份的方面，所述备份与恢复模块具体用于：

[0047] 通过调用第一接口将所述目的PE当前的状态数据缓存到本地内存；

[0048] 启动备份线程，以使所述备份线程将所述本地内存中的状态数据备份至所述分布式存储器。

[0049] 结合第三方面第三种可能的实现方式，在第四种可能的实现方式中，在加载自身备份的状态数据，并基于所述备份的状态数据进行状态恢复和数据回放的方面，所述备份与恢复模块具体用于：

[0050] 从所述分布式存储器中加载自身最近一次备份的状态数据，并基于所述最近一次备份的状态数据进行状态恢复和数据回放。

[0051] 第四方面，本发明实施例提供了一种流计算系统中的源执行单元，所述流计算系统包括所述源执行单元和目的执行单元；所述源执行单元的发送队列中缓存有待发送给所述目的执行单元的元组，且所述待发送的元组包括普通元组和检查点元组，且每个元组携带有用于表示该元组批次的批号；其中，检查点元组用于指示执行单元进行数据备份或数据恢复；所述源执行单元的发送队列中不同的检查点元组具有不同的批号，处于相邻的两个检查点元组之间的普通元组具有相同的批号，且每个检查点元组的批号与其相邻的一个普通元组的批号相同；所述源执行单元包括：

[0052] 检查点模块，用于接收用于状态数据备份的第一检查点命令；根据所述第一检查点命令生成第一检查点元组；

[0053] 发送模块，用于将所述检查点模块生成的检查点元组加入所述源执行单元的发送队列；并将所述发送队列中缓存的元组发送至所述目的执行单元的接收队列，以使所述目的执行单元从所述接收队列中读取所述第一检查点元组之后，若判断所述第一检查点元组的批号与当前批号相同，且与所述第一检查点元组具有相同批号的所有普通元组均已处理完毕时，将所述目的执行单元当前的状态数据进行备份；其中，所述目的执行单元的状态数据包括所述目的执行单元在处理元组的过程中产生的数据。

[0054] 第五方面，本发明实施例提供了一种流计算系统，包括：多个工作节点，所述多个工作节点通过调用多个执行单元(PE)来对元组进行处理，所述多个执行单元包括：源PE和目的PE；其中，所述源PE，用于将自身的发送队列中缓存的元组发送到所述目的PE的接收队列中；所述目的PE，用于读取自身的接收队列中的元组并进行处理；其中，所述源PE的发送队列中缓存的元组包括普通元组和检查点元组，且每个元组携带有用于表示该元组批次的批号；其中，检查点元组用于指示PE进行数据备份或数据恢复；所述源PE的发送队列中不同的检查点元组具有不同的批号，处于相邻的两个检查点元组之间的普通元组具有相同的批号，且每个检查点元组的批号与其相邻的一个普通元组的批号相同；其中：

[0055] 所述源PE，还用于接收用于状态数据备份的第一检查点命令，根据所述第一检查点命令生成第一检查点元组，将生成的检查点元组加入所述源PE的发送队列，将所述发送队列中缓存的所述第一检查点元组发送至所述目的PE的接收队列；

[0056] 所述目的PE，还用于从自身的接收队列中读取所述第一检查点元组，若判断所述第一检查点元组的批号与当前批号相同，且与所述第一检查点元组具有相同批号的所有普通元组都已处理完毕，则将自身的状态数据进行备份；其中，所述当前批号为所述目的PE当前处理的普通元组的批号；所述目的PE的状态数据包括所述目的PE在处理所述普通元组过程中产生的数据。

[0057] 在第五方面的第一种可能的实现方式中，

[0058] 所述源PE还用于，接收用于数据恢复的第二检查点命令，根据所述第二检查点命令生成用于指示所述目的PE进行状态数据恢复的第二检查点元组，将生成的第二检查点元组加入所述源PE的发送队列，将所述发送队列中缓存的所述第二检查点元组发送至所述目的PE的接收队列；

[0059] 所述目的PE还用于，从自身的接收队列中读取所述第二检查点元组，根据所述第二检查点元组加载自身备份的状态数据，并基于所述状态数据进行状态恢复和数据回放。

[0060] 结合第五方面，或者第五方面第一种可能的实现方式，在第二种可能的实现方式中，

[0061] 所述源PE还用于，接收用于状态数据清理的第三检查点命令，根据所述第三检查点命令生成第三检查点元组，将生成的第三检查点元组加入所述源PE的发送队列，将所述发送队列中缓存的所述第三检查点元组通过消息通道发送至所述目的PE的接收队列；

[0062] 所述目的PE还用于，从自身的接收队列中读取所述第三检查点元组，并根据所述第三检查点元组清理自身备份的状态数据。

[0063] 结合第五方面，或者第五方面第一至第二种任意一种可能的实现方式，在第三种可能的实现方式中，所述流计算系统还包括：分布式存储器；

[0064] 在将自身的状态数据进行备份的方面，所述目的PE具体用于，通过调用第一接口

将自身当前的状态数据缓存到本地内存;通过第二接口调用备份与恢复模块,以使所述备份与恢复模块启动备份恢复线程,将所述本地内存中的状态数据备份至所述分布式存储器。

[0065] 结合第五方面第三种可能的实现方式,在第四种可能的实现方式中,在加载自身备份的状态数据的方面,所述目的PE具体用于,从所述分布式存储器中加载自身备份的状态数据。

[0066] 结合第五方面,或者第五方面第一至第四种任意一种可能的实现方式,在第五种可能的实现方式中,所述流计算系统还包括:主控节点,用于向所述源PE发送所述第一检查点命令、第二检查点命令和第三检查点命令中的至少一个。

[0067] 结合第五方面,或者第五方面第一至第四种任意一种可能的实现方式,在第六种可能的实现方式中,所述流计算系统还包括:主控节点;所述多个工作节点包括检查点PE所处的工作节点;

[0068] 所述主控节点,用于向所述检查点PE下发用于数据备份的备份命令、用于数据恢复的恢复命令或者用于状态数据清理的数据清理命令;所述检查点PE用于,在接收到所述主控节点发送的备份命令后,向所述源PE发送所述第一检查点命令;或者,在接收到所述主控节点发送的恢复命令后,向所述源PE发送所述第二检查点命令;或者,在接收到所述主控节点发送的数据清理命令后,向所述源PE发送所述第三检查点命令。

[0069] 由上述技术方案可知,本发明实施例提供的数据备份方法和流计算系统,通过检查点元组来触发PE进行数据备份,PE从接收队列中读取到检查点元组之后,再执行备份操作,各个PE的备份操作不需要同步,使得数据备份不受PE之间数据传递时延的影响;同时通过设置元组的批次,以及批号的比较,使得同一批号的所有元组到齐之后再进行状态备份,避免了流计算系统中数据处理的并发性和无序性对数据备份一致性的影响,从而保证了数据备份的一致性。

附图说明

[0070] 为了更清楚地说明本发明的技术方案,下面将对实施例中所需要使用的附图作简单地介绍。

[0071] 图1为本发明提供的流计算系统架构示意图;

[0072] 图2为现有技术的中流计算系统数据备份方法示意图;

[0073] 图3为本发明实施例提供的流计算系统逻辑划分示意图;

[0074] 图4为本发明实施例提供的业务处理逻辑示意图;

[0075] 图5为本发明实施例提供的数据备份方法的流程图;

[0076] 图6为本发明实施例提供的数据备份方法的原理示意图;

[0077] 图7为本发明实施例提供的源PE的工作流程图;

[0078] 图8为本发明实施例提供的目的PE的工作流程图;

[0079] 图9为本发明实施例提供的异步备份方法的示意图;

[0080] 图10为本发明实施例提供的一种流计算系统的示意图;

[0081] 图11为本发明实施例提供的另一种流计算系统的示意图;

[0082] 图12为本发明实施例提供的另一种流计算系统的示意图。

具体实施方式

[0083] 为使本发明的目的、技术方案和优点更加清楚，下面将结合本发明实施例中的附图，对本发明的技术方案进行清楚、完整地描述。显然，下述的各个实施例都只是本发明一部分的实施例。基于本发明下述的各个实施例，本领域普通技术人员即使没有作出创造性劳动，也可以通过等效变换部分甚至全部的技术特征，而获得能够解决本发明技术问题，实现本发明技术效果的其它实施例，而这些变换而来的各个实施例显然并不脱离本发明所公开的范围。

[0084] 本发明实施例提供的技术方案可典型地应用于流计算系统中，图3描述了流计算系统的基本结构，包括一个主控节点(Master)和多个工作节点(worker)，主控节点主要负责调度和管理各个工作节点，而工作节点是承载实际的数据处理操作的逻辑实体，工作节点具体通过调用若干个执行单元(PE,Process Element)来对待处理的数据进行处理(如图3中的PE1、PE2)，PE是业务逻辑的物理执行单元，其具体可以为处理器核、进程、线程或其它具有数据处理能力的功能模块、逻辑器件等;同时，为了快速有序地处理数据，工作节点中还设置有多个数据缓冲队列(如图3中的Q1、Q2、Q3、Q4)。PE1为PE2的上游处理单元，即经过PE1处理的数据，会从PE1发送给PE2做进一步处理，PE1和PE2属于不同的工作节点(在本发明实施例中，也称PE1为源PE，PE1下游的处理单元PE2为目的PE)，数据通信层首先从上游接收到数据并缓存在队列Q3中，数据转发层的接收线程从底层通讯层读取数据，并将发往PE1的元组数据路由至PE1对应的处理队列Q1。PE1循环从Q1中读取数据并进行处理，同时将处理过程中产生的中间状态数据缓存在state1中。PE1在处理数据过程中也会发送处理结果数据至发送队列Q2，该数据会被标记为发往PE2。数据转发层的发送线程从Q2中读取数据，并调用通信层的发送接口路由并发送数据，发送的数据首先会被缓存在底层通讯的发送队列Q4之中。数据通讯层会循环发送Q4中的数据至目标PE所属的通讯层模块。

[0085] 流计算是基于流式数据处理模型进行的，在流计算系统中，业务处理逻辑通常需要转化为无回路有向图(Directed Acyclic Graph,DAG)，如图4所示，其中算子(Operator)是业务逻辑载体，是可被流计算系统调度执行的最小单元;stream代表各Operator间的数据传输，PE是承载实际的数据处理操作的物理载体，PE可以动态加载并执行对应的operator所承载的业务逻辑，对业务产生的数据流进行处理;其中，数据流中单个数据段，称为元组，元组可以是结构化或非结构化数据。通常，元组中的数据表示特定时间点某事物的状态，流计算系统中的PE以元组为单位对业务产生的数据流进行处理，也可以认为元组是流计算系统中的数据的最小粒度划分和表示。同理，流处理在DAG处理模型下，数据经过传输处理转发等等一系列流程，所以在流计算系统中存在大量的队列数据和数据处理过程中的状态数据，对于数据处理的可靠性实现，最直接有效的手段就是进行数据的备份与恢复，但对于流计算系统的实际特点，很难实现数据的一致性备份与恢复，而本发明技术方案就是根据这一技术问题提出的。需要说明的是，流计算系统只是本发明技术方案的一个典型应用场景，并不对本发明的应用场景构成限制，其它涉及分布式系统数据一致性备份与恢复的应用场景，本发明实施例的技术方案均适用。

[0086] 本发明实施例提供一种流计算系统中的数据备份方法，该方法可应用与图3所示的流计算系统中，如图5、图6所示所示，该备份方法主要过程描述如下：

[0087] S501：源PE接收元组并缓存在自身的元组发送队列中；

[0088] S502：当源PE接收到检查点命令后，生成检查点元组，并将生成的检查点元组加入元组发送队列；其中，元组的格式如图6所示，根据图6，元组中携带有用于指示该元组类型的元组类型标识以及用于表示该元组批次的批号，元组类型标识用于区分一个元组是普通元组还是检查点元组，本发明实施例中，元组类型标识为0表示普通元组，元组类型标识为1表示检查点元组；可以理解的是，还可以用其他标识来区分普通元组和检查点元组，本发明实施例不做特别限定。本发明实施例的普通元组是指承载业务数据的元组，检查点元组是指承载系统控制消息的元组，更具体地，检查点元组主要用于指示PE进行数据备份、数据清理或数据恢复，同时，检查点元组与普通元组格式相同，以便于将其嵌入到数据流中，保证不阻塞PE正常的数据处理，提高效率。同时，基于流计算的特点，一个PE通常同时会接收并处理多个上游PE发送的元组，为保证数据备份及恢复的一致性，本发明实施例在元组中增加了批号标识，具体而言，源PE发送队列中的两个检查点元组之间的数据元组定义为同批次数据，通过在元组中增加批号字段来标识元组的批次，属于同一批次的元组，批号相同，例如图6中的B6-B10，在检查点ckp01和ckp02之间，属于同一批次的元组，故具有相同的批号。另外，检查点元组作为各批次元组的边界，与其相邻的普通元组的批号相同，具体而言，在源PE的发送队列中，检查点元组的批号可以与其前相邻的元组的批号相同，也可以与其后相邻的元组的批号相同，本发明实施例不做特别限定。

[0089] S503：源PE将元组发送队列中的元组(包括普通元组和检查点元组)通过消息通道发送至目的PE的元组接收队列；

[0090] S504：目的PE接收源PE发送的元组(包括普通元组和检查点元组)，并顺序缓存在元组接收队列中；

[0091] S505：目的PE根据业务处理逻辑，依次读取元组接收队列中的元组(包括普通元组和检查点元组)，对读取到的元组进行处理，并缓存处理过程中的状态数据；其中，PE的状态数据用于表示PE的处理数据状态，其具体包含的内容是本领域技术人员熟知的，例如状态数据可包括：算子状态数据、业务处理逻辑、元组接收队列中的缓存数据、消息通道中的缓存数据、PE在处理自身接收队列中的一个或多个普通元组的过程中产生的数据(比如当前处理的普通元组的处理结果及中间过程数据)中的一种或多种数据。

[0092] S506：如果目的PE读取到的元组为检查点元组，且为用于指示所述目的PE进行状态数据备份的第一检查点元组，则判断该第一检查点元组的批号与当前批号是否相同，以及与所述第一检查点元组具有相同批号的所有元组是否都已到齐，如果第一检查点元组的批号与当前批号相同，且与第一检查点元组具有相同批号的所有元组都已到齐，则目的PE将自身当前的状态数据备份；具体地，目的PE根据读取到的元组的元组类型标识，可以判断出该元组是普通元组还是检查点元组，若读取到的元组为检查点元组，则进一步判断该元组的批号是否满足备份要求(即批号与当前批号相同，且同批次的所有元组都已到齐)，若满足，就进行状态数据备份操作。在一个实施例中，目的PE可将自身的状态数据备份至分布式存储中；其中，该分布式存储器是流计算系统中的一个非易失性存储装置，用于流计算系统中各个PE进行状态数据的备份。需要说明的是，分布式存储器不应理解为对本发明实施方式的特别限定，其它类型的具备可靠性的存储装置，均能用于实施本发明方案。另外还需要说明的是，本发明实施例中的“当前批号”用于指示目的PE目前处理到什么批次的元组，

具体而言，当前批号为目的PE当前处理的普通元组的批号；需要说明的是，这里的“当前处理的元组”，应当理解为目的PE执行S506之前，最近一次读取并处理的元组，该元组通常为普通元组；“与第一检查点元组具有相同批号的所有元组都已到齐”，具体是指与第一检查点元组同批次(批号相同)的所有普通元组均已被目的PE接收并处理完毕。

[0093] 本发明提供的流计算系统中的数据备份方法，通过检查点元组来触发PE进行数据备份，PE从接收队列中读取到检查点元组之后，再执行备份操作，各个PE的备份操作不需要同步，使得数据备份不受PE之间数据传递时延的影响；同时通过设置元组的批次，以及批号的比较，使得同一批号的所有元组到齐之后再进行状态备份，避免了流计算系统中数据处理的并发性和无序性对数据备份一致性的影响，从而保证了数据备份的一致性。同时，将检查点命令以检查点元组的形式嵌入到待处理的普通元组中，也可以保证数据备份操作不阻塞PE正常的数据处理，提高数据备份的效率。

[0094] 基于上述实施例，下面分别进一步描述源PE和目的PE详细的处理流程，如图7所示，源PE的具体处理流程如下：

[0095] 步骤701：判断是否接收到检查点命令，如果是，执行步骤702；如果否，执行步骤705；

[0096] 步骤702：判断检查点命令的类型，如果是用于数据备份的检查点命令，则执行步骤703；如果是用于数据清理的检查点命令，则执行步骤706；

[0097] 步骤703：生成第一检查点元组，并将生成的第一检查点元组加入源PE的发送队列；其中，第一检查点元组用于指示下游的目的PE进行状态数据备份；

[0098] 步骤704：将第一检查点元组发送给目的PE，以使目的PE调用自身的备份与恢复模块对自身的状态数据进行备份；

[0099] 步骤705：源PE调用算子，以使算子根据业务处理逻辑依次对源PE接收队列中接收的元组进行处理；

[0100] 步骤706：清理发送队列；

[0101] 步骤707：发送清理检查点命令给下游的目的PE，以使下游的目的PE调用自身的备份与恢复模块对自身的状态数据进行清理。

[0102] 相应地，如图8所示，目的PE的具体处理流程如下：

[0103] 步骤801：目的PE读取自身的元组接收队列(recRB)中的元组数据；

[0104] 步骤802：判断读取到的元组的类型，若该元组为检查点元组，则执行步骤803；如果该元组为普通元组，则执行步骤807；

[0105] 需要说明的是，由于元组中携带有用于指示元组类型的元组类型标识，通过该元组类型标识即可区分出一个元组是普通元组还是检查点元组，本发明实施例中，元组类型标识为0表示普通元组，元组类型标识为1表示检查点元组；可以理解的是，还可以用其他标识来区分普通元组和检查点元组，本发明实施例不做特别限定。

[0106] 步骤803：判断该检查点元组的类型，如果为备份类型的检查点元组，则执行步骤804；如果为恢复类型的检查点元组，则执行步骤805；如果为清理类型的检查点元组，则执行步骤806；其中，备份类型的检查点元组是指用于指示所述目的PE进行状态数据备份的检查点元组，恢复类型的检查点元组是指用于指示所述目的PE进行状态数据恢复的检查点元组，清理类型的检查点元组是指用于指示所述目的PE进行状态数据清理的检查点元组；需

要说明的是，在本发明的实施例中，可以通过在检查点元组中设定检查点类型标识来区分不同类型的检查点元组，例如，检查点类型为1表示是备份类型的检查点元组，检查点类型为2表示是恢复类型的检查点元组，检查点类型为3表示是清理类型的检查点元组，本发明实施例不做特别限定。

[0107] 步骤804：将该元组的批号与当前批号进行比较，如果该元组的批号大于当前批号，执行步骤808；如果该元组的批号小于当前批号，返回步骤801；如果该元组的批号等于当前批号，说明该元组的批号符合备份要求，则设置检查点状态数据中的备份标记，执行步骤809；其中，“当前批号”用于指示目的PE目前处理到什么批次的数据，具体而言，当前批号为目的PE当前处理的元组的批号；需要说明的是，这里的“当前处理的元组”，应当理解为目的PE执行上述步骤之前，最近一次读取并处理的元组，该元组通常为普通元组。在一个较佳的实施例中，目的PE可以维护检查点状态数据，检查点状态数据的格式如图8所示，该检查点状态数据包括：当前批号、备份标记以及元组到齐标记，备份标记用于指示元组的批号是否满足备份要求；元组到齐标记用于指示同一批次(批号相同)的所有元组是否均已被目的PE接收并处理完毕；可以理解的是，在目的PE处理数据的过程中，检查点状态数据是动态更新的，例如，在一个实施例中，如判断与该元组同批次的元组都到齐了，则将元组到齐标记置为1，若未到齐则将元组到齐标记设置为0；若该元组的批号等于当前批号，且与该元组同批次的数据都已被目的PE处理完毕，说明该元组的批号符合备份要求，则将备份标记设置为1。可以理解的是，元组到齐标记和备份标记的设置方法还可以采用其它方式，只要能区分不同的状态即可，本发明不做特别限定。

[0108] 步骤805：从加载自身备份的状态数据，并基于所述状态数据进行状态恢复和数据回放，返回步骤801；其中，PE基于自身的状态数据进行状态恢复和数据回放属于本领域常规技术手段，此处不再赘述。

[0109] 步骤806：清理检查点状态数据，返回步骤801；需要说明的是，在本发明实施例中，状态数据备份是由检查点元组触发的，每个检查点元组都对应有的检查点状态信息，在一个较佳的实施例中，如果在新的检查点元组触发下，PE备份状态数据成功，则之前的检查点元组对应的检查点状态信息，以及PE在之前的检查点元组触发下备份的状态数据均可以可删除，这样可以及时释放存储空间。

[0110] 步骤807：将该元组的批号与当前批号进行比较，如果该元组的批号等于当前批号，则调用算子对该元组进行处理，并将该元组的批号记录为当前批号，返回执行步骤801；如果该元组的批号小于当前批号，则丢弃该元组，返回步骤801；如果该元组的批号如果该元组的批号大于当前批号，执行步骤808；

[0111] 步骤808：则将该元组加入缓冲队列，以便与当前批号具有相同批号的所有元组均被处理完毕之后，再处理该元组；

[0112] 步骤809：更新源PE元组到齐标记；

[0113] 步骤810：若检查点状态信息中的备份标记为1且当前批号所有元组已到齐，则将当前的状态数据进行备份；其中，所述目的PE的状态数据包括所述目的PE在处理所述普通元组过程中产生的数据；例如，在一个优选的实施例中，目的PE可以将状态数据备份至分布式存储中；相应地，在步骤805中，目的PE具体是从所述分布式存储器中加载自身最近一次备份的状态数据，并基于所述最近一次备份的状态数据进行状态恢复和数据回放；可以理

解的是，目的PE可能会在不同的时间点，对自身的状态数据做多次备份，在目的PE读取到恢复类型的检查点元组之后，优选距离读取该检查点元组时刻最近一次备份的状态数据来进行状态恢复和数据回放。

[0114] 步骤811：清理检查点状态信息；具体地，清理检查点状态信息包括：将检查点状态信息中的当前批号加1，将备份标记置为0，将源PE元组到齐标记清零；

[0115] 步骤812：向下游PE派发备份类型的检查点元组。

[0116] 需要说明的是，在另一个较优的实施例中，为了进一步提高流计算系统的运行效率，在步骤810中，目的PE可以采用异步备份的方式来备份状态数据；具体地，如图9所示，Operator提供接口1，目的PE通过调用接口1提取状态数据并存放到本地内存，同时目的PE通过接口2调用备份与恢复模块将本地内存中的状态数据备份至分布式存储；具体地，备份与恢复模块启动备份恢复线程，以使备份恢复线程通过分布式存储接口将本地内存中的状态数据备份至分布式存储。

[0117] 通过上面的详细描述可以看出，本发明实施例提供的流计算系统中的数据备份方法，通过检查点元组来触发PE进行数据备份，PE从接收队列中读取到检查点元组之后，再执行备份操作，各个PE的备份操作不需要同步，使得数据备份不受PE之间数据传递时延的影响；同时通过设置元组的批次，以及批号的比较，使得同一批号的所有元组到齐之后再进行状态备份，避免了流计算系统中数据处理的并发性和无序性对数据备份一致性的影响，从而保证了数据备份的一致性。进一步地，PE通过接口调用，采用异步备份的方式来备份状态数据，可以保证数据备份操作不阻塞PE正常的数据处理，提高流计算系统的运行效率。

[0118] 基于上述方法实施例，本发明实施例还提供一种流计算系统，用于实施上述方法，如图10所示，该流计算系统，包括：多个工作节点(101-103)，工作节点(101-103)通过调用多个执行单元(PE)来对元组进行处理，所述执行单元包括：源PE(如图10中的PE1)和目的PE(如图10中的PE2)；其中，所述源PE，用于将自身的发送队列中缓存的元组发送到所述目的PE的接收队列中；所述目的PE，用于依次读取自身的接收队列中的元组并进行处理；其中，所述源PE的发送队列中缓存的元组包括普通元组和检查点元组，且每个元组携带有用于表示该元组批次的批号；其中，检查点元组用于指示PE进行数据备份或数据恢复；所述源PE的发送队列中不同的检查点元组具有不同的批号，处于相邻的两个检查点元组之间的普通元组具有相同的批号，且每个检查点元组的批号与其相邻的一个普通元组的批号相同；其中：

[0119] 所述源PE，还用于接收用于状态数据备份的第一检查点命令，根据所述第一检查点命令生成第一检查点元组，将生成的检查点元组加入所述源PE的发送队列，将所述发送队列中缓存的所述第一检查点元组发送至所述目的PE的接收队列；

[0120] 所述目的PE，还用于从自身的接收队列中读取所述第一检查点元组，判断所述第一检查点元组的批号与当前批号是否相同，以及与所述第一检查点元组具有相同批号的所有普通元组是否都已处理完毕；所述当前批号为所述目的PE当前处理的普通元组的批号；若所述第一检查点元组的批号与所述当前批号相同，且与所述第一检查点元组具有相同批号的所有普通元组都已处理完毕，则将自身的状态数据进行备份。本发明实施例提供的流计算系统，通过检查点元组来触发PE进行数据备份，各个PE从接收队列中读取到检查点元组之后，再执行备份操作，各个PE的备份操作不需要同步，使得数据备份不受PE之间数据传递时延的影响；同时通过设置元组的批次，以及批号的比较，使得同一批号的所有元组到齐

之后再进行状态备份，避免了流计算系统中数据处理的并发性和无序性对数据备份一致性的影响，从而保证了数据备份的一致性。

[0121] 进一步地，在另一个实施例中，目的PE在进行状态数据备份之后，如果流计算系统发生故障，或者外部触发的情形下，可以基于最近一次备份的状态数据进行数据恢复；具体地，所述源PE接收用于数据恢复的第二检查点命令，根据所述第二检查点命令生成用于指示所述目的PE进行状态数据恢复的第二检查点元组，将生成的第二检查点元组加入所述源PE的发送队列，将所述发送队列中缓存的所述第二检查点元组发送至所述目的PE的接收队列；

[0122] 所述目的PE从自身的接收队列中读取所述第二检查点元组，根据所述第二检查点元组加载自身备份的状态数据，并基于所述状态数据进行状态恢复和数据回放。

[0123] 进一步地，在另一个实施例中，目的PE在还可以定期对自身备份的状态数据做清理，以释放存储空间；具体地，所述源PE接收用于状态数据清理的第三检查点命令，根据所述第三检查点命令生成第三检查点元组，将生成的第三检查点元组加入所述源PE的发送队列，将所述发送队列中缓存的所述第三检查点元组通过消息通道发送至所述目的PE的接收队列；

[0124] 所述目的PE从自身的接收队列中读取所述第三检查点元组，并根据所述第三检查点元组清理自身备份的状态数据。

[0125] 进一步的，在一个优选的实施例中，上述第一检查点命令、第二检查点命令和第三检查点命令中是由检查点PE(如图10中的PE3)发送的；检查点PE用于负责管理整个流计算系统检查点相关信息的发送和管理。

[0126] 在一个优选的实施例中，该流计算系统还包括：分布式存储器中104；

[0127] 在将自身的状态数据进行备份的方面，所述目的PE具体用于，通过调用第一接口将自身当前的状态数据缓存到本地内存；通过第二接口调用备份与恢复模块，以使所述备份与恢复模块启动备份恢复线程，将所述本地内存中的状态数据备份至分布式存储器104。相应地，在加载自身备份的状态数据的方面，所述目的PE具体用于，从分布式存储器中104加载自身备份的状态数据。

[0128] 可以看到，目的PE通过接口调用，采用异步备份的方式来备份状态数据，可以保证数据备份操作不阻塞PE正常的数据处理，提高流计算系统的运行效率。

[0129] 基于上述方法及系统实施例，本发明实施例还提供另一种流计算系统，如图11所示，该流计算系统包括：分布式存储器、主控节点(Master)和多个工作节点(worker)；其中，工作节点通过调用多个执行单元PE来对元组进行处理，所述执行单元包括：源PE和目的PE；主控节点中保存有检查点信息，所述检查点信息包括：检查点ID，时间戳、开始时间、完成时间，完成标记等，；主控节点主要用于检查点状态信息的管理，以及当系统出现异常时，根据检查点信息进行系统恢复决策；具体地，该主控节点，用于向所述检查点PE下发用于数据备份的备份命令、用于数据恢复的恢复命令或者用于状态数据清理的数据清理命令；所述检查点PE用于，在接收到所述主控节点发送的备份命令后，向所述源PE发送所述第一检查点命令；或者，在接收到所述主控节点发送的恢复命令后，向所述源PE发送所述第二检查点命令；或者，在接收到所述主控节点发送的数据清理命令后，向所述源PE发送所述第三检查点命令。

[0130] 所述源PE，用于将自身的发送队列中缓存的元组发送到所述目的PE的接收队列中；所述目的PE，用于读取自身的接收队列中的元组并进行处理；所述源PE的发送队列中缓存的元组包括普通元组和检查点元组，且每个元组携带有用于表示该元组批次的批号；其中，检查点元组用于指示PE进行数据备份或数据恢复；所述源PE的发送队列中不同的检查点元组具有不同的批号，相邻的两个检查点元组之间的普通元组具有相同的批号，且每个检查点元组的批号与其相邻的一个普通元组的批号相同；其中：

[0131] 所述检查点PE用于，在接收到所述主控节点发送的备份命令后，向所述源PE发送所述第一检查点命令；或者，在接收到所述主控节点发送的恢复命令后，向所述源PE发送所述第二检查点命令；或者，在接收到所述主控节点发送的数据清理命令后，向所述源PE发送所述第三检查点命令。

[0132] 所述源PE，还用于接收所述第一检查点命令，根据所述第一检查点命令生成第一检查点元组，将生成的检查点元组加入所述源PE的发送队列，将所述发送队列中缓存的所述第一检查点元组通过消息通道发送至所述目的PE的接收队列；

[0133] 所述目的PE，还用于从自身的接收队列中读取所述第一检查点元组，判断所述第一检查点元组的批号与当前批号是否相同，以及与所述第一检查点元组具有相同批号的所有元组是否都已处理完毕；所述当前批号为所述目的PE当前处理的普通元组的批号；若所述第一检查点元组的批号与所述当前批号相同，且与所述第一检查点元组具有相同批号的所有普通元组都已处理完毕，则将自身的状态数据备份至所述分布式存储器中。

[0134] 进一步地，在另一个实施例中，所述源PE还用于，接收所述第二检查点命令，根据所述第二检查点命令生成第二检查点元组，将生成的第二检查点元组加入所述源PE的发送队列，将所述发送队列中缓存的所述第二检查点元组通过消息通道发送至所述目的PE的接收队列；

[0135] 所述目的PE还用于，从自身的接收队列中读取所述第二检查点元组，并根据所述第二检查点元组从所述分布式存储器中加载自身的状态数据，并基于所述状态数据进行状态恢复和数据回放。

[0136] 进一步地，在另一个实施例中，所述源PE还用于，接收所述第三检查点命令，根据所述第三检查点命令生成第三检查点元组，将生成的第三检查点元组加入所述源PE的发送队列，将所述发送队列中缓存的所述第三检查点元组通过消息通道发送至所述目的PE的接收队列；

[0137] 所述目的PE还用于，从自身的接收队列中读取所述第三检查点元组，并根据所述第二检查点元组从所述分布式存储器中清理自身备份的状态数据。

[0138] 优选地，如图11所示，PE在进行状态数据备份时，可以先通过调用第一接口将当前的状态数据缓存到本地内存；然后通过第二接口调用备份与恢复模块，以使所述备份与恢复模块启动备份恢复线程，将所述本地内存中的状态数据备份至所述分布式存储器。

[0139] 本发明实施例提供的流计算系统中，通过检查点元组来触发PE进行数据备份，PE从接收队列中读取到检查点元组之后，再执行备份操作，各个PE的备份操作不需要同步，使得数据备份不受PE之间数据传递时延的影响；同时通过设置元组的批次，以及批号的比较，使得同一批号的所有元组到齐之后再进行状态备份，避免了流计算系统中数据处理的并发性和无序性对数据备份一致性的影响，从而保证了数据备份的一致性。进一步地，通过采用

异步备份的方式，可以避免对PE造成阻塞，提高了流计算系统的运行效率。

[0140] 本发明实施例还提供另一种流计算系统，用于实现本发明实施例提供的数据备份方法，如图12所示，该流计算系统包括：源执行单元(PE)102、目的执行单元103；其中，源执行单元102和目的执行单元103位于不同的工作节点上，且源执行单元102为目的执行单元103的上游执行单元；源执行单元102用于将自身的发送队列中缓存的元组通过消息通道发送到目的执行单元103的接收队列；其中源执行单元102的发送队列中缓存的元组具体包括普通元组和检查点元组，且每个元组携带有用于表示该元组批次的批号；其中，检查点元组用于指示执行单元进行数据备份或数据恢复；所述源执行单元的发送队列中不同的检查点元组具有不同的批号，处于相邻的两个检查点元组之间的普通元组具有相同的批号，且每个检查点元组的批号与其相邻的一个普通元组的批号相同；其中：

[0141] 源执行单元102包括：

[0142] 检查点模块1021，用于接收用于状态数据备份的第一检查点命令；根据所述第一检查点命令生成第一检查点元组；

[0143] 发送模块1022，用于将所述检查点模块生成的检查点元组加入源执行单元102的发送队列；并将所述发送队列中缓存的元组通过消息通道发送至目的执行单元103的接收队列。

[0144] 目的执行单元103包括：业务数据处理模块1031，用于从目的执行单元103的接收队列中读取元组(包括普通元组和检查点元组)并对读取到的元组进行处理；

[0145] 备份与恢复模块1032，用于当业务数据处理模块1031读取到的元组为用于指示目的执行单元103进行状态数据备份的第一检查点元组时，判断所述第一检查点元组的批号与当前批号是否相同，以及与所述第一检查点元组具有相同批号的所有普通元组是否都已处理完毕；所述当前批号为业务数据处理模块1031当前处理的普通元组的批号；若所述第一检查点元组的批号与所述当前批号相同，且与所述第一检查点元组具有相同批号的所有普通元组都已处理完毕，则将目的执行单元103当前的状态数据进行备份。

[0146] 进一步地，在另一个实施例中，备份与恢复模块1032，还用于当业务数据处理模块1031读取到的元组为用于指示所述目的执行单元进行状态数据恢复的第一检查点元组时，加载自身备份的状态数据，并基于所述备份的状态数据进行状态恢复和数据回放。

[0147] 进一步地，在另一个实施例中，目的执行单元103还维护有检查点状态信息，所述检查点状态信息包括：所述当前批号以及元组到齐标记；

[0148] 相应地，在判断所述第一检查点元组的批号与当前批号是否相同，以及与所述第一检查点元组具有相同批号的所有普通元组是否都已到齐的方面，备份与恢复模块1032具体用于：

[0149] 比较所述第一检查点元组的批号与所述检查点状态信息中包含的当前批号是否相等，以及根据所述检查点状态信息中的元组到齐标记确定与所述第一检查点元组具有相同批号的所有普通元组是否都已处理完毕。

[0150] 本发明实施例提供的流计算系统，源执行单元通过检查点元组来触发下游的目的执行单元进行数据备份，同时通过设置元组的批次，以及批号的比较，使得同一批号的所有元组到齐之后再进行状态备份，避免了流计算系统中数据处理的并发性和无序性对数据备份一致性的影响，从而保证了数据备份的一致性。

[0151] 进一步地，在一个优选的实施例中，所述流计算系统还包括：分布式存储器104；备份与恢复模块1032具体可采用异步备份的方式对目的执行单元103的状态数据进行备份，具体地，备份与恢复模块1032通过调用第一接口将目的执行单元103当前的状态数据缓存到本地内存，然后再启动备份线程，以使所述备份线程将所述本地内存中的状态数据备份至所述分布式存储器，相应地，备份与恢复模块1032可以从所述分布式存储器中加载自身最近一次备份的状态数据，并基于所述最近一次备份的状态数据进行状态恢复和数据回放。

[0152] 备份与恢复模块1032具体采用上述异步备份的方式，可以避免对PE造成阻塞，提高了流计算系统的运行效率。需要说明的是，本发明提供的流计算系统用于实施上述方法，其具体实现细节，可以参照上述方法实施例，此处不再赘述。本发明实施例中的执行单元(PE)可以以软件形态存在，例如进程、线程或软件功能模块，也可以以硬件的形态存在，比如处理器核，或具有数据处理能力的逻辑电路等，通过读取存储器中的可执行代码或业务处理逻辑，实现本发明实施例所描述的功能，本发明不做特别限定。

[0153] 在本申请所提供的几个实施例中，应该理解到，所揭露数据备份和流计算系统可以通过其它的方式实现。例如，以上所描述的装置实施例仅仅是示意性的。

[0154] 所述作为分离部件说明的单元可以是或者也可以不是物理上分开的，作为单元显示的部件可以是或者也可以不是物理单元，即可以位于一个地方，或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0155] 另外，在本发明各个实施例提供的网络设备中的各功能单元可以集成在一个处理单元中，也可以是各个单元单独物理存在，也可以两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现，也可以采用软件功能单元的形式实现。

[0156] 所述集成的单元如果以软件功能单元的形式实现并作为独立的产品销售或使用时，可以存储在一个计算机可读取存储介质中。基于这样的理解，本发明的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的全部或部分可以以软件产品的形式体现出来，该计算机软件产品存储在一个存储介质中，包括若干指令用以使得一台计算机设备(可以是个人计算机，服务器，或者网络设备等)执行本发明各个实施例所述方法的全部或部分步骤。而前述的存储介质包括：U盘、移动硬盘、只读存储器(ROM，Read-Only Memory)、随机存取存储器(RAM，Random Access Memory)、磁碟或者光盘等各种可以存储程序代码的介质。

[0157] 最后应说明的是：以上实施例仅用以说明本发明的技术方案，而非对其限制；尽管参照前述实施例对本发明进行了详细的说明，本领域的普通技术人员应当理解：其依然可以对前述各实施例所记载的技术方案进行修改，或者对其中部分技术特征进行等同替换；而这些修改或者替换，并不使相应技术方案的本质脱离本发明各实施例技术方案的精神和范围。

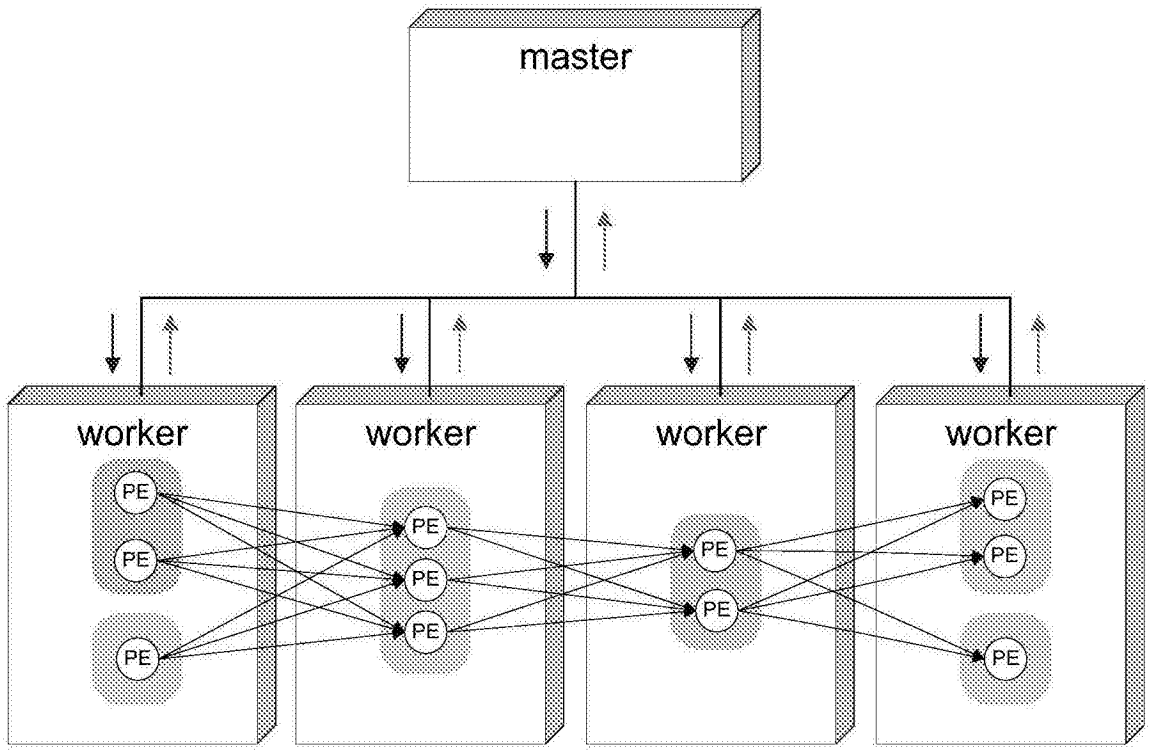

master
worker
worker
worker
worker
PE
PE
PE
PE
PE
PE
PE
PE
PE

图1

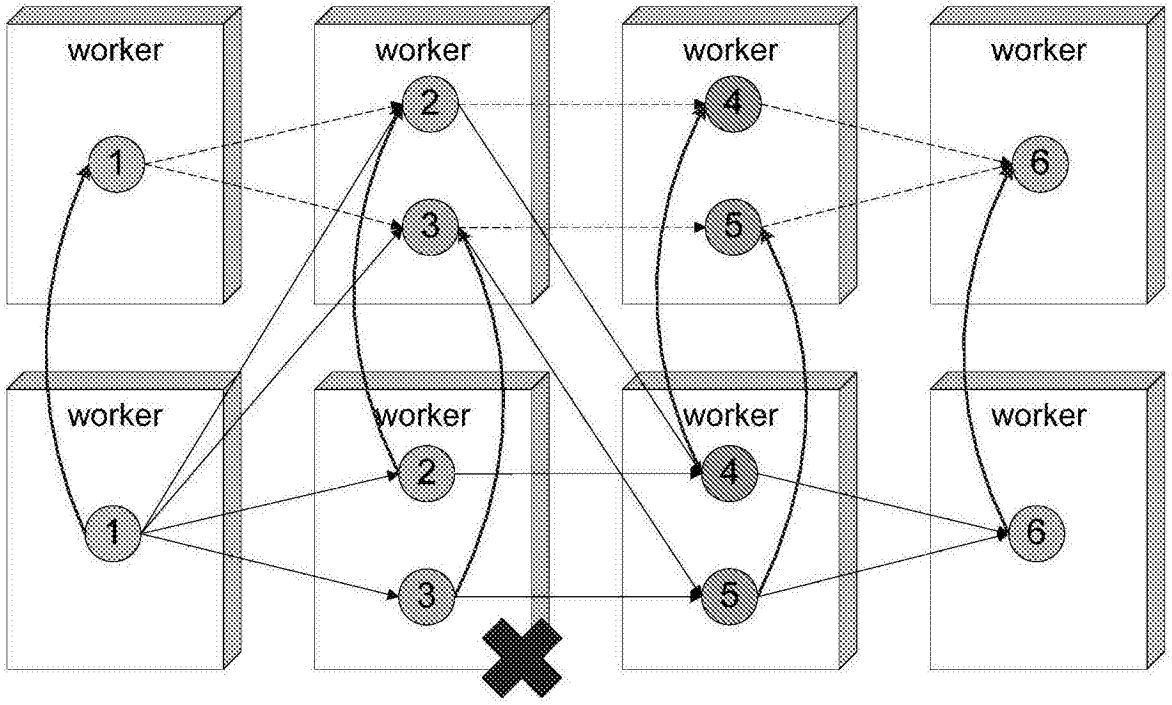

worker
worker
worker
worker
worker
worker
worker
worker
1
2
3
4
5
6
1
2
3
4
5
6

图2

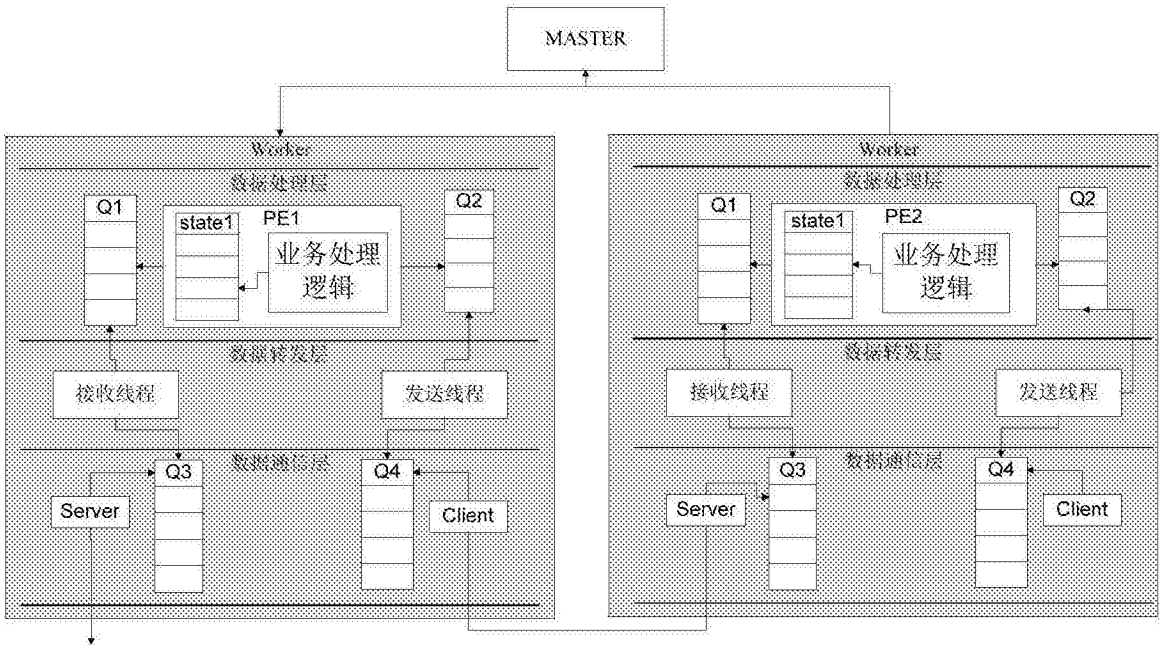

MASTER
Worker
数据处理层
Q1
state1
PE1
业务处理逻辑
Q2
数据转发层
接收线程
发送线程
数据通信层
Q3
Q4
Server
Client
Worker
数据处理层
Q1
state1
PE2
业务处理逻辑
Q2
数据转发层
接收线程
发送线程
数据通信层
Q3
Q4
Server
Client

图3

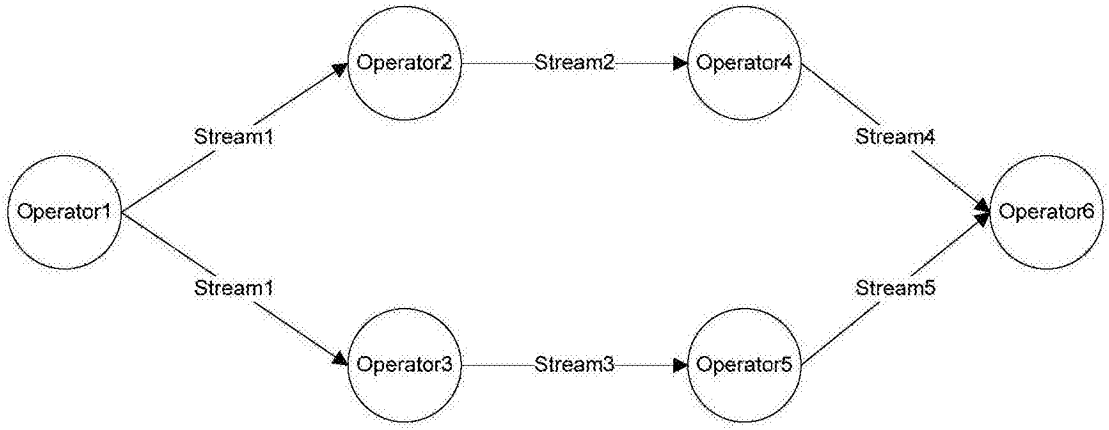

Operator1
Operator2
Operator3
Operator4
Operator5
Operator6
Stream1
Stream1
Stream2
Stream3
Stream4
Stream5

图4

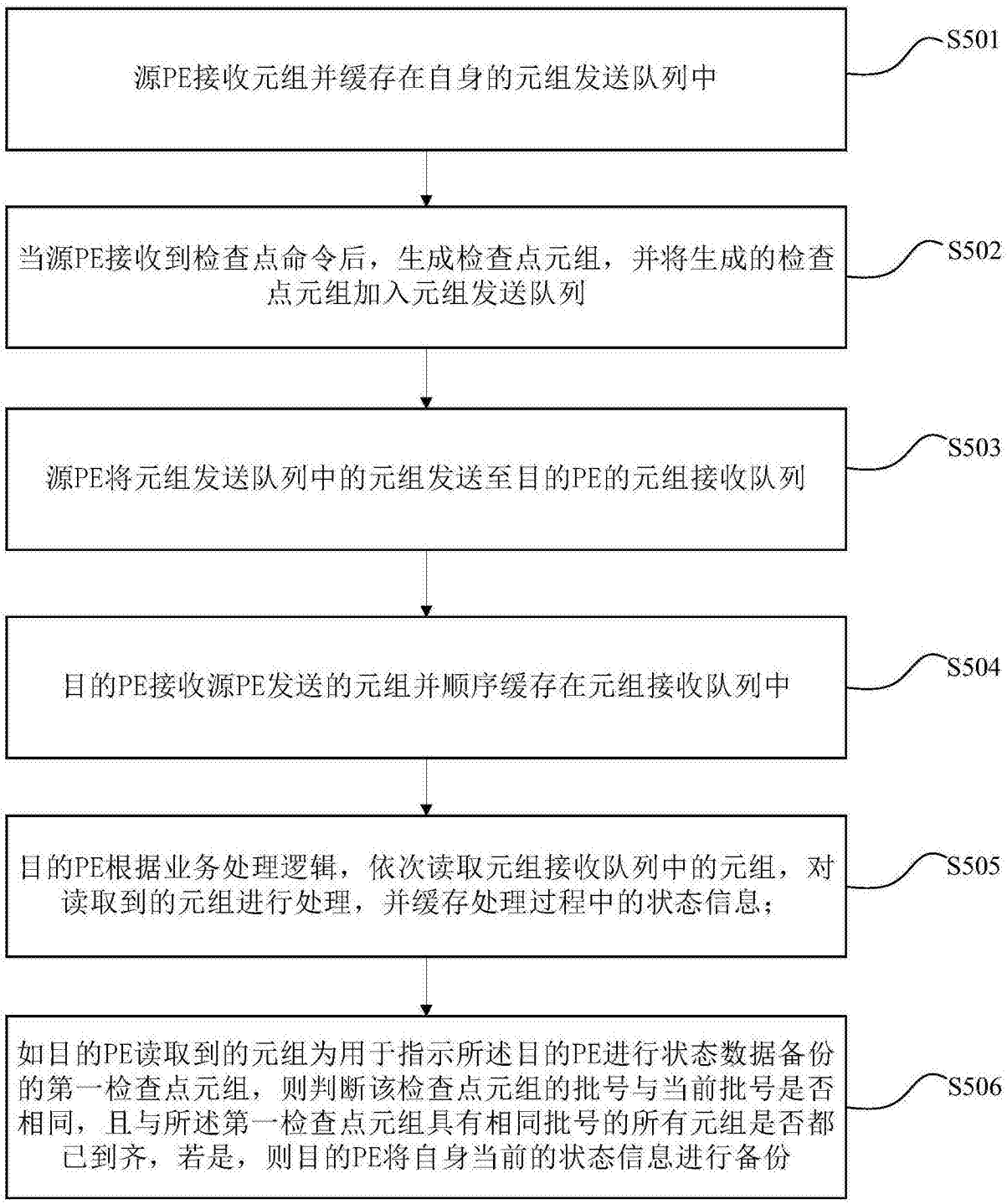

源PE接收元组并缓存在自身的元组发送队列中
S501
当源PE接收到检查点命令后，生成检查点元组，并将生成的检查点元组加入元组发送队列
S502
源PE将元组发送队列中的元组发送至目的PE的元组接收队列
S503
目的PE接收源PE发送的元组并顺序缓存在元组接收队列中
S504
目的PE根据业务处理逻辑，依次读取元组接收队列中的元组，对读取到的元组进行处理，并缓存处理过程中的状态信息；
S505
如目的PE读取到的元组为用于指示所述目的PE进行状态数据备份的第一检查点元组，则判断该检查点元组的批号与当前批号是否相同，且与所述第一检查点元组具有相同批号的所有元组是否都已到齐，若是，则目的PE将自身当前的状态信息进行备份
S506

图5

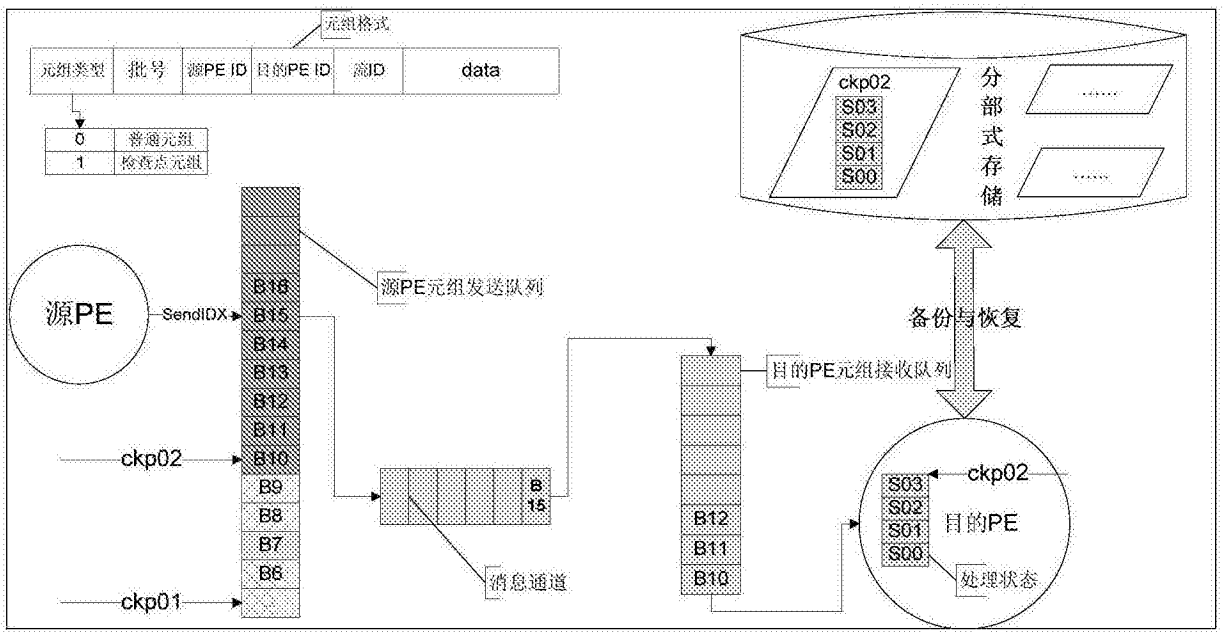

元组格式
元组类型
批号
源PE ID
目的PE ID
流ID
data
0
普通元组
1
检查点元组
ckp02
S03
S02
S01
S00
分部式存储
源PE
SendIDX
B16
B15
B14
B13
B12
B11
B10
B9
B8
B7
B6
ckp02
ckp01
源PE元组发送队列
B
15
消息通道
目的PE元组接收队列
B12
B11
B10
备份与恢复
S03
S02
S01
S00
ckp02
目的PE
处理状态

图6

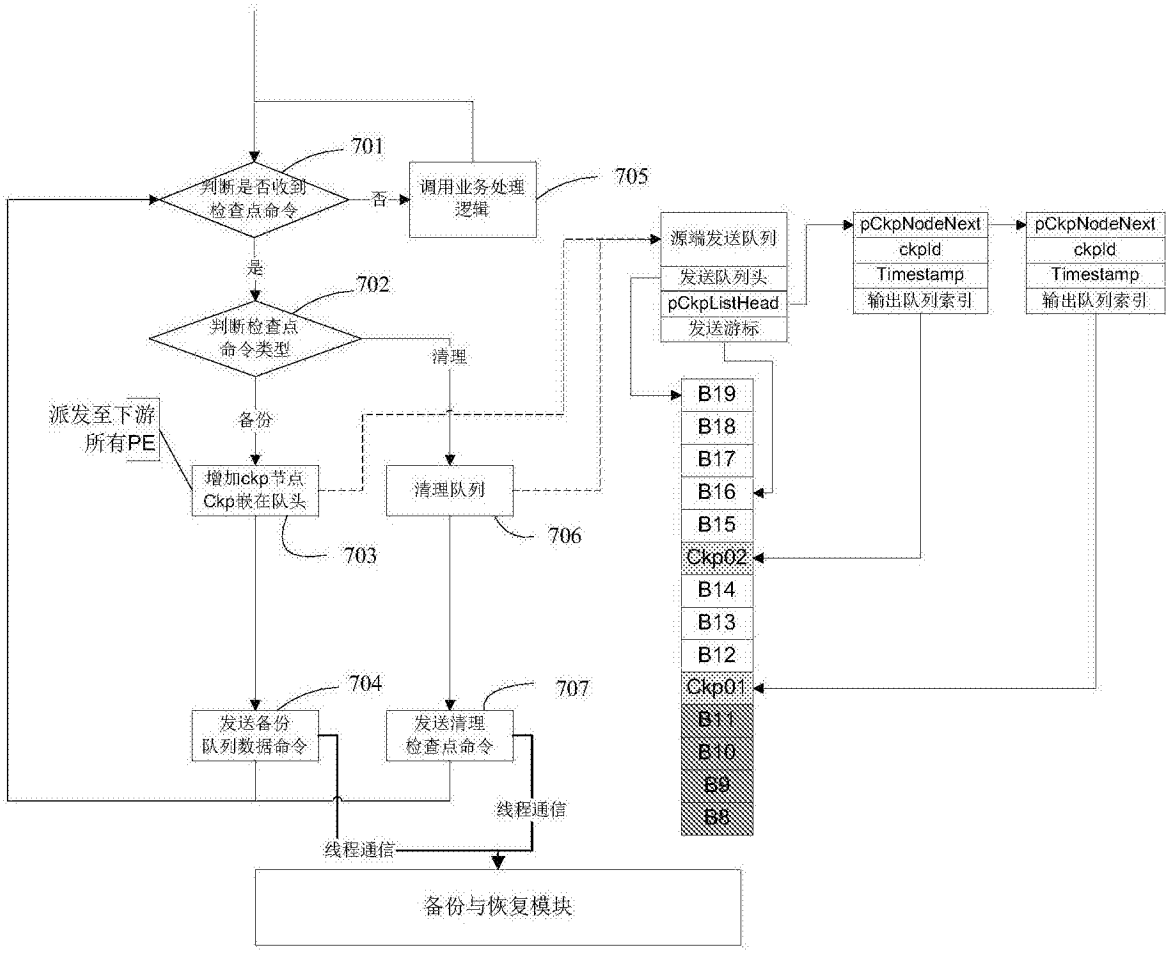

701
判断是否收到检查点命令
否
调用业务处理逻辑
705
是
702
判断检查点命令类型
清理
派发至下游所有PE
备份
增加ckp节点Ckp嵌在队头
清理队列
706
703
704
707
发送备份队列数据命令
发送清理检查点命令
线程通信
线程通信
备份与恢复模块
源端发送队列
发送队列头
pCkpListHead
发送游标
pCkpNodeNext
ckpId
Timestamp
输出队列索引
pCkpNodeNext
ckpId
Timestamp
输出队列索引
B19
B18
B17
B16
B15
Ckp02
B14
B13
B12
Ckp01
B11
B10
B9
B8

图7

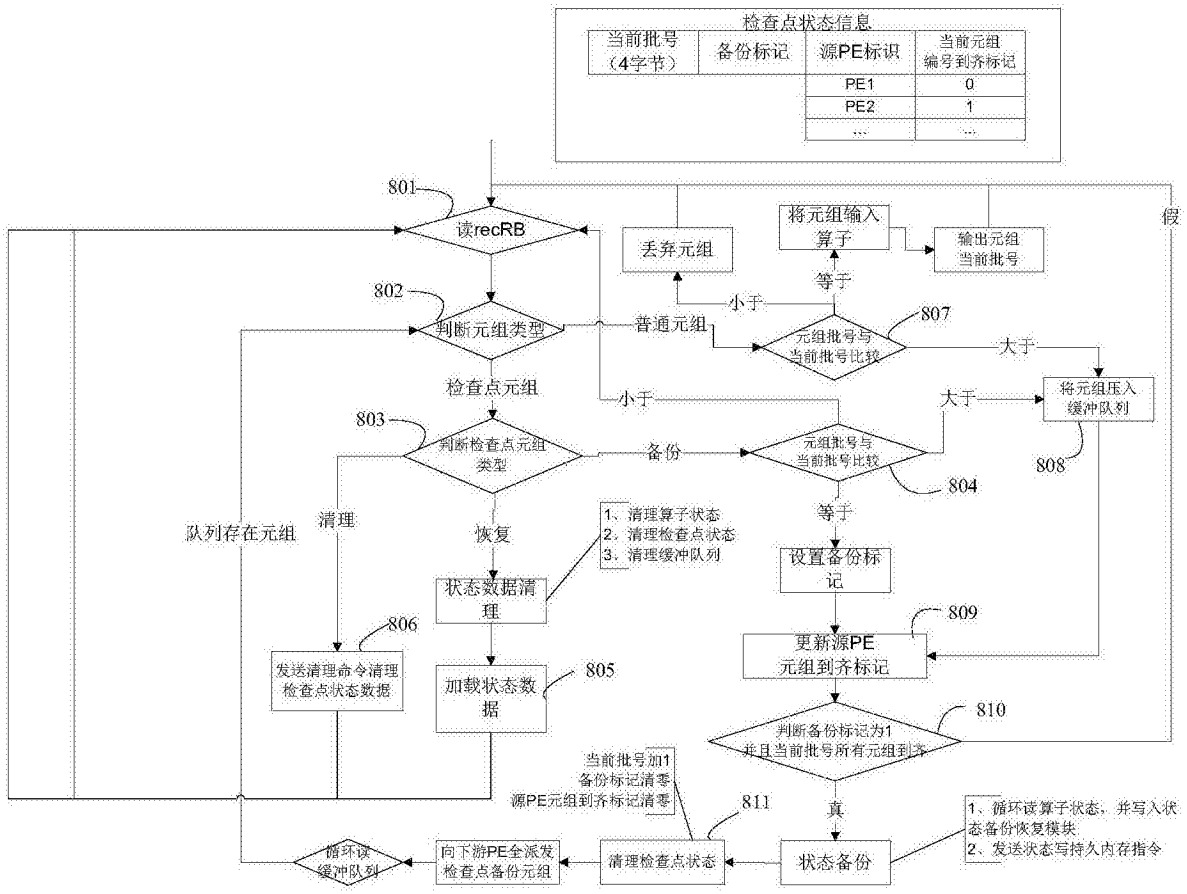

检查点状态信息
当前批号（4字节）
备份标记
源PE标识
当前元组编号到齐标记
PE1
0
PE2
1
...
...
801
读recRB
丢弃元组
将元组输入算子
输出元组当前批号
假
802
判断元组类型
普通元组
小于
等于
大于
元组批号与当前批号比较
807
检查点元组
小于
大于
将元组压入缓冲队列
808
803
判断检查点元组类型
备份
元组批号与当前批号比较
804
队列存在元组
清理
恢复
等于
1、清理算子状态
2、清理检查点状态
3、清理缓冲队列
设置备份标记
状态数据清理
809
806
更新源PE元组到齐标记
发送清理命令清理检查点状态数据
加载状态数据
805
810
判断备份标记为1并且当前批号所有元组到齐
当前批号加1
备份标记清零
源PE元组到齐标记清零
811
真
1、循环读算子状态，并写入状态备份恢复模块
2、发送状态写持久内存指令
循环读缓冲队列
向下游PE全派发检查点备份元组
清理检查点状态
状态备份

图8

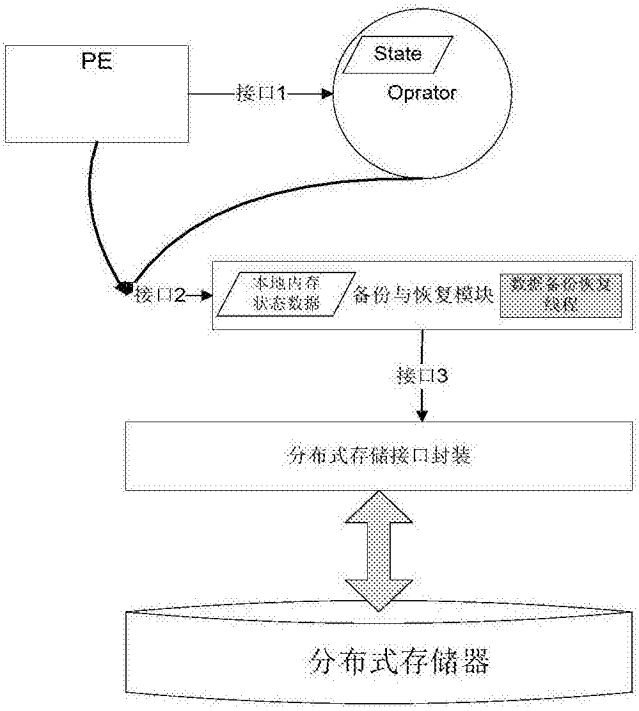

PE
接口1
State
Oprator
接口2
本地内存状态数据
备份与恢复模块
接口3
分布式存储接口封装
分布式存储器

图9

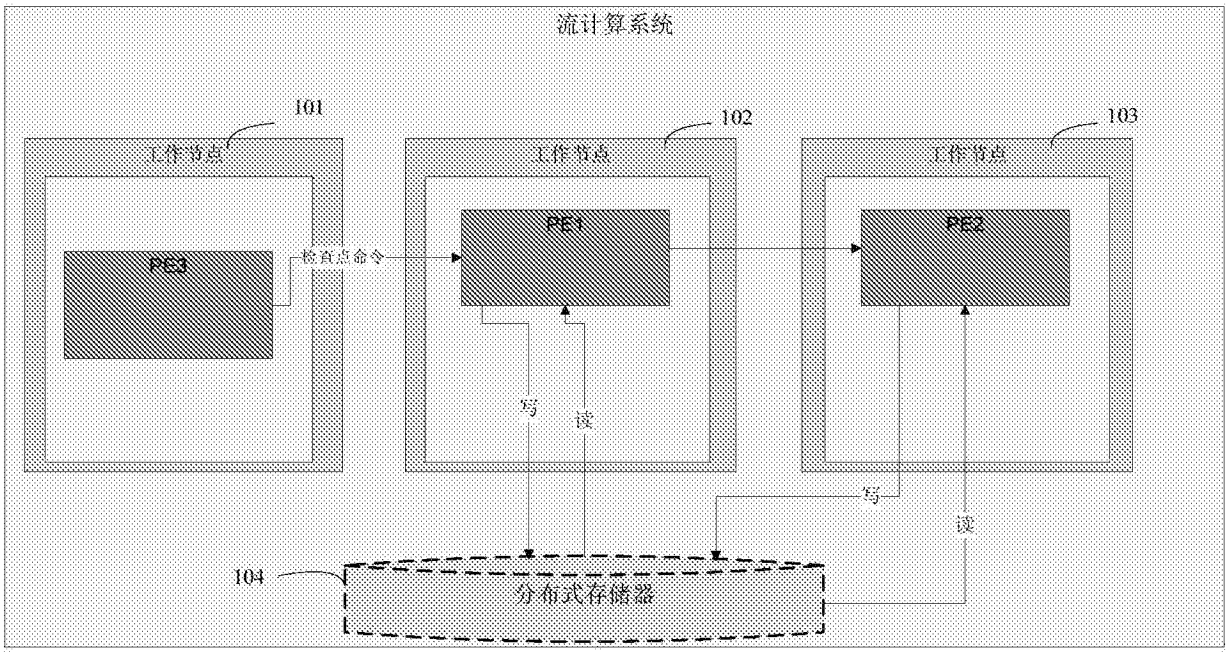

流计算系统
101
工作节点
102
工作节点
103
工作节点
PE3
检查点命令
PE1
PE2
写
读
写
读
104
分布式存储器

图10

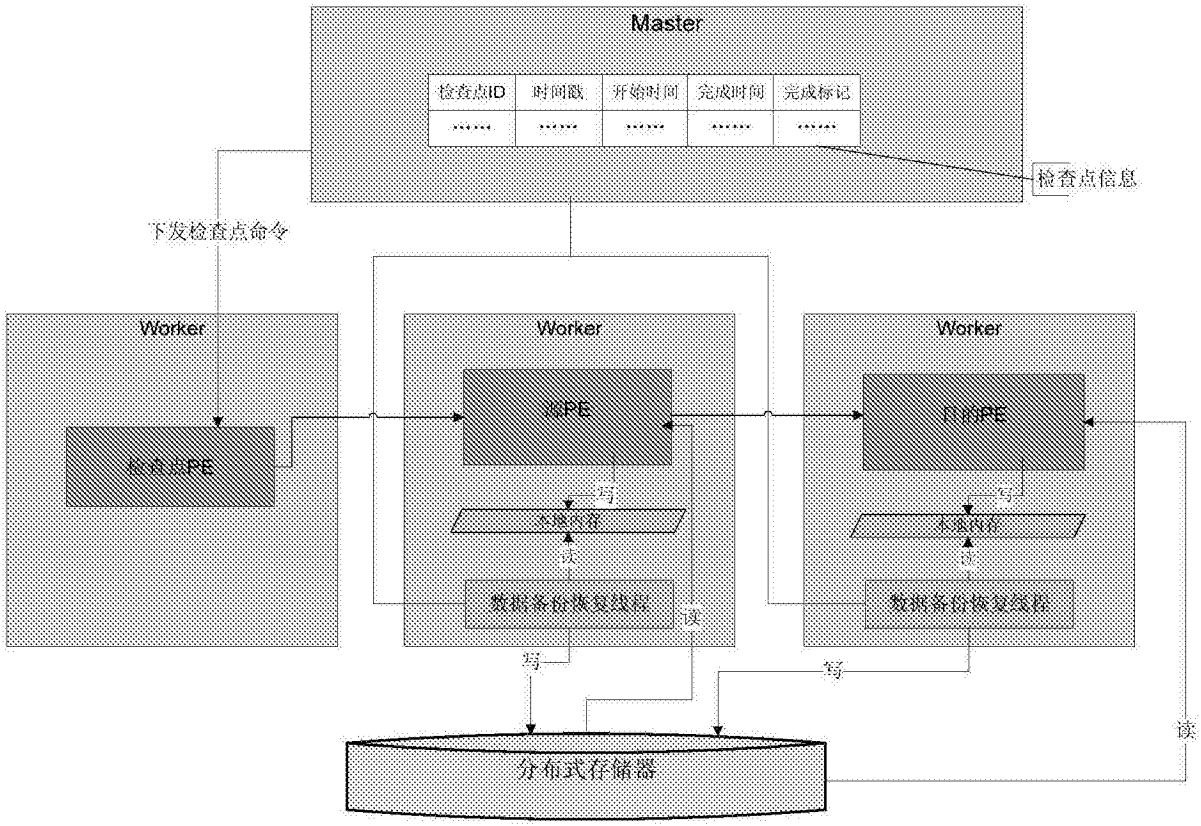

Master
检查点ID
时间戳
开始时间
完成时间
完成标记
检查点信息
下发检查点命令
Worker
Worker
Worker
写
读
写
读
写
读
分布式存储器

图11

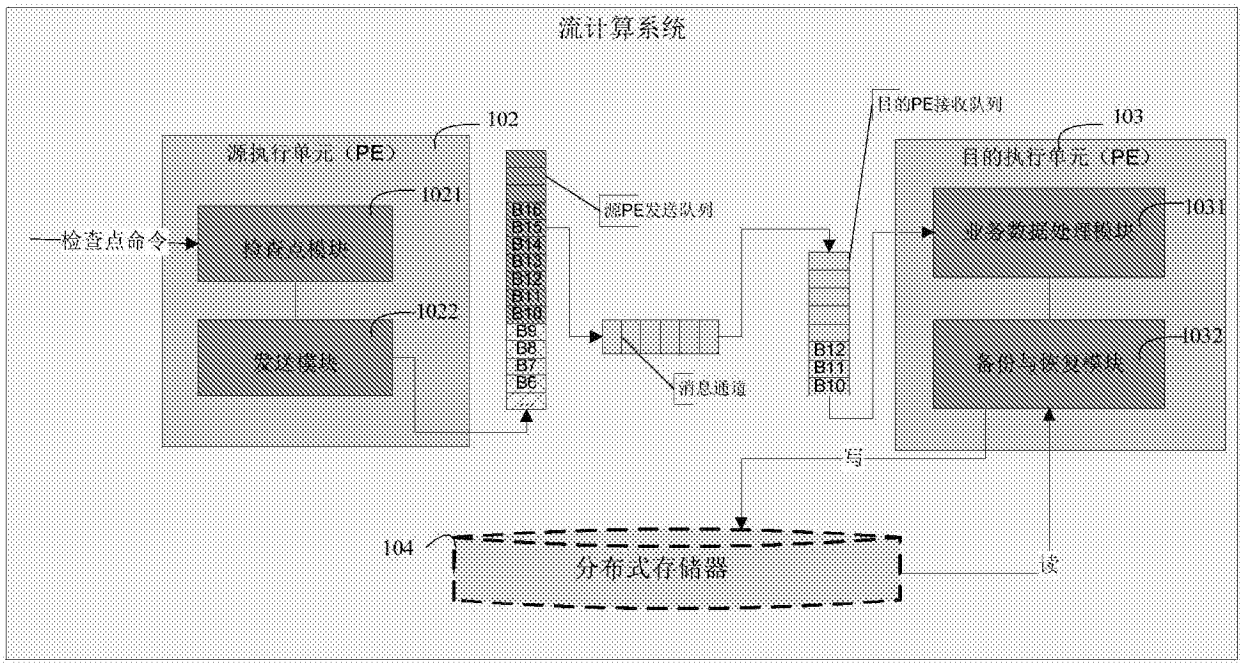

流计算系统
源执行单元（PE）
102
1021
检查点模块
1022
发送模块
检查点命令
B16
B15
B14
B13
B12
B11
B10
B9
B8
B7
B6
源PE发送队列
消息通道
目的PE接收队列
103
目的执行单元（PE）
1031
业务数据处理模块
1032
备份与恢复模块
B12
B11
B10
写
读
104
分布式存储器

图12