

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局

(43) 国际公布日
2017年6月8日 (08.06.2017)



(10) 国际公布号
WO 2017/092582 A1

- (51) 国际专利分类号:
G06F 17/30 (2006.01)
- (21) 国际申请号: PCT/CN2016/106580
- (22) 国际申请日: 2016年11月21日 (21.11.2016)
- (25) 申请语言: 中文
- (26) 公布语言: 中文
- (30) 优先权:
201510867351.7 2015年12月1日 (01.12.2015) CN
- (71) 申请人: 阿里巴巴集团控股有限公司 (ALIBABA GROUP HOLDING LIMITED) [—/CN]; 英属开曼群岛大开曼资本大厦一座四层 847 号邮箱, Grand Cayman (KY)。
- (72) 发明人: 冯粮城 (FENG, Liangcheng); 中国浙江省杭州市余杭区文一西路 969 号 3 号楼 5 楼阿里巴巴集团法务部, Zhejiang 311121 (CN)。 李俊良 (LI, Junliang); 中国浙江省杭州市余杭区文一西路 969 号 3 号楼 5 楼阿里巴巴集团法务部, Zhejiang 311121 (CN)。 强琦 (QIANG, Qi); 中国浙江省杭州市余杭区文一西路 969 号 3 号楼 5 楼阿里巴巴集团法务部, Zhejiang 311121 (CN)。
- (74) 代理人: 北京三友知识产权代理有限公司 (BEIJING SANYOU INTELLECTUAL PROPERTY AGENCY LTD.); 中国北京市金融街 35 号国际企业大厦 A 座 16 层, Beijing 100033 (CN)。
- (81) 指定国 (除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

[见续页]

(54) Title: DATA PROCESSING METHOD AND APPARATUS

(54) 发明名称: 一种数据处理方法和装置

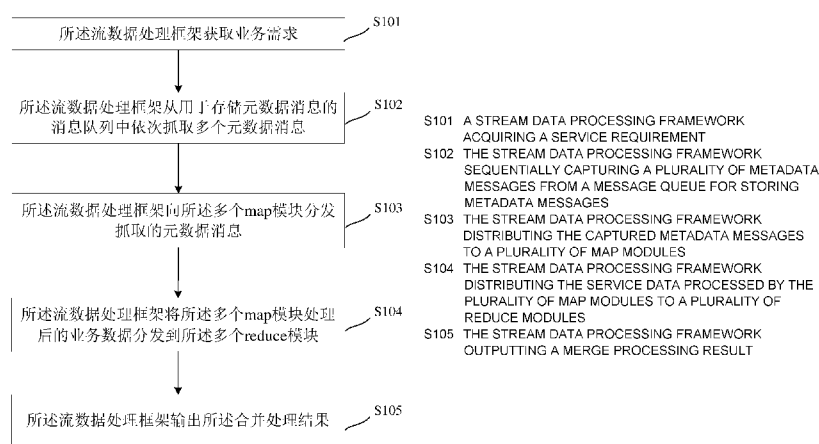


图 1

(57) Abstract: Disclosed are a data processing method and apparatus, which are applied to a stream data processing framework. The method comprises: a stream data processing framework acquiring a service requirement, sequentially capturing a plurality of metadata messages from a message queue for storing metadata messages, and distributing the captured metadata messages to a plurality of map modules, so that the plurality of map modules acquire corresponding service data according to a storage location in the received metadata messages; correspondingly processing the acquired service data; and distributing the service data processed by the plurality of map modules to a plurality of reduce modules, so that the plurality of reduce modules merge the processed service data, and outputting a merge processing result. By capturing a metadata message in a message queue in real time and then using real-time processing characteristics of a stream data processing framework, the stream data processing framework implementing a MapReduce function is used to process massive data generated in real time in a network, and pre-planning is no longer necessary before processing.

(57) 摘要:

[见续页]



WO 2017/092582 A1



(84) **指定国** (除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ,

CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

— 包括国际检索报告(条约第 21 条(3))。

本发明实施例公开了一种数据处理方法和装置, 应用于流数据处理框架, 流数据处理框架获取业务需求, 从用于存储元数据消息的消息队列中依次抓取多个元数据消息, 向多个 map 模块分发抓取的元数据消息, 使得多个 map 模块根据接收到的元数据消息中的存储位置, 获取对应的业务数据; 并相应的处理获取的业务数据; 将多个 map 模块处理后的业务数据分发到多个 reduce 模块, 使得多个 reduce 模块对所述处理后的业务数据进行合并处理, 输出合并处理结果。通过实时抓取消息队列中的元数据消息, 再利用流数据处理框架的实时处理特点, 使用实现了 MapReduce 功能的流数据处理框架处理网络中实时产生的海量数据, 处理前不再需要进行预先规划。

一种数据处理方法和装置

本申请要求 2015 年 12 月 01 日递交的申请号为 201510867351.7、发明名称为“一种数据处理方法和装置”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

5 技术领域

本发明涉及数据处理领域，特别是涉及一种数据处理方法和装置。

背景技术

随着互联网的发展，使用网络的用户增多，用户对网络的依赖变大。伴随着用户在网络上的操作，例如浏览新闻、购物等，网络上会因此实时的产生海量数据。这些海量数据对于互联网企业来说具有重要价值，例如可以通过分析数据确定出一个用户对哪些类型的网络资源更有兴趣，或者，一个网络资源被哪些用户近期所浏览过等。可见，如何能够快速有效的处理网络中实时产生的海量数据是急需解决的问题。

针对海量数据的传统处理方式是使用 MapReduce，MapReduce 是一种编程模型，用于对大规模数据集进行并行运算。在 MapReduce 中包括映射（英文：Map）函数和归约（英文：Reduce）函数。Map 函数主要用于根据需求对数据进行相应的处理，这里的需求可以是关键值（英文：key 或 value）或者键值对（英文：key/value pair）的形式。Map 函数处理后的数据通过中间传输过程（英文：Shuffle）传输到 Reduce 函数，由 Reduce 函数根据所述需求对处理后的数据进行合并，得到符合所述需求的数据处理结果。

然而，MapReduce 主要在 Hadoop（一种分布式系统基础架构）平台上运行，主要用于对海量数据进行离线的批量计算。也就是说，MapReduce 需要对待处理的数据集进行预先的规划后，例如确定哪些数据由哪个 Map 函数处理等之后，才能对这个数据集进行批量处理。处理前需要预先规划的特点导致了 MapReduce 主要用于处理离线数据，而难以对实时产生的数据进行快速的处理。可见，传统的 MapReduce 不能有效的解决目前需要快速处理网络中实时产生的海量数据的问题。

发明内容

为了解决上述技术问题，本发明提供了一种数据处理方法和装置，实现了对网络中实时产生的海量数据进行快速的处理。

30 本发明实施例公开了如下技术方案：

一种数据处理方法，应用于流数据处理框架，所述流数据处理框架包括多个 map 模块和多个 reduce 模块，所述方法包括：

所述流数据处理框架获取业务需求，所述业务需求包括至少一个关键值；

所述流数据处理框架从用于存储元数据消息的消息队列中依次抓取多个元数据消息，所述消息队列通过所述业务需求确定，所述元数据消息与业务消息一一对应，所述元数据消息包括对应的业务数据的存储位置，所述元数据消息依据所对应业务数据的生成时间先后，被顺序添加到所述消息队列中；

所述流数据处理框架向所述多个 map 模块分发抓取的元数据消息，使得所述多个 map 模块根据接收到的元数据消息中的存储位置，获取对应的业务数据；并使得所述多个 map 模块根据所述至少一个关键值，相应的处理获取的业务数据；

所述流数据处理框架将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块，使得所述多个 reduce 模块根据所述至少一个关键值，对所述处理后的业务数据进行合并处理以得到合并处理结果；

所述流数据处理框架输出所述合并处理结果。

可选的，还包括：

所述流数据处理框架记录所述多个元数据消息在所述流数据处理框架中的处理状态，所述处理状态包括成功状态和失败状态，若一个元数据消息所对应的业务数据被作为合并处理结果输出，则这个元数据消息的处理状态为成功状态，若一个元数据消息所对应的业务数据在预定时间内未能被作为合并处理结果输出，则这个元数据消息的处理状态为失败状态；

所述流数据处理框架根据所述处理状态更新位置寄存器中的位置信息，所述位置寄存器中的位置信息为下一次所述流数据处理框架抓取元数据消息在所述消息队列中的位置信息；若所述多个元数据消息中具有失败状态的元数据消息，则所述位置寄存器中的位置信息为第一元数据消息的位置信息，所述第一元数据消息为相对于更靠近所述消息队列的队列头的具有失败状态的元数据消息，若所述多个元数据消息中不具有失败状态的元数据消息，则所述位置寄存器中的位置信息为第二元数据消息的位置信息，所述第二元数据消息为所述多个元数据消息中更靠近所述消息队列的队列尾的元数据消息。

可选的，在所述流数据处理框架再一次从所述消息队列中依次抓取元数据消息之前，还包括：

所述流数据处理框架检测缓存器是否还保存有元数据消息，所述缓存中保存所述流

数据处理框架从所述消息队列中抓取的、且尚未被分发到 map 模块的元数据消息；

若有，所述流数据处理框架从所述缓存器中题述元数据消息向所述多个 map 模块分发；

若无，执行所述流数据处理框架再一次从所述消息队列中依次抓取元数据消息。

- 5 可选的，在所述流数据处理框架向所述多个 map 模块分发抓取的元数据消息之前，还包括：

所述流数据处理框架实时计算已消耗能力，所述已消耗能力为所述流数据处理框架用于处理分发到所述多个 map 模块中的元数据消息所消耗的处理能力；

- 10 所述流数据处理框架判断所述流数据处理框架的总处理能力和所述已消耗能力之间差值是否满足预设阈值，

若满足，所述流数据处理框架向所述多个 map 模块分发抓取的元数据消息。

可选的，所述流数据处理框架将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块，包括：

- 15 若一个处理后的业务数据的数据量大于预设传输量，所述流数据处理框架在将这个处理后的业务数据向 reduce 模块分发前，对这个处理后的业务数据进行部分合并处理，以压缩这个处理后的业务数据的数据量。

一种数据处理装置，应用于流数据处理框架，所述流数据处理框架包括多个 map 模块和多个 reduce 模块，所述装置包括：

获取单元，用于获取业务需求，所述业务需求包括至少一个关键值；

- 20 抓取单元，用于从用于存储元数据消息的消息队列中依次抓取多个元数据消息，所述消息队列通过所述业务需求确定，所述元数据消息与业务消息一一对应，所述元数据消息包括对应的业务数据的存储位置，所述元数据消息依据所对应业务数据的生成时间先后，被顺序添加到所述消息队列中；

- 25 第一分发单元，用于向所述多个 map 模块分发抓取的元数据消息，使得所述多个 map 模块根据接收到的元数据消息中的存储位置，获取对应的业务数据；并使得所述多个 map 模块根据所述至少一个关键值，相应的处理获取的业务数据；

第二分发单元，用于将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块，使得所述多个 reduce 模块根据所述至少一个关键值，对所述处理后的业务数据进行合并处理以得到合并处理结果；

- 30 输出单元，用于输出所述合并处理结果。

可选的，还包括：

记录单元，用于记录所述多个元数据消息在所述流数据处理框架中的处理状态，所述处理状态包括成功状态和失败状态，若一个元数据消息所对应的业务数据被作为合并处理结果输出，则这个元数据消息的处理状态为成功状态，若一个元数据消息所对应的业务数据在预定时间内未能被作为合并处理结果输出，则这个元数据消息的处理状态为失败状态；

更新单元，用于根据所述处理状态更新位置寄存器中的位置信息，所述位置寄存器中的位置信息为下一次所述流数据处理框架抓取元数据消息在所述消息队列中的位置信息；若所述多个元数据消息中具有失败状态的元数据消息，则所述位置寄存器中的位置信息为第一元数据消息的位置信息，所述第一元数据消息为相对于更靠近所述消息队列的队列头的具有失败状态的元数据消息，若所述多个元数据消息中不具有失败状态的元数据消息，则所述位置寄存器中的位置信息为第二元数据消息的位置信息，所述第二元数据消息为所述多个元数据消息中更靠近所述消息队列的队列尾的元数据消息。

可选的，还包括：

检测单元，用于在触发所述抓取单元之前，检测缓存器是否还保存有元数据消息，所述缓存中保存所述流数据处理框架从所述消息队列中抓取的、且尚未被分发到 map 模块的元数据消息；

若有，触发所述第一分发单元；若无，触发所述抓取单元；

所述第一分发单元还用于从所述缓存器中题述元数据消息向所述多个 map 模块分发。

可选的，还包括：

计算单元，用于在触发所述第一分发单元之前，实时计算已消耗能力，所述已消耗能力为所述流数据处理框架用于处理分发到所述多个 map 模块中的元数据消息所消耗的处理能力；

判断单元，用于判断所述流数据处理框架的总处理能力和所述已消耗能力之间差值是否满足预设阈值，

若满足，触发所述第一分发单元。

可选的，所述第二分发单元，还用于若一个处理后的业务数据的数据量大于预设传输量，在将这个处理后的业务数据向 reduce 模块分发前，对这个处理后的业务数据进行部分合并处理，以压缩这个处理后的业务数据的数据量。

由上述技术方案可以看出，通过使用针对于实时数据处理的流数据处理框架，将传统的 MapReduce 的功能通过 map 模块和 reduce 模块在流数据处理框架中实现，在接收到业务需求时，从确定出的消息队列中抓取容量远比业务数据要小的元数据消息，并将元数据消息分发给 map 模块，由 map 模块通过元数据消息中的存储位置获取其相对应的业务数据，再根据业务需求进行相应处理，所述流数据处理框架将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块，使得所述多个 reduce 模块根据所述至少一个关键值，对所述处理后的业务数据进行合并处理以得到合并处理结果，再将所述合并处理结果进行输出，通过实时抓取消息队列中的元数据消息，再利用流数据处理框架的实时处理特点，处理前不再需要如传统做法中对待处理数据进行预先规划，可以使用实现了 MapReduce 功能的流数据处理框架快速的处理网络中实时产生的海量数据。

附图说明

为了更清楚地说明本发明实施例或现有技术中的技术方案，下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本发明的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动性的前提下，还可以根据这些附图获得其他的附图。

图 1 为本发明实施例提供的一种数据处理方法的方法流程图；

图 2 为本发明实施例提供的一种分发元数据消息判断方法的方法流程图；

图 3 为本发明实施例提供的一种数据处理装置的装置结构图。

具体实施方式

为使本发明实施例的目的、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，对本发明实施例中的技术方案进行清楚地描述，显然，所描述的实施例是本发明一部分实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本发明保护的范围。

随着用户在网络上的活动，互联网会实时的产生海量的数据，这些数据可以显示用户的偏好，行为规律等，这些海量数据对于互联网企业来说具有重要价值。

针对海量数据的传统处理方式是使用一种编程模型 MapReduce。然而，MapReduce 主要用于对海量数据进行离线的批量计算。也就是说在处理前，需要对待处理的数据集进行预先的规划。对于实时生成的数据，MapReduce 需要收集一定数据规模后，并预先

规划后才能处理。可见，传统的 MapReduce 不能有效的解决目前需要快速处理网络中实时产生的海量数据的问题。

然而，在互联网的具体应用场景中，有一些情况下，需要对应用中不断产生的新数据进行实时计算，并能快速给出计算结果。比如，卖家投放广告或者微博上用户添加关注等。以卖家投放广告这一场景为例，卖家在定制好自己感兴趣的人群规则后，希望能够尽快的将广告投放到相关人群，给自己带来收益。假设卖家设置的人群规则集合用 g_1 、 g_2 ...标识，用户用 u_1 、 u_2 ...来标识。那么符合这个人群规则集合的用户集合 ($u_1, u_2, u_3 \dots$) 可能会有数千万，甚至上亿。同时，为了知道具体某个用户符合哪些具体的人群规则，卖家需要通过对互联网中的数据进行数据处理得到用户(u)→用户集合 ($g_1, g_2, g_3 \dots$) 的对应关系。而且对于处理系统来说，每天需要处理大量卖家设置的成千上万的广告规则，这就要求对互联网中实时产生数据的数据处理速度能够快速，最好能够达到一个用户正在浏览相关网页时，卖家就可以将与这个相关网页有关系的广告展示给这个用户的效果。这种处理效果显然是使用传统的 MapReduce 对数据进行离线处理无法达成的。可见，MapReduce 不能有效的解决目前需要快速处理网络中实时产生的海量数据的问题。

为此，本发明实施例提供了一种数据处理方法和装置，通过使用针对于实时数据处理的流数据处理框架，将传统的 MapReduce 的功能通过 map 模块和 reduce 模块在流数据处理框架中实现，在接收到业务需求时，从确定出的消息队列中抓取容量远比业务数据要小的元数据消息，并将元数据消息分发给 map 模块，由 map 模块通过元数据消息中的存储位置获取其相对应的业务数据，再根据业务需求进行相应处理，所述流数据处理框架将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块，使得所述多个 reduce 模块根据所述至少一个关键值，对所述处理后的业务数据进行合并处理以得到合并处理结果，再将所述合并处理结果进行输出，通过实时抓取消息队列中的元数据消息，再利用流数据处理框架的实时处理特点，处理前不再需要如传统做法中对待处理数据进行预先规划，可以使用实现了 MapReduce 功能的流数据处理框架快速的处理网络中实时产生的海量数据。

在描述本发明实施例之前，需要对如何发现解决上述技术问题的过程进行说明。发明人仔细分析目前这种技术问题，确定出产生这种技术问题的应用场景特点，即主要是需要对实时产生的数据进行实时处理的特点。

虽然使用 MapReduce 的方式在大多数场景下也可以有效处理海量网络数据，获取针对业务需求的准确处理结果。但是针对需要对实时产生的数据及时、快速处理的场景，

由于 MapReduce 需要对待处理的数据进行预先规划，再加上使用一次 MapReduce 进行数据处理的开销较大，故一般是先收集一定规模的这种实时产生的数据（例如可以与 MapReduce 的处理量相当，以节约运算成本），然后再对收集的数据进行预先规划和处理。也就是说，虽然能够得出符合需求的处理结果，但是时效性很差，或者说不能及时的给出处理结果。这样可能会导致向用户推送广告失败，或者推送时，用户早已离开相应的页面了，导致没法达到最佳的推送广告的效果。发明人在发现这个隐蔽的技术缺陷后，为了弥补传统方式处理时缺乏时效性的特点，发明人有针对性的选取了用于处理流数据的流数据处理框架，将流数据处理框架的处理特点和 MapReduce 的处理特点进行有机的结合，即将流数据处理框架对数据的实时处理能力以及 MapReduce 易于处理海量数据的特点进行有机的结合，以流数据处理框架为基础，在流数据处理框架中的功能模块上实现 MapReduce 的函数处理能力，从而实现了一种实时的 MapReduce 计算方法。

在本发明实施例中所所述的流数据处理框架，可以理解作为一种用于处理流数据的计算机程序，或者一种用于处理流数据的计算机服务。所述流数据处理框架可以安装部署在一台服务器中、多台服务器中或者服务器集群中。服务器通过运行部署在自身的流数据处理框架，以实现对流数据的处理。本发明并不限定所述流数据处理框架的具体类型，例如可以为 Spark Streaming（一种用于实时计算的计算框架）或者 Storm（一种实时处理大数据的处理框架）。

以 Storm 为例说明与 MapReduce 的功能结合，Storm 可被用于“流处理”之中，实时处理数据。Storm 的模块中主要包括 Spout（类似数据源）模块和 Bolt（数据处理模块）模块。根据模块的特点，可以将 Spout 模块用于从消息队列中抓取元数据消息，以及用于将抓取的元数据消息进行分发，而在一部分 Bolt 模块上实现 map 函数的功能，在一部分 Bolt 模块上实现 reduce 函数的功能。其中，实现了 map 函数功能的 Bolt 模块可以标识为 map-bolt 模块，实现了 reduce 函数功能的 Bolt 模块可以标识为 reduce-bolt 模块。

由 map-bolt 模块接收 Spout 模块分发的元数据消息，并以此获取需要处理的业务数据。在 map-bolt 模块和 reduce-bolt 模块之间，依然可以使用 Shuffle 的方式从 map-bolt 模块将处理后的数据发向 reduce-bolt 模块。reduce-bolt 模块实现对处理后的数据的合并处理。Storm 将所述合并处理的结果进行输出，从而实现了快速的处理网络中实时产生的海量数据。

针对其他流数据处理框架，例如 Spark Streaming，同理，也可以将 map 函数和 reduce 函数的功能在 Spark Streaming 中相应的模块上实现。这里不再一一穷举。

通过所述流数据处理框架输出的合并处理结果可以根据不同的场景需求进行后续的处理，例如可以保持到持久化存储器中，也可以添加到数据队列中等。

图 1 为本发明实施例提供的一种数据处理方法的方法流程图，应用于流数据处理框架，所述流数据处理框架包括多个 map 模块和多个 reduce 模块，所述方法包括：

5 S101：所述流数据处理框架获取业务需求，所述业务需求包括至少一个关键值。

举例说明，这里的业务需求可以理解为卖家的投放广告需求，其中的关键值可以包括 key、value 或者 key/value pair。以卖家投放广告为例，关键值可以为特定人群规则、特定人群规则集合，也可以是特定人群规则集合和用户之间的对应关系。

10 S102：所述流数据处理框架从用于存储元数据消息的消息队列中依次抓取多个元数据消息，所述消息队列通过所述业务需求确定，所述元数据消息与业务消息一一对应，所述元数据消息包括对应的业务数据的存储位置，所述元数据消息依据所对应业务数据的生成时间先后，被顺序添加到所述消息队列中。

15 举例说明，这里的消息队列可以是根据业务需求确定的，不同的业务需求可以确定出不同的消息队列。所述消息队列中主要包括元数据消息，网络系统中生成一个业务数据，会对应生成一个用于标识这个业务数据的元数据消息，这个元数据消息的数据量一般会远小于对应的业务数据的数据量大小，这个元数据消息中可以包括对应业务数据的存储位置，例如用于标识业务数据保存在哪一个服务器的哪一个位置中，或者保存在分布式文件系统的哪一个位置等。这个元数据消息还可以包括对应业务数据的数据类型，用于标识对应业务数据的数据所属数据类别、数据特点等。

20 在抓取元数据消息过程中，所述流数据处理框架可以依据所述消息队列中元数据消息的排列顺序，依次抓取。所述消息队列的排列顺序一般是按照新入队的元数据消息排列在所述消息队列的队列尾部。所述流数据处理框架一次抓取的元数据消息的数量可以预先设定，一般可以设置为大于所述流数据处理框架处理能力的值。所述流数据处理框架可以将一次抓取的元数据消息，分为多批次的向 map 模块分发，从而降低了所述流数
25 据处理框架向所述消息队列抓取元数据消息的抓取频率。

在所述流数据处理框架一次抓取的元数据消息过多，不能一次分发出去的情况下，本发明实施例提供一种缓存思路，可以将抓取后尚未分发出去的元数据消息缓存到缓冲器中，相应的，在所述流数据处理框架再次向所述消息队列抓取元数据消息之前，可以增加判断缓存中是否有元数据消息的过程。

30 可选的，在所述流数据处理框架再一次从所述消息队列中依次抓取元数据消息之前，

还包括：

所述流数据处理框架检测缓存器是否还保存有元数据消息，所述缓存中保存所述流数据处理框架从所述消息队列中抓取的、且尚未被分发到 map 模块的元数据消息；

若有，所述流数据处理框架从所述缓存器中题述元数据消息向所述多个 map 模块分
5 发；

若无，执行所述流数据处理框架再一次从所述消息队列中依次抓取元数据消息。

也就是说，在所述流数据处理框架分发元数据消息时，优先分发保存在缓冲器中的元数据消息。这样可以尽量按照消息队列中的排列顺序对业务数据进行处理，保证时序性。

10 需要注意的是，由于在完成对一个业务需求的数据处理中，所述流数据处理框架一般需要多次从所述消息队列中抓取元数据消息，为了确保数据处理结果的准确性，需要保证不会有业务数据被遗漏处理。那么，准确的确定出下一次抓取元数据消息时，在所述消息队列的抓取位置是非常必要的。为此，本发明实施例提供了一种通过记录元数据消息处理状态的方式确定下一次抓取位置。

15 可选的，所述流数据处理框架记录所述多个元数据消息在所述流数据处理框架中的处理状态，所述处理状态包括成功状态和失败状态，若一个元数据消息所对应的业务数据被作为合并处理结果输出，则这个元数据消息的处理状态为成功状态，若一个元数据消息所对应的业务数据在预定时间内未能被作为合并处理结果输出，则这个元数据消息的处理状态为失败状态。

20 所述流数据处理框架根据所述处理状态更新位置寄存器中的位置信息，所述位置寄存器中的位置信息为下一次所述流数据处理框架抓取元数据消息在所述消息队列中的位置信息；若所述多个元数据消息中具有失败状态的元数据消息，则所述位置寄存器中的位置信息为第一元数据消息的位置信息，所述第一元数据消息为相对于更靠近所述消息队列的队列头的具有失败状态的元数据消息，若所述多个元数据消息中不具有失败状态
25 的元数据消息，则所述位置寄存器中的位置信息为第二元数据消息的位置信息，所述第二元数据消息为所述多个元数据消息中更靠近所述消息队列的队列尾的元数据消息。

举例说明，所述位置寄存器可以是一种持久化的存储单元，例如可以利用 Zookeeper（一个分布式、开放源码的应用程序协调服务）提供类似的存储服务。通过持久化的存储方式，即使所述流数据处理框架在数据处理过程中出现重启、断电等情况，在再次启
30 动时，都可以根据所述位置寄存器中存储的位置信息，从所述消息队列的所述位置寄存

器中的位置信息开始继续进行元数据消息抓取。

一个元数据消息从开始向 map 模块分发（例如 S103）开始，直到这个元数据消息对应的业务数据经过处理后被所述流数据处理框架输出（S105）之间的这段过程中，若任何一个步骤没有成功，都可以理解为这个元数据消息的处理状态为失败状态。例如可以
5 为接收到这个元数据消息的 map 模块根据其携带的存储位置未能获取对应的业务数据，也可以为 map 模块在处理这个元数据消息对应的业务数据不成功，也可以为所述流数据处理框架未能将通过 map 模块处理后的业务数据（对应这个元数据消息的业务数据）分发到 reduce 模块等等。需要注意的是，这里的失败处理明确的不成功（例如接收到确认处理失败的消息）以外，还可以包括处理超时的情况，例如在上述过程中的一个处理步
10 骤中长时间无响应的情况。

所述流数据处理框架可以通过接收对于元数据消息的确认消息（缩写：ACK）和失败消息（英文：Fail）来确认一个元数据消息的处理状态是成功状态（接收到确认消息）还是失败状态（接收到失败消息）。

需要注意的是，如果一个元数据消息在多次尝试后，处理状态依然保持失败状态时，
15 为了提高处理效率，可以将这个元数据消息舍弃，不再进行相应处理，而在位置寄存器中继续更新其他元数据消息的位置信息。

S103：所述流数据处理框架向所述多个 map 模块分发抓取的元数据消息，使得所述多个 map 模块根据接收到的元数据消息中的存储位置，获取对应的业务数据；并使得所述多个 map 模块根据所述至少一个关键值，相应的处理获取的业务数据。

举例说明，在分发元数据消息时，为了保持处理的均衡性，所述流数据处理框架可以将一批元数据消息较为平均的分给各个 map 模块，例如一批 10 个元数据消息，所述流数据处理框架只有 5 个 map 模块工作，那么可以一个 map 模块分 2 个元数据消息。平均分发的好处在于，各个 map 模块可以在差不多时间内完成处理任务，以便接收所述流数据处理框架的下一次元数据消息的分发。

25 业务数据可以存储在分布式文件系统中，map 模块根据获取的元数据 u 消息中存储位置，可以从分布式文件系统中的相应存储位置获取对应业务数据。map 模块在获取业务数据后，根据业务需求中的关键值，从业务数据中确定出与关键值相关的数据关系。例如，业务需求的关键值是用用户(u1)->用户集合（g1,g2,g3...）的对应关系，那么 map 模块通过处理，从业务数据中可以确定出 u1->g1，u1->g2...等与关键值相关的数据关系。

30 S104：所述流数据处理框架将所述多个 map 模块处理后的业务数据分发到所述多个

reduce 模块,使得所述多个 reduce 模块根据所述至少一个关键值,对所述处理后的业务数据进行合并处理以得到合并处理结果。

举例说明,所述流数据处理框架将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块的过程,类似于 MapReduce 中的 Shuffle 过程。其中,可以将具有相同关键值的、处理后的业务数据发给同一个 reduce 模块进行合并处理。

需要注意的是,有些情况下,通过 map 模块处理后的业务数据的数据量将会很大,如果不进行处理直接向 reduce 模块传输的话,会给传输环节带来较大的传输压力,甚至导致传输错误等问题。为此,可选的,所述流数据处理框架将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块,还包括:

若一个处理后的业务数据的数据量大于预设传输量,所述流数据处理框架在将这个处理后的业务数据向 reduce 模块分发前,对这个处理后的业务数据进行部分合并处理,以压缩这个处理后的业务数据的数据量。

S105: 所述流数据处理框架输出所述合并处理结果。

举例说明,所述多个 reduce 模块的合并处理结果相当于针对所述业务需求的处理结果,所述流数据处理框架可以将所述合并处理结果进行输出,再由系统根据不同的场景、业务需求对输出的所述合并处理结果进行后续的处理,例如将所述合并处理结果存储到持久化存储器中等。

可见,通过使用针对于实时数据处理的流数据处理框架,将传统的 MapReduce 的功能通过 map 模块和 reduce 模块在流数据处理框架中实现,在接收到业务需求时,从确定的消息队列中抓取容量远比业务数据要小的元数据消息,并将元数据消息分发给 map 模块,由 map 模块通过元数据消息中的存储位置获取其相对应的业务数据,再根据业务需求进行相应处理,所述流数据处理框架将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块,使得所述多个 reduce 模块根据所述至少一个关键值,对所述处理后的业务数据进行合并处理以得到合并处理结果,再将所述合并处理结果进行输出,通过实时抓取消息队列中的元数据消息,再利用流数据处理框架的实时处理特点,处理前不再需要如传统做法中对待处理数据进行预先规划,可以使用实现了 MapReduce 功能的流数据处理框架快速的处理网络中实时产生的海量数据。

在处理过程中,所述流数据处理框架可以通过分析自身的剩余处理能力,判断向所述多个 map 模块分发元数据消息的时机。

可选的,在图 1 所对应实施例的基础上,图 2 为本发明实施例提供的一种分发元数

据消息判断方法的方法流程图，在所述流数据处理框架向所述多个 map 模块分发抓取的元数据消息之前，还包括：

S201：所述流数据处理框架实时计算已消耗能力，所述已消耗能力为所述流数据处理框架用于处理分发到所述多个 map 模块中的元数据消息所消耗的处理能力；

5 S202：所述流数据处理框架判断所述流数据处理框架的总处理能力和所述已消耗能力之间差值是否满足预设阈值，若满足，执行 S203。

S203：所述流数据处理框架向所述多个 map 模块分发抓取的元数据消息。

举例说明，所述流数据处理框架的分发的速率需要注意控制，如果过快，可能会导致短时间内进入到后续 Map 模块的消息数量剧增，导致 Map 模块和 Reduce 模块用于处理相应业务数据所消耗的系统资源超出了所述流数据处理框架或者所在系统的承受能力；如果过慢，又会使得所述消息队列中的元数据消息过多，处理的实时性降低。为了解决这个问题，所述流数据处理框架实时计算已消耗能力。所述已消耗能力可以理解为目前所述流数据处理框架执行数据处理所消耗的能力，所述流数据处理框架的总处理能力可以理解为所述流数据处理框架能够实现的最大处理功能时所消耗的能力。所述已消耗能力和所述总处理能力可以通过对内存、CPU、网卡等资源进行衡量。

假设所述已消耗能力为 R_{curr} ，所述总处理能力为 R_{max} ，那么所述流数据处理框架的剩余能力，也就是所述流数据处理框架的总处理能力和所述已消耗能力之间差值 $R_{left} = R_{max} - R_{curr}$ 。如果所述流数据处理框架需要分发的元数据消息所要消耗的能力值（也就是所述预设阈值）大于 R_{left} ，这时再进行元数据消息的分发可能会引起所述流数据处理框架处理延时，降低处理效率。故在所述差值满足预设阈值时才继续向所述多个 map 模块分发元数据消息可以更为有效的利用系统资源，提高数据处理的效率。

需要注意的是，若所述差值不满足所述预设阈值，也就是说目前所述流数据处理框架的剩余处理能力不足以处理本次分发的元数据消息时，可以进行等待，一旦所述流数据处理框架完成对本批次元数据消息处理后，所述流数据处理框架将会释放出相应的处理能力。当等待所述差值满足所述预设阈值之后，所述流数据处理框架可以再一次向所述多个 map 模块分发抓取的元数据消息。

图 3 为本发明实施例提供的一种数据处理装置的装置结构图，应用于流数据处理框架，所述流数据处理框架包括多个 map 模块和多个 reduce 模块，所述装置包括：

30 获取单元 301，用于获取业务需求，所述业务需求包括至少一个关键值。

抓取单元 302, 用于从用于存储元数据消息的消息队列中依次抓取多个元数据消息, 所述消息队列通过所述业务需求确定, 所述元数据消息与业务消息一一对应, 所述元数据消息包括对应的业务数据的存储位置, 所述元数据消息依据所对应业务数据的生成时间先后, 被顺序添加到所述消息队列中。

- 5 在所述流数据处理框架一次抓取的元数据消息过多, 不能一次分发出去的情况下, 本发明实施例提供一种缓存思路, 可以将抓取后尚未分发出去的元数据消息缓存到缓冲器中, 可选的, 还包括:

10 检测单元, 用于在触发所述抓取单元 302 之前, 检测缓存器是否还保存有元数据消息, 所述缓存中保存所述流数据处理框架从所述消息队列中抓取的、且尚未被分发到 map 模块的元数据消息。

若有, 触发所述第一分发单元 303; 若无, 触发所述抓取单元 302。

所述第一分发单元 303 还用于从所述缓存器中题述元数据消息向所述多个 map 模块分发。

15 也就是说, 在所述流数据处理框架分发元数据消息时, 优先分发保存在缓冲器中的元数据消息。这样可以尽量按照消息队列中的排列顺序对业务数据进行处理, 保证时序性。

20 需要注意的是, 由于在完成对一个业务需求的数据处理中, 所述流数据处理框架一般需要多次从所述消息队列中抓取元数据消息, 为了确保数据处理结果的准确性, 需要保证不会有业务数据被遗漏处理。那么, 准确的确定出下一次抓取元数据消息时, 在所述消息队列的抓取位置是非常必要的。为此, 本发明实施例提供了一种通过记录元数据消息处理状态的方式确定下一次抓取位置。可选的, 还包括:

25 记录单元, 用于记录所述多个元数据消息在所述流数据处理框架中的处理状态, 所述处理状态包括成功状态和失败状态, 若一个元数据消息所对应的业务数据被作为合并处理结果输出, 则这个元数据消息的处理状态为成功状态, 若一个元数据消息所对应的业务数据在预定时间内未能被作为合并处理结果输出, 则这个元数据消息的处理状态为失败状态;

30 更新单元, 用于根据所述处理状态更新位置寄存器中的位置信息, 所述位置寄存器中的位置信息为下一次所述流数据处理框架抓取元数据消息在所述消息队列中的位置信息; 若所述多个元数据消息中具有失败状态的元数据消息, 则所述位置寄存器中的位置信息为第一元数据消息的位置信息, 所述第一元数据消息为相对于更靠近所述消息队列

的队列头的具有失败状态的元数据消息，若所述多个元数据消息中不具有失败状态的元数据消息，则所述位置寄存器中的位置信息为第二元数据消息的位置信息，所述第二元数据消息为所述多个元数据消息中更靠近所述消息队列的队列尾的元数据消息。

第一分发单元 303，用于向所述多个 map 模块分发抓取的元数据消息，使得所述多个 map 模块根据接收到的元数据消息中的存储位置，获取对应的业务数据；并使得所述多个 map 模块根据所述至少一个关键值，相应的处理获取的业务数据。

第二分发单元 304，用于将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块，使得所述多个 reduce 模块根据所述至少一个关键值，对所述处理后的业务数据进行合并处理以得到合并处理结果。

需要注意的是，有些情况下，通过 map 模块处理后的业务数据的数据量将会很大，如果不进行处理直接向 reduce 模块传输的话，会给传输环节带来较大的传输压力，甚至导致传输错误等问题。为此，可选的，所述第二分发单元，还用于若一个处理后的业务数据的数据量大于预设传输量，在将这个处理后的业务数据向 reduce 模块分发前，对这个处理后的业务数据进行部分合并处理，以压缩这个处理后的业务数据的数据量。

输出单元 305，用于输出所述合并处理结果。

可见，通过使用针对于实时数据处理的流数据处理框架，将传统的 MapReduce 的功能通过 map 模块和 reduce 模块在流数据处理框架中实现，在接收到业务需求时，从确定出的消息队列中抓取容量远比业务数据要小的元数据消息，并将元数据消息分发给 map 模块，由 map 模块通过元数据消息中的存储位置获取其相对应的业务数据，再根据业务需求进行相应处理，所述流数据处理框架将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块，使得所述多个 reduce 模块根据所述至少一个关键值，对所述处理后的业务数据进行合并处理以得到合并处理结果，再将所述合并处理结果进行输出，通过实时抓取消息队列中的元数据消息，再利用流数据处理框架的实时处理特点，处理前不再需要如传统做法中对待处理数据进行预先规划，可以使用实现了 MapReduce 功能的流数据处理框架快速的处理网络中实时产生的海量数据。

在处理过程中，所述流数据处理框架可以通过分析自身的剩余处理能力，判断向所述多个 map 模块分发元数据消息的时机。

可选的，还包括：

计算单元，用于在触发所述第一分发单元 303 之前，实时计算已消耗能力，所述已消耗能力为所述流数据处理框架用于处理分发到所述多个 map 模块中的元数据消息所消

耗的处理能力；

判断单元，用于判断所述流数据处理框架的总处理能力和所述已消耗能力之间差值是否满足预设阈值，

若满足，触发所述第一分发单元。

5 需要注意的是，若所述差值不满足所述预设阈值，也就是说目前所述流数据处理框架的剩余处理能力不足以处理本次分发的元数据消息时，可以进行等待，一旦所述流数据处理框架完成对本批次元数据消息处理后，所述流数据处理框架将会释放出相应的处理能力。当等待所述差值满足所述预设阈值之后，所述流数据处理框架可以再一次向所述多个 map 模块分发抓取的元数据消息。

10 本领域普通技术人员可以理解：实现上述方法实施例的全部或部分步骤可以通过程序指令相关的硬件来完成，前述程序可以存储于一计算机可读取存储介质中，该程序在执行时，执行包括上述方法实施例的步骤；而前述的存储介质可以是下述介质中的至少一种：只读存储器（英文：read-only memory，缩写：ROM）、RAM、磁碟或者光盘等各种可以存储程序代码的介质。

15 需要说明的是，本说明书中的各个实施例均采用递进的方式描述，各个实施例之间相同相似的部分互相参见即可，每个实施例重点说明的都是与其他实施例的不同之处。尤其，对于设备及系统实施例而言，由于其基本相似于方法实施例，所以描述得比较简单，相关之处参见方法实施例的部分说明即可。以上所描述的设备及系统实施例仅仅是示意性的，其中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
20 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模块来实现本实施例方案的目的。本领域普通技术人员在不付出创造性劳动的情况下，即可以理解并实施。

 以上所述，仅为本发明较佳的具体实施方式，但本发明的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本发明揭露的技术范围内，可轻易想到的变化或替换，
25 都应涵盖在本发明的保护范围之内。因此，本发明的保护范围应该以权利要求的保护范围为准。

权 利 要 求 书

1、一种数据处理方法，其特征在于，应用于流数据处理框架，所述流数据处理框架包括多个映射 map 模块和多个归约 reduce 模块，所述方法包括：

所述流数据处理框架获取业务需求，所述业务需求包括至少一个关键值；

5 所述流数据处理框架从用于存储元数据消息的消息队列中依次抓取多个元数据消息，所述消息队列通过所述业务需求确定，所述元数据消息与业务消息一一对应，所述元数据消息包括对应的业务数据的存储位置，所述元数据消息依据所对应业务数据的生成时间先后，被顺序添加到所述消息队列中；

所述流数据处理框架向所述多个 map 模块分发抓取的元数据消息，使得所述多个
10 map 模块根据接收到的元数据消息中的存储位置，获取对应的业务数据；并使得所述多个 map 模块根据所述至少一个关键值，相应的处理获取的业务数据；

所述流数据处理框架将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块，使得所述多个 reduce 模块根据所述至少一个关键值，对所述处理后的业务数据进行合并处理以得到合并处理结果；

15 所述流数据处理框架输出所述合并处理结果。

2、根据权利要求 1 所述的方法，其特征在于，还包括：

所述流数据处理框架记录所述多个元数据消息在所述流数据处理框架中的处理状态，所述处理状态包括成功状态和失败状态，若一个元数据消息所对应的业务数据被作为合并处理结果输出，则这个元数据消息的处理状态为成功状态，若一个元数据消息所
20 对应的业务数据在预定时间内未能被作为合并处理结果输出，则这个元数据消息的处理状态为失败状态；

所述流数据处理框架根据所述处理状态更新位置寄存器中的位置信息，所述位置寄存器中的位置信息为下一次所述流数据处理框架抓取元数据消息在所述消息队列中的位置信息；若所述多个元数据消息中具有失败状态的元数据消息，则所述位置寄存器中的位置信息为第一元数据消息的位置信息，所述第一元数据消息为相对于更靠近所述消息队列的队列头的具有失败状态的元数据消息，若所述多个元数据消息中不具有失败状态的元数据消息，则所述位置寄存器中的位置信息为第二元数据消息的位置信息，所述
25 第二元数据消息为所述多个元数据消息中更靠近所述消息队列的队列尾的元数据消息。

3、根据权利要求 1 或 2 所述的方法，其特征在于，在所述流数据处理框架再一次
30 从所述消息队列中依次抓取元数据消息之前，还包括：

所述流数据处理框架检测缓存器是否还保存有元数据消息，所述缓存中保存所述流数据处理框架从所述消息队列中抓取的、且尚未被分发到 map 模块的元数据消息；

若有，所述流数据处理框架从所述缓存器中题述元数据消息向所述多个 map 模块分发；

5 若无，执行所述流数据处理框架再一次从所述消息队列中依次抓取元数据消息。

4、根据权利要求 1 或 2 所述的方法，其特征在于，在所述流数据处理框架向所述多个 map 模块分发抓取的元数据消息之前，还包括：

所述流数据处理框架实时计算已消耗能力，所述已消耗能力为所述流数据处理框架用于处理分发到所述多个 map 模块中的元数据消息所消耗的处理能力；

10 所述流数据处理框架判断所述流数据处理框架的总处理能力和所述已消耗能力之间差值是否满足预设阈值，

若满足，所述流数据处理框架向所述多个 map 模块分发抓取的元数据消息。

5、根据权利要求 1 所述的方法，其特征在于，所述流数据处理框架将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块，包括：

15 若一个处理后的业务数据的数据量大于预设传输量，所述流数据处理框架在将这个处理后的业务数据向 reduce 模块分发前，对这个处理后的业务数据进行部分合并处理，以压缩这个处理后的业务数据的数据量。

6、一种数据处理装置，其特征在于，应用于流数据处理框架，所述流数据处理框架包括多个映射 map 模块和多个归约 reduce 模块，所述装置包括：

20 获取单元，用于获取业务需求，所述业务需求包括至少一个关键值；

抓取单元，用于从用于存储元数据消息的消息队列中依次抓取多个元数据消息，所述消息队列通过所述业务需求确定，所述元数据消息与业务消息一一对应，所述元数据消息包括对应的业务数据的存储位置，所述元数据消息依据所对应业务数据的生成时间先后，被顺序添加到所述消息队列中；

25 第一分发单元，用于向所述多个 map 模块分发抓取的元数据消息，使得所述多个 map 模块根据接收到的元数据消息中的存储位置，获取对应的业务数据；并使得所述多个 map 模块根据所述至少一个关键值，相应的处理获取的业务数据；

第二分发单元，用于将所述多个 map 模块处理后的业务数据分发到所述多个 reduce 模块，使得所述多个 reduce 模块根据所述至少一个关键值，对所述处理后的业务数据进行

30 行合并处理以得到合并处理结果；

输出单元，用于输出所述合并处理结果。

7、根据权利要求 6 所述的装置，其特征在于，还包括：

记录单元，用于记录所述多个元数据消息在所述流数据处理框架中的处理状态，所述处理状态包括成功状态和失败状态，若一个元数据消息所对应的业务数据被作为合并
5 处理结果输出，则这个元数据消息的处理状态为成功状态，若一个元数据消息所对应的业务数据在预定时间内未能被作为合并处理结果输出，则这个元数据消息的处理状态为失败状态；

更新单元，用于根据所述处理状态更新位置寄存器中的位置信息，所述位置寄存器中的位置信息为下一次所述流数据处理框架抓取元数据消息在所述消息队列中的位置
10 信息；若所述多个元数据消息中具有失败状态的元数据消息，则所述位置寄存器中的位置信息为第一元数据消息的位置信息，所述第一元数据消息为相对于更靠近所述消息队列的队列头的具有失败状态的元数据消息，若所述多个元数据消息中不具有失败状态的元数据消息，则所述位置寄存器中的位置信息为第二元数据消息的位置信息，所述第二元数据消息为所述多个元数据消息中更靠近所述消息队列的队列尾的元数据消息。

15 8、根据权利要求 6 或 7 所述的装置，其特征在于，还包括：

检测单元，用于在触发所述抓取单元之前，检测缓存器是否还保存有元数据消息，所述缓存中保存所述流数据处理框架从所述消息队列中抓取的、且尚未被分发到 map 模块的元数据消息；

若有，触发所述第一分发单元；若无，触发所述抓取单元；

20 所述第一分发单元还用于从所述缓存器中题述元数据消息向所述多个 map 模块分发。

9、根据权利要求 6 或 7 所述的装置，其特征在于，还包括：

计算单元，用于在触发所述第一分发单元之前，实时计算已消耗能力，所述已消耗能力为所述流数据处理框架用于处理分发到所述多个 map 模块中的元数据消息所消耗
25 的处理能力；

判断单元，用于判断所述流数据处理框架的总处理能力和所述已消耗能力之间差值是否满足预设阈值，

若满足，触发所述第一分发单元。

10、根据权利要求 6 所述的装置，其特征在于，所述第二分发单元，还用于若一个
30 处理后的业务数据的数据量大于预设传输量，在将这个处理后的业务数据向 reduce 模块

分发前，对这个处理后的业务数据进行部分合并处理，以压缩这个处理后的业务数据的数据量。

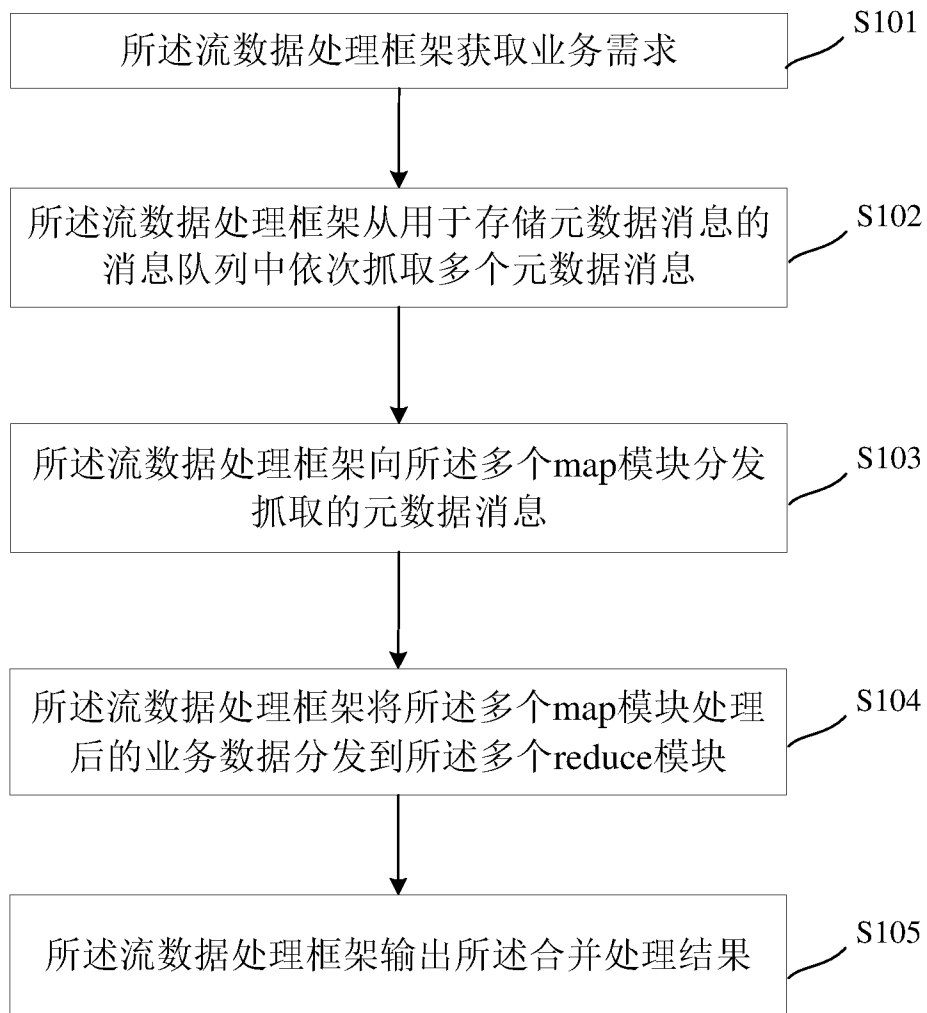


图 1

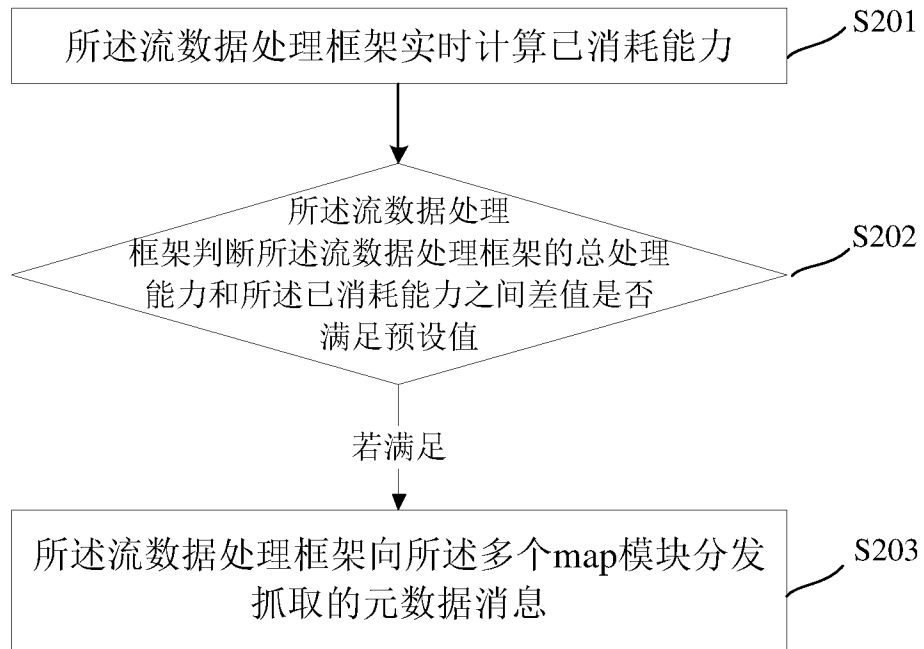


图 2

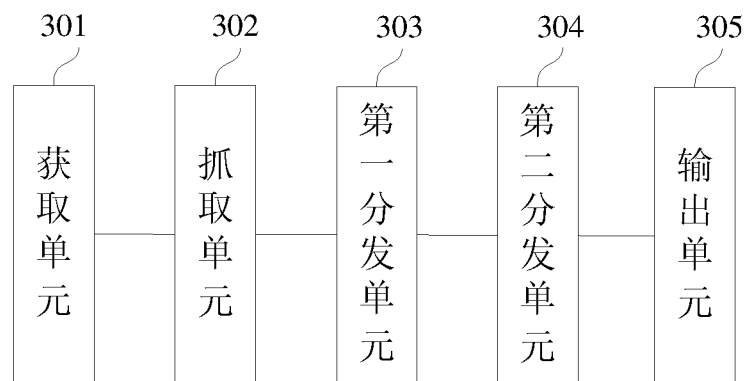


图 3

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2016/106580

A. CLASSIFICATION OF SUBJECT MATTER

G06F 17/30 (2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F 17/30

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CPRSABS; CNTXT; DWPI: map, reduce, meta data, real time

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CN 104903894 A (FACEBOOK, INC.), 09 September 2015 (09.09.2015), description, paragraphs 72-78	1-10
A	CN 103530182 A (HAINAN UNIVERSITY), 22 January 2014 (22.01.2014), the whole document	1-10
A	CN 104951509 A (INSTITUTE OF INFORMATION ENGINEERING, CHINESE ACADEMY OF SCIENCES), 30 September 2015 (30.09.2015), the whole document	1-10

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:	“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
“A” document defining the general state of the art which is not considered to be of particular relevance	“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
“E” earlier application or patent but published on or after the international filing date	“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	“&” document member of the same patent family
“O” document referring to an oral disclosure, use, exhibition or other means	
“P” document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search 06 February 2017 (06.02.2017)	Date of mailing of the international search report 16 February 2017 (16.02.2017)
Name and mailing address of the ISA/CN: State Intellectual Property Office of the P. R. China No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088, China Facsimile No.: (86-10) 62019451	Authorized officer YAN, Shiyong Telephone No.: (86-10) 62411767

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2016/106580

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN 104903894 A	09 September 2015	IL 239717 D0	31 August 2015
		CA 2896855 A1	10 July 2014
		US 2016188677 A1	30 June 2016
		EP 2752779 B1	29 June 2016
		US 2015261831 A1	17 September 2015
		IL 239717 A	31 August 2015
		WO 2014107359 A1	10 July 2014
		AU 2013371448 A1	23 July 2015
		KR 20150104585 A	15 September 2015
		US 9081826 B2	14 July 2015
		EP 3096250 A1	23 November 2016
		CA 2896855 C	01 November 2016
		US 9361344 B2	07 June 2016
		MX 2015008799 A	25 April 2016
		EP 2752779 A2	09 July 2014
		EP 2752779 A3	18 February 2015
		US 2014195558 A1	10 July 2014
		JP 2016509294 A	24 March 2016
		IN 201503844 P4	01 July 2016
CN 103530182 A	22 January 2014	None	
CN 104951509 A	30 September 2015	None	

A. 主题的分类 G06F 17/30 (2006.01) i 按照国际专利分类 (IPC) 或者同时按照国家分类和 IPC 两种分类		
B. 检索领域 检索的最低限度文献 (标明分类系统和分类号) G06F17/30 包含在检索领域中的除最低限度文献以外的检索文献 在国际检索时查阅的电子数据库 (数据库的名称, 和使用的检索词 (如使用)) CPRSABS;CNTXT;DWPI:元数据, 实时, map, reduce, meta data, real time		
C. 相关文件		
类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求
X	CN 104903894 A (脸谱公司) 2015年 9月 9日 (2015 - 09 - 09) 说明书72-78段	1-10
A	CN 103530182 A (海南大学) 2014年 1月 22日 (2014 - 01 - 22) 全文	1-10
A	CN 104951509 A (中国科学院信息工程研究所) 2015年 9月 30日 (2015 - 09 - 30) 全文	1-10
☐ 其余文件在C栏的续页中列出。 ☒ 见同族专利附件。		
* 引用文件的具体类型: “A” 认为不特别相关的表示了现有技术一般状态的文件 “E” 在国际申请日的当天或之后公布的在先申请或专利 “L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件 (如具体说明的) “O” 涉及口头公开、使用、展览或其他方式公开的文件 “P” 公布日先于国际申请日但迟于所要求的优先权日的文件 “T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件 “X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性 “Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性 “&” 同族专利的文件		
国际检索实际完成的日期 2017年 2月 6日		国际检索报告邮寄日期 2017年 2月 16日
ISA/CN的名称和邮寄地址 中华人民共和国国家知识产权局 (ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088 传真号 (86-10)62019451		受权官员 颜世莹 电话号码 (86-10)62411767

国际检索报告
关于同族专利的信息

国际申请号

PCT/CN2016/106580

检索报告引用的专利文件			公布日 (年/月/日)	同族专利			公布日 (年/月/日)
CN	104903894	A	2015年 9月 9日	IL	239717	D0	2015年 8月 31日
				CA	2896855	A1	2014年 7月 10日
				US	2016188677	A1	2016年 6月 30日
				EP	2752779	B1	2016年 6月 29日
				US	2015261831	A1	2015年 9月 17日
				IL	239717	A	2015年 8月 31日
				WO	2014107359	A1	2014年 7月 10日
				AU	2013371448	A1	2015年 7月 23日
				KR	20150104585	A	2015年 9月 15日
				US	9081826	B2	2015年 7月 14日
				EP	3096250	A1	2016年 11月 23日
				CA	2896855	C	2016年 11月 1日
				US	9361344	B2	2016年 6月 7日
				MX	2015008799	A	2016年 4月 25日
				EP	2752779	A2	2014年 7月 9日
				EP	2752779	A3	2015年 2月 18日
				US	2014195558	A1	2014年 7月 10日
				JP	2016509294	A	2016年 3月 24日
				IN	201503844	P4	2016年 7月 1日
CN	103530182	A	2014年 1月 22日	无			
CN	104951509	A	2015年 9月 30日	无			

表 PCT/ISA/210 (同族专利附件) (2009年7月)