(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
2020 年 4 月 16 日 (16.04.2020)

WIPO | PCT

(10) 国际公布号
WO 2020/073687 A1

(51) 国际专利分类号:
G06F 16/22 (2019.01)

(21) 国际申请号: PCT/CN2019/092893

(22) 国际申请日: 2019 年 6 月 26 日 (26.06.2019)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201811182661.5 2018年10月11日 (11.10.2018) CN

(71) 申请人: 平安科技（深圳）有限公司 (PING AN TECHNOLOGY (SHENZHEN) CO., LTD.) [CN/CN]; 中国广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼, Guangdong 518033 (CN)。

(72) 发明人: 陈俊峰 (CHEN, Junfeng); 中国广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼, Guangdong 518033 (CN)。

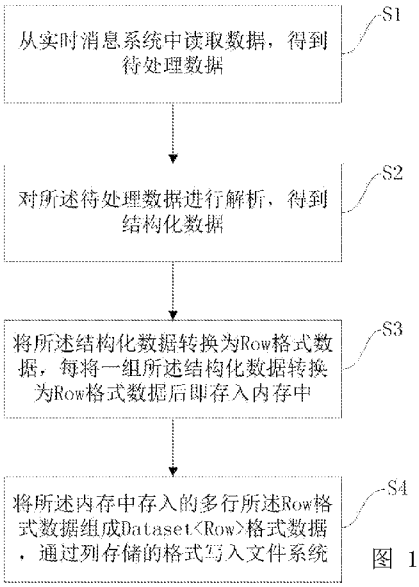
(74) 代理人: 北京市京大律师事务所 (BEIJING JINGDA LAW FIRM); 中国北京市海淀区海淀中街16号7层2单元703, Beijing 518033 (CN)。

(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, JO, JP, KE, KG, KH, 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,

WO 2020/073687 A1

(54) **Title:** COLUMNAR STORAGE METHOD AND APPARATUS FOR STREAMING DATA, DEVICE, AND STORAGE MEDIUM

(54) **发明名称:** 流式数据列存储方法、装置、设备和存储介质



S1 READ DATA FROM A REAL-TIME MESSAGE SYSTEM TO OBTAIN DATA TO BE PROCESSED
S2 PARSE THE DATA TO BE PROCESSED TO OBTAIN STRUCTURED DATA
S3 CONVERT THE STRUCTURED DATA INTO DATA IN ROW FORMAT, AND EVERY TIME A GROUP OF STRUCTURED DATA IS CONVERTED INTO DATA IN ROW FORMAT, STORE THE DATA IN ROW FORMAT INTO A MEMORY
S4 FORM MULTIPLE ROWS OF DATA IN ROW FORMAT STORED IN THE MEMORY INTO DATA IN DATASET<ROW> FORMAT, AND WRITE SAME TO A FILE SYSTEM IN A COLUMNAR STORAGE FORMAT

图 1

(57) **Abstract:** The present application relates to the field of streaming data storage, in particular, to a columnar storage method and apparatus for streaming data, a device, and a storage medium. The columnar storage method for streaming data comprises: reading data from a real-time message system to obtain data to be processed; parsing the data to be processed to obtain structured data; converting the structured data into data in Row format, and every time a group of structured data is converted into data in Row format, storing the data in Row format into a memory; and forming multiple rows of data in Row format stored in the memory into data in Dataset<Row> format, and writing same to a file system in a columnar storage format. According to the present application, streaming data in a real-time message system is processed by means of Spark Streaming, thereby solving the problem that streaming data in a real-time message system cannot be stored in a columnar storage format at present, greatly improving the speed of subsequent processing of a

[见续页]

SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

(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))。

large amount of data, and reducing the time for converting a row storage structure into a columnar storage structure.

(57) 摘要：本申请涉及流式数据存储领域，尤其涉及一种流式数据列存储方法、装置、设备和存储介质。流式数据列存储方法包括：从实时消息系统中读取数据，得到待处理数据；对所述待处理数据进行解析，得到结构化数据；将所述结构化数据转换为Row格式数据，每将一组所述结构化数据转换为Row格式数据后，即存入内存中；将所述内存中存入的多行所述Row格式数据组成Dataset<Row>格式数据，通过列存储的格式写入文件系统。本申请通过Spark Streaming对实时消息系统中的流式数据进行处理，解决了当前无法把实时消息系统中的流式数据保存为列存储格式的问题，极大地提高了后续对大量数据处理的速度，节省了把行存储结构转换为列存储结构的时间。

流式数据列存储方法、装置、设备和存储介质

本申请要求于2018年10月11日提交中国专利局、申请号为201811182661.5、发明名称为“流式数据列存储方法、装置、设备和存储介质”的中国专利申请的优先权，其全部内容通过引用结合在申请中。

技术领域

本申请涉及流式数据存储领域，尤其涉及一种流式数据列存储方法、装置、设备和存储介质。

背景技术

近年来，随着互联网的迅猛发展，数据的快速增长成了许多行业共同面临的机遇与挑战。在当今网络环境下，大量数据源是实时的，不间断的，要求对用户的响应时间也是实时的。这些数据以流式的形式被采集、计算与查询，如实时消息系统，对流入的数据均采取流式方式处理。其每时每刻都有各式各样的、海量的网络数据流入，流入速度各异，且数据结构复杂多样，包括二进制文件、文本文件、压缩文件等。对于此类系统，需要底层存储系统能够支持：对流入的数据以统一格式存储，对上层应用提供统一接口，方便检索，并且对实时性也有一定要求。

针对现今的大数据趋势，涌现了一批大数据处理平台，比如kafka，flume等。具体为前置应用把消息通过流式的方式输入到消息队列中，然后消息队列再通过某种形式把这些数据写入到磁盘，比如hdfs，或者本地磁盘。

发明人意识到由于实时消息系统的流式处理形式，使得消息最终都是以行存储的形式写入磁盘，比如json，或者普通文本。而在大数据处理中，很多情况下需要数据以列存储的形式进行保存，这时候传统的flume等工具就无法满足需求。

发明内容

有鉴于此，有必要针对现有实时消息系统中的数据均是以行存储的形式写入文件系统，而不是以列存储的形式写入文件系统，提供一种流式数据列存储方法、装置、设备和存储介质。

一种流式数据列存储方法，包括如下步骤：

从实时消息系统中读取数据，得到待处理数据；

对所述待处理数据进行解析，得到结构化数据；

将所述结构化数据转换为 Row 格式数据，每将一组所述结构化数据转换为 Row 格式数据后，即存入内存中；

将所述内存中存入的多行所述 Row 格式数据组成 Dataset<Row>格式数据，通过列存储的格式写入文件系统。

在其中一个实施例中，所述从实时消息系统中读取数据，得到待处理数据，包括：

获取所述实时消息系统的访问权限，并连接到所述实时消息系统；

设定执行周期，按照所述执行周期从所述实时消息系统中读取数据。

在其中一个实施例中，所述对所述待处理数据进行解析，得到结构化数据，包括对所述待处理数据的格式进行判断后，按照判断结果采用不同的方法进行解析，具体包括：

若所述待处理数据为 json 格式，则调用 FastJSON 将所述 json 格式的待处理数据解析为所述结构化数据；

若所述待处理数据为 csv 格式，则根据所述待处理数据的内容，并通过 DataFrame()方法给所述 csv 格式的待处理数据添加结构化信息，得到所述结构化数据。

在其中一个实施例中，所述将所述内存中存入的多行所述 Row 格式数据组成 Dataset<Row>格式数据，通过列存储的格式写入文件系统，包括：

通过数据框架的方法将多行所述 Row 格式数据组成所述 Dataset<Row>格式数据；

通过 parquet()将所述 Dataset<Row>格式数据转换为 parquet 格式数据，并使用 spark.read()将 parquet 格式数据写入文件系统。

在其中一个实施例中，所述设定执行周期，按照所述执行周期从所述实时消息系统中读取数据，包括：

从所述实时消息系统中第一条数据所在位置开始读取；

接收读取完毕的指令，停止读取，并记录读取完毕的位置；

获取上次读取完毕的位置，从上次读取完毕的位置开始读取，直到接收到读取完毕的指令，停止读取，并记录读取完毕的位置。

在其中一个实施例中，所述若所述待处理数据为 json 格式，则调用 FastJSON 将所述 json 格式的待处理数据解析为所述结构化数据，包括：

提取所述 json 格式的待处理数据的字段信息 ；

根据所述字段信息对所述 json 格式的待处理数据进行排序，得到所述结构化数据。

在其中一个实施例中，所述将所述内存中存入的多行所述 Row 格式数据组成 Dataset<Row>格式数据，通过列存储的格式写入文件系统之后，还包括：

根据所述待处理数据的列信息对存储路径进行分割；

调用 partitionBy()函数，将所述待处理数据中列名相同的列，按照所述列中不同的值存储于不同目录。

一种流式数据列存储装置，包括如下模块：

数据获取模块，设置为从实时消息系统中读取数据，得到待处理数据；

数据解析模块，设置为对所述待处理数据进行解析，得到结构化数据；

数据转换模块，设置为将所述结构化数据转换为 Row 格式数据，每将一组所述结构化数据转换为 Row 格式数据后，即存入内存中；

数据存储模块，设置为将所述内存中存入的多行所述 Row 格式数据组成 Dataset<Row>格式数据，通过列存储的格式写入文件系统。

一种计算机设备，包括存储器和处理器，所述存储器中存储有计算机可读指令，所述计算机可读指令被一个或多个所述处理器执行时，使得一个或多个所述处理器执行上述流式数据列存储方法的步骤。

一种存储有计算机可读指令的存储介质，所述计算机可读指令被一个或多个处理器执行时，使得一个或多个所述处理器执行上述流式数据列存储方法的步骤。

本申请通过 Spark Streaming 对实时消息系统中的流式数据进行处理，解决了当前无法把实时消息系统中的流式数据保存为列存储格式的问题，极大地提高了后续对大量数据处理的速度，也节省了把行存储结构转换为列存储结构的时间，使用 Spark Streaming 作为计算框架，极大地利用分布式计算提高了转换和存储性能。

附图说明

附图仅用于示出优选实施方式的目的，而并不认为是对本申请的限制。

图 1 为本申请的一种流式数据列存储方法的整体流程图；

图 2 为本申请的一种流式数据列存储方法中的数据获取过程的示意图；

图 3 为本申请的一种流式数据列存储方法中的数据解析过程的示意图；

图 4 为本申请的一种流式数据列存储方法中的数据存储过程的示意图；

图 5 为本申请的一种流式数据列存储装置的结构图。

具体实施方式

本技术领域技术人员可以理解，除非特意声明，这里使用的单数形式“一”、“一个”、“所述”和“该”也可包括复数形式。应该进一步理解的是，本申请的说明书中使用的措辞“包括”是指存在所述特征、整数、步骤、操作、元件和/或组件，但是并不排除存在或添加一个或多个其他特征、整数、步骤、操作、元件、组件和/或它们的组。

图 1 为本申请的一种流式数据列存储方法的整体流程图，如图 1 所示，一种流式数据列存储方法，包括以下步骤：

步骤 S1，从实时消息系统中读取数据，得到待处理数据。

其中，本申请的一种流式数据列存储装置主要包括 Spark Streaming 程序，本申请主要依托 Spark Streaming 对实时消息系统的流式数据进行处理，从而实现将实时消息系统中的流式数据转换为列存储的形式写入文件系统。实时消息系统中的数据为流式数据，实时消息系统也是流式数据的处理组件。

其中，Spark Streaming 包括数据获取模块、数据解析模块、数据转换模块和数据存储模块。

上述步骤执行时，数据获取模块的其中一个子模块每隔一段时间发出数据获取指令，另一个子模块则接收上述数据获取指令，接收到数据获取指令后，执行指令，从实时消息系统中读取数据，得到待处理数据。

步骤 S2，对所述待处理数据进行解析，得到结构化数据。

上述步骤执行时，数据获取模块将得到的待处理数据发送给数据解析模块，数据解析模块对待处理数据进行解析，并根据待处理数据的不同格式采用不同的方法进行解析。从实时消息系统中获取的数据，其数据结构复杂多样，包括二进制文件、文本文件、压缩文件等各种格式的数据。数据解析模块接收到不同格式的待处理数据后，采用不同方法进行解析，最终将待处理数据统一解析为结构化数据，再发送给数据转换模块。

步骤 S3，将所述结构化数据转换为 Row 格式数据，每将一组所述结构化数据转换为 Row 格式数据后，即存入内存中。

上述步骤执行时，数据转换模块将数据解析模块发来的结构化数据转换为Row格式数据，暂时存入数据存储模块中。

在其中一个优选的实施例中，通过spark.createRow()将解析后的结构化数据转换为Row格式数据。

其中，Row格式是Spark Streaming自带的一种格式，并且Row格式为一种带列信息的数据结构，其实质为一行数据。

步骤S4，将所述内存中存入的多行所述Row格式数据组成Dataset<Row>格式数据，通过列存储的格式写入文件系统。

上述步骤执行时，经过数据转换模块转换好的Row格式数据暂时存入数据存储模块，每隔一段时间，把累加的多行Row格式数据组成Dataset<Row>格式数据，一次性以列存储的格式写入文件系统。

其中，Dataset<Row>格式是Spark Streaming自带的一种格式，Dataset<Row>格式是很多行Row格式组成的矩阵，是Row格式数据的有序集合，Dataset<Row>即为列信息结构，将Dataset<Row>格式的数据转换为列存储的格式即是数据以列的形式存在。

本实施例，通过Spark Streaming对实时消息系统中的流式数据进行解析处理，将每一条数据转换为Spark Streaming中的Row格式数据，并将多行Row格式数据合并累加暂时放到数据存储模块中，再组成Dataset<Row>格式一次性写入文件系统，解决了当前流式数据无法列存储的问题，使用Spark Streaming作为计算框架，提高了数据的转换和存储性能。

在一个实施例中，图2为本申请的一种流式数据列存储方法中的数据获取过程的示意图，如图2所示，一种流式数据列存储方法的数据获取过程，包括如下步骤：

步骤S101，获取所述实时消息系统的访问权限，并连接到所述实时消息系统。

上述步骤执行时，通过使用远程连接权限的用户名和密码获取所述实时消息系统的访问权限，并通过Hibernate对象关系映射框架与所述实时消息系统进行连接。

步骤S102，设定执行周期，按照所述执行周期从所述实时消息系统中读取数据。

上述步骤执行时，设定 Spark Streaming 程序的执行周期，并将执行周期作为参数值传入 Spark Streaming 程序。

在其中一个优选的实施例中，也可以将执行周期作为固定值写在程序中，设定在 Spark Streaming 程序的配置参数中。

在其中一个优选的实施例中，执行周期可以设置为每次读取时间间隔相同，也可以根据实时消息系统数据流入速度的快慢设置为读取时间间隔不相同。

本实施例，将执行周期作为参数值传入 Spark Streaming 程序中，比较灵活，将执行周期作为固定值写在程序中，可以确保数值的安全性较高，而且执行周期可以根据实时消息系统数据流入速度的快慢灵活设置。

在一个实施例中，图 3 为本申请的一种流式数据列存储方法中的数据解析过程的示意图，如图 3 所示，一种流式数据列存储方法的数据解析过程，包括如下步骤：

步骤 S201，若所述待处理数据为 json 格式，则调用 FastJSON 将所述 json 格式的待处理数据解析为所述结构化数据。

上述步骤执行时，若从实时消息系统中获取的数据为 json 格式，使用相关库进行对其进行解析，在其中一个优选的实施例中，使用 FastJSON 对其进行解析。

具体的，若从实时消息系统中获取的数据为{"id"：0，"name"："Alice"，"age"：21}的 json 格式数据，其结构包括 3 个字段，分别为 id、name 和 age，分别代表 id、姓名和年龄。使用 Fastjson 对其进行解析之后，则会解析为一个包含 id，name，age 的结构化数据，之后，再将解析之后的数据转换为 Spark Streaming 自带的 Row 格式数据。

步骤 S202，若所述待处理数据为 csv 格式，则根据所述待处理数据的内容，并通过 DataFrame()方法给所述 csv 格式的待处理数据添加结构化信息，得到所述结构化数据。

其中，不同于 json 和 avro 等格式的数据，csv 格式的数据一般只包含数据信息，不包含结构信息。如上述步骤 S201 中提到的{"id"：0，"name"："Alice"，"age"：21}的 json 格式数据，如果是 csv 格式，则其数据内容只有 0，Alice，21。这样格式的数据，无法通过数据内容确定每一列所表示的意思，需要根据用户对数据的认知，设定第一列为 id，第二列为姓名，第三列为年龄，即根据

数据的内容自行添加结构化信息，将数据解析为结构化数据，再将数据转换为Row格式数据。

上述步骤执行时，通过 spark.createDataFrame(RowJavaRDD,type)方法来给数据添加结构化信息。其中，RowJavaRDD指的是数据信息，type为结构信息。

本实施例，对不同格式的数据采用不同的解析方法，使数据统一解析为结构化数据，再将数据转换为 Row 格式数据，节省了数据处理的时间，且提高了数据处理的准确度。

在一个实施例中，图 4 为本申请的一种流式数据列存储方法中的数据存储过程的示意图，如图 4 所示，一种流式数据列存储方法的数据存储过程，包括如下步骤：

步骤 S301，通过数据框架的方法将多行所述 Row 格式数据组成所述Dataset<Row>格式数据。

在其中一个优选的实施例中，使用 spark.createDataFrame(RowJavaRDD,type) 将多行 Row 格式数据组成Dataset<Row>格式数据，其中，RowJavaRDD表示数据信息，type表示结构信息。

步骤 S302，通过 parquet()将所述 Dataset<Row>格式数据转换为 parquet格式数据，并使用 spark.read()将 parquet 格式数据写入文件系统。

上述步骤执行时，使用 spark.read().parquet(filename)将 Dataset<Row>格式数据以 parquet 格式，写入文件系统。

上述步骤执行时，如果要以 parquet 格式进行列存储，使用 parquet()将Dataset<Row>格式数据转换为parquet格式，具体的，使用parquet(filename)将 Dataset<Row>格式数据转换为 parquet 格式，parquet 是一种支持列式存储的文件格式。

上述步骤执行时，使用 spark.read()将转换后的 parquet 格式的数据，写入文件系统。

本步骤中，还可以通过其他列存储格式将数据写入文件系统。

文件系统包括本地文件（file://）和 HDFS(hdfs://)，亦可包括其他 spark所支持的其他文件系统，比如亚马逊 S3（s3://）。一般是通过文件名来制定，比如要写入 hdfs 根目录下的 data 文件夹，则可以设置为 hdfs:///data/。

本实施例，通过使用 spark.createDataFrame(RowJavaRDD,type)将多行 Row格式数据组成 Dataset<Row>格式数据，使流式数据转换为列数据结构，为后续

数据列存储打好基础。使用 spark.read()将组成的 Dataset<Row>格式的数据写入文件系统，实现了流式数据以列存储的格式写入文件系统。

在一个实施例中，按照所述执行周期从所述实时消息系统中读取数据，包括如下具体步骤：

从所述实时消息系统中第一条数据所在位置开始读取。

接收读取完毕的指令，停止读取，并记录读取完毕的位置。

获取上次读取完毕的位置，从上次读取完毕的位置开始读取，直到接收到读取完毕的指令，停止读取，并记录读取完毕的位置。

上述步骤执行时，当程序为首次读取数据时，则从实时消息系统中第一条数据所在位置开始读取，直到读取时产生的最新数据读取完，此时，会接收到读取完毕的指令，则停止读取，Spark Streaming 自动记录下读取完毕的位置。

其中，首次读取数据是指第一次启动程序时的读取，Spark Streaming 程序为永久运行，如果不暂停，可以一直运行下去。由于实时消息系统的数据是源源不断的被写入的，所以每次数据读取完毕时，由 Spark Streaming 记录下每次读取完毕的位置，以便下次读取。

以后每次读取数据时，获取上次读取完毕的位置，从上次读取完毕的位置开始读取，直到接收到读取完毕的指令，停止读取，并记录读取完毕的位置。

本实施例，每次读取完毕，都会记录下读取完毕的位置，便于下次读取，且不易出错，提高了数据获取的速度和质量。

在一个实施例中，调用 FastJSON 将所述 json 格式的待处理数据解析为所述结构化数据，包括如下具体步骤：

提取所述 json 格式的待处理数据的字段信息；

根据所述字段信息对所述 json 格式的待处理数据进行排序，得到所述结构化数据。

从实时消息系统中获取的数据为{"age"：21，"id"：0，"name"："Alice"， }的 json 格式数据，使用 FastJSON 将数据的字段信息提取出来，分别为 age、id、和 name，分别代表年龄、id 和姓名。再根据字段信息对待处理数据进行排序，比如，排好序的数据结构为{"id"，"name"，"age"}，则{"id"，"name"，"age"}即为结构化数据。

在一个实施例中，数据转换模块根据需要决定是否对解析后的结构化数据进行修改。若存储时需要按照年月日进行存储，若从实时消息系统中获取的数

据含有时间戳，如“2017-09-21 08:16:05.011”，而存储时需要按照年月日进行存储，则需要把时间戳中的年月日信息提取出来。

在一个实施例中，可以根据待处理数据的列信息对存储路径进行分割，通过 partitionBy()函数，将所述待处理数据中列名相同的列，按照所述列中不同的值存储于不同目录。上述步骤执行时，通过 spark.read().partitionBy()对存储路径进行分割，比如参数填写为 newDf.write().mode(SaveMode.Append).partitionBy("stream", "year", "month", "day", "hour").orc("orc")，指的是根据 stream，year，month，day 字段来进行路径的分割。

partitionBy 是分析性函数的一部分，它和聚合函数 groupBy 不同的地方在于它能返回一个分组中的多条记录，而聚合函数一般只有一条反映统计值的记录，partitionBy 用于给结果集分组，如果没有指定那么它把整个结果集作为一个分组，partitionBy 返回的是分组里的每一条数据，并且可以对分组数据进行排序操作。

本实施例，通过使用 partitionBy 函数实现了对存储路径的分割，方便了后续对大量数据的处理。

一种流式数据列存储装置，如图 5 所示，包括如下模块：

数据获取模块，设置为从实时消息系统中读取数据，得到待处理数据；

数据解析模块，设置为对所述待处理数据进行解析，得到结构化数据；

数据转换模块，设置为将所述结构化数据转换为 Row 格式数据，每将一组所述结构化数据转换为 Row 格式数据后，即存入内存中；

数据存储模块，设置为将所述内存中存入的多行所述 Row 格式数据组成 Dataset<Row>格式数据，通过列存储的格式写入文件系统。

在一个实施例中，提出了一种计算机设备，包括存储器和处理器，存储器中存储有计算机可读指令，计算机可读指令被一个或多个处理器执行时，使得一个或多个处理器执行计算机可读指令时实现上述各实施例中所述的流式数据列存储方法的步骤。

在一个实施例中，提出了一种存储有计算机可读指令的存储介质，计算机可读指令被一个或多个处理器执行时，使得一个或多个处理器执行上述各实施例中所述的流式数据列存储方法的步骤。其中，所述存储介质可以为非易失性存储介质。

本领域普通技术人员可以理解上述实施例的各种方法中的全部或部分步骤是可以通过程序来指令相关的硬件来完成，该程序可以存储于一计算机可读存储介质中，存储介质可以包括：只读存储器（ROM，Read Only Memory）、随机存取存储器（RAM，Random Access Memory）、磁盘或光盘等。

以上所述实施例的各技术特征可以进行任意的组合，为使描述简洁，未对上述实施例中的各个技术特征所有可能的组合都进行描述，然而，只要这些技术特征的组合不存在矛盾，都应当认为是本说明书记载的范围。

以上所述实施例仅表达了本申请一些示例性实施例，其描述较为具体和详细，但并不能因此而理解为对本申请专利范围的限制。应当指出的是，对于本领域的普通技术人员来说，在不脱离本申请构思的前提下，还可以做出若干变形和改进，这些都属于本申请的保护范围。因此，本申请专利的保护范围应以所附权利要求为准。

权 利 要 求 书

1、一种流式数据列存储方法，包括如下步骤：

从实时消息系统中读取数据，得到待处理数据；

对所述待处理数据进行解析，得到结构化数据；

将所述结构化数据转换为 Row 格式数据，每将一组所述结构化数据转换为 Row 格式数据后，即存入内存中；

将所述内存中存入的多行所述 Row 格式数据组成 Dataset<Row>格式数据，通过列存储的格式写入文件系统。

2、根据权利要求 1 所述的流式数据列存储方法，所述从实时消息系统中读取数据，得到待处理数据，包括：

获取所述实时消息系统的访问权限，并连接到所述实时消息系统；

设定执行周期，按照所述执行周期从所述实时消息系统中读取数据。

3、根据权利要求 1 所述的流式数据列存储方法，所述对所述待处理数据进行解析，得到结构化数据，包括对所述待处理数据的格式进行判断后，按照判断结果采用不同的方法进行解析，具体包括：

若所述待处理数据为 json 格式，则调用 FastJSON 将所述 json 格式的待处理数据解析为所述结构化数据；

若所述待处理数据为 csv 格式，则根据所述待处理数据的内容，并通过 DataFrame()方法给所述 csv 格式的待处理数据添加结构化信息，得到所述结构化数据。

4、根据权利要求 1 所述的流式数据列存储方法，所述将所述内存中存入的多行所述 Row 格式数据组成 Dataset<Row>格式数据，通过列存储的格式写入文件系统，包括：

通过数据框架的方法将多行所述 Row 格式数据组成所述 Dataset<Row>格式数据；

通过 parquet()将所述 Dataset<Row>格式数据转换为 parquet 格式数据，并使用 spark.read()将 parquet 格式数据写入文件系统。

5、根据权利要求 2 所述的流式数据列存储方法，所述设定执行周期，按照所述执行周期从所述实时消息系统中读取数据，包括：

从所述实时消息系统中第一条数据所在位置开始读取；

接收读取完毕的指令，停止读取，并记录读取完毕的位置；

获取上次读取完毕的位置，从上次读取完毕的位置开始读取，直到接收到读取完毕的指令，停止读取，并记录读取完毕的位置。

6、根据权利要求 3 所述的流式数据列存储方法，所述若所述待处理数据为 json 格式，则调用 FastJSON 将所述 json 格式的待处理数据解析为所述结构化数据，包括：

提取所述 json 格式的待处理数据的字段信息；

根据所述字段信息对所述 json 格式的待处理数据进行排序，得到所述结构化数据。

7、根据权利要求 1 所述的流式数据列存储方法，所述将所述内存中存入的多行所述 Row 格式数据组成 Dataset<Row>格式数据，通过列存储的格式写入文件系统之后，还包括：

根据所述待处理数据的列信息对存储路径进行分割；

调用 partitionBy()函数，将所述待处理数据中列名相同的列，按照所述列中不同的值存储于不同目录。

8、一种流式数据列存储装置，包括如下模块：

数据获取模块，设置为从实时消息系统中读取数据，得到待处理数据；

数据解析模块，设置为对所述待处理数据进行解析，得到结构化数据；

数据转换模块，设置为将所述结构化数据转换为 Row 格式数据，每将一组所述结构化数据转换为 Row 格式数据后，即存入内存中；

数据存储模块，设置为将所述内存中存入的多行所述 Row 格式数据组成 Dataset<Row>格式数据，通过列存储的格式写入文件系统。

9、一种计算机设备，包括存储器和处理器，所述存储器中存储有计算机可读指令，所述计算机可读指令被一个或多个所述处理器执行时，使得一个或多

个所述处理器执行如下步骤：

从实时消息系统中读取数据，得到待处理数据；

对所述待处理数据进行解析，得到结构化数据；

将所述结构化数据转换为 Row 格式数据，每将一组所述结构化数据转换为 Row 格式数据后，即存入内存中；

将所述内存中存入的多行所述 Row 格式数据组成 Dataset<Row>格式数据，通过列存储的格式写入文件系统。

10、如权利要求 9 所述的计算机设备，所述计算机可读指令被一个或多个所述处理器执行时，使得一个或多个所述处理器还执行如下步骤：

获取所述实时消息系统的访问权限，并连接到所述实时消息系统；

设定执行周期，按照所述执行周期从所述实时消息系统中读取数据。

11、如权利要求 9 所述的计算机设备，所述计算机可读指令被一个或多个所述处理器执行时，使得一个或多个所述处理器还执行如下步骤：

若所述待处理数据为 json 格式，则调用 FastJSON 将所述 json 格式的待处理数据解析为所述结构化数据；

若所述待处理数据为 csv 格式，则根据所述待处理数据的内容，并通过 DataFrame()方法给所述 csv 格式的待处理数据添加结构化信息，得到所述结构化数据。

12、如权利要求 9 所述的计算机设备，所述计算机可读指令被一个或多个所述处理器执行时，使得一个或多个所述处理器还执行如下步骤：

通过数据框架的方法将多行所述 Row 格式数据组成所述 Dataset<Row>格式数据；

通过 parquet()将所述 Dataset<Row>格式数据转换为 parquet 格式数据，并使用 spark.read()将 parquet 格式数据写入文件系统。

13、如权利要求 10 所述的计算机设备，所述计算机可读指令被一个或多个所述处理器执行时，使得一个或多个所述处理器还执行如下步骤：

从所述实时消息系统中第一条数据所在位置开始读取；

接收读取完毕的指令，停止读取，并记录读取完毕的位置；

获取上次读取完毕的位置，从上次读取完毕的位置开始读取，直到接收到读取完毕的指令，停止读取，并记录读取完毕的位置。

14、如权利要求 11 所述的计算机设备，所述计算机可读指令被一个或多个所述处理器执行时，使得一个或多个所述处理器还执行如下步骤：

提取所述 json 格式的待处理数据的字段信息；

根据所述字段信息对所述 json 格式的待处理数据进行排序，得到所述结构化数据。

15、如权利要求 9 所述的计算机设备，所述计算机可读指令被一个或多个所述处理器执行时，使得一个或多个所述处理器还执行如下步骤：

根据所述待处理数据的列信息对存储路径进行分割；

调用 partitionBy()函数，将所述待处理数据中列名相同的列，按照所述列中不同的值存储于不同目录。

16、一种存储有计算机可读指令的存储介质，所述计算机可读指令被一个或多个处理器执行时，使得一个或多个所述处理器执行如下步骤：

从实时消息系统中读取数据，得到待处理数据；

对所述待处理数据进行解析，得到结构化数据；

将所述结构化数据转换为 Row 格式数据，每将一组所述结构化数据转换为 Row 格式数据后，即存入内存中；

将所述内存中存入的多行所述 Row 格式数据组成 Dataset<Row>格式数据，通过列存储的格式写入文件系统。

17、如权利要求 16 所述的存储介质，所述计算机可读指令被一个或多个处理器执行时，使得一个或多个所述处理器执行还如下步骤：

获取所述实时消息系统的访问权限，并连接到所述实时消息系统；

设定执行周期，按照所述执行周期从所述实时消息系统中读取数据。

18、如权利要求 16 所述的存储介质，所述计算机可读指令被一个或多个处

理器执行时，使得一个或多个所述处理器执行还如下步骤：

若所述待处理数据为 json 格式，则调用 FastJSON 将所述 json 格式的待处理数据解析为所述结构化数据；

若所述待处理数据为 csv 格式，则根据所述待处理数据的内容，并通过 DataFrame()方法给所述 csv 格式的待处理数据添加结构化信息，得到所述结构化数据。

19、如权利要求 16 所述的存储介质，所述计算机可读指令被一个或多个处理器执行时，使得一个或多个所述处理器执行还如下步骤：

通过数据框架的方法将多行所述 Row 格式数据组成所述 Dataset<Row>格式数据；

通过 parquet()将所述 Dataset<Row>格式数据转换为 parquet 格式数据，并使用 spark.read()将 parquet 格式数据写入文件系统。

20、如权利要求 17 所述的存储介质，所述计算机可读指令被一个或多个处理器执行时，使得一个或多个所述处理器执行还如下步骤：

从所述实时消息系统中第一条数据所在位置开始读取；

接收读取完毕的指令，停止读取，并记录读取完毕的位置；

获取上次读取完毕的位置，从上次读取完毕的位置开始读取，直到接收到读取完毕的指令，停止读取，并记录读取完毕的位置。

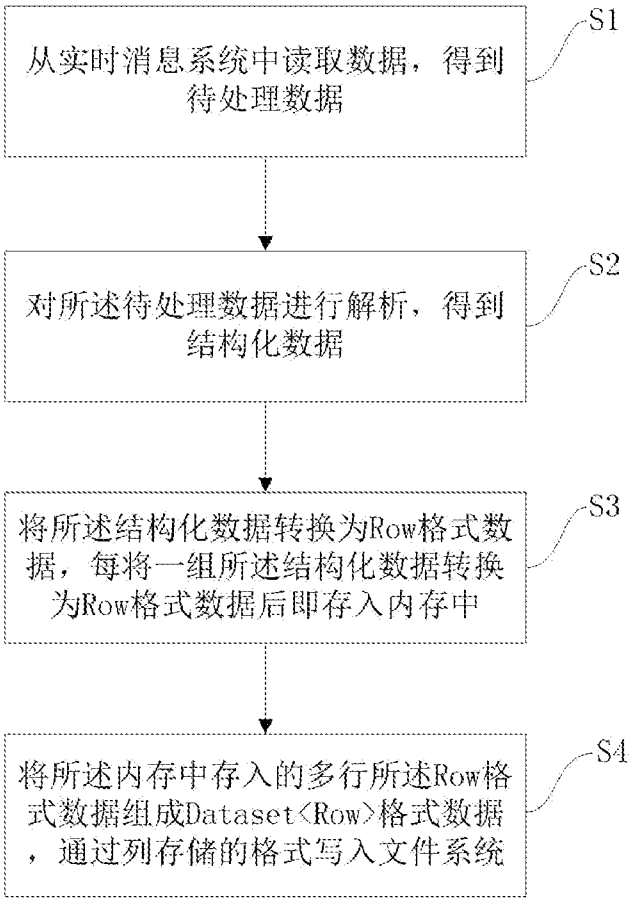

从实时消息系统中读取数据，得到待处理数据
S1
对所述待处理数据进行解析，得到结构化数据
S2
将所述结构化数据转换为Row格式数据，每将一组所述结构化数据转换为Row格式数据后即存入内存中
S3
将所述内存中存入的多行所述Row格式数据组成Dataset<Row>格式数据，通过列存储的格式写入文件系统
S4

图 1

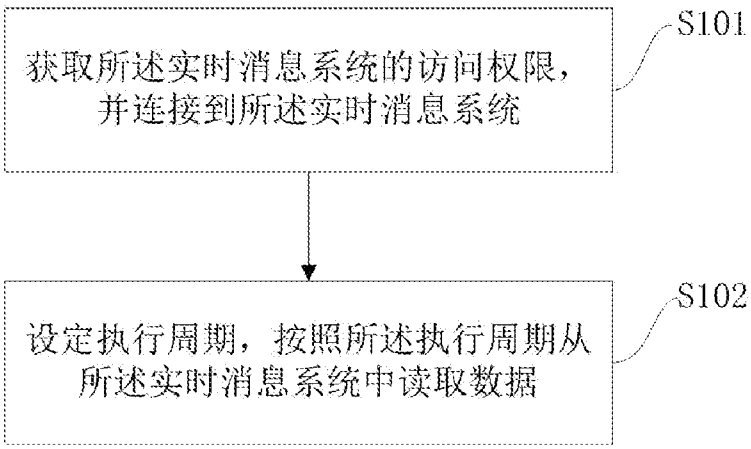

获取所述实时消息系统的访问权限，并连接到所述实时消息系统
S101
设定执行周期，按照所述执行周期从所述实时消息系统中读取数据
S102

图 2

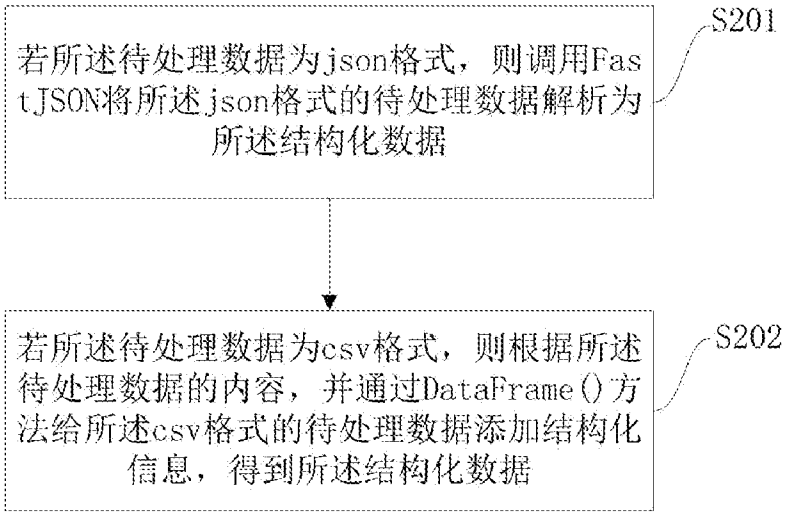

若所述待处理数据为json格式，则调用FastJSON将所述json格式的待处理数据解析为所述结构化数据
S201
若所述待处理数据为csv格式，则根据所述待处理数据的内容，并通过DataFrame()方法给所述csv格式的待处理数据添加结构化信息，得到所述结构化数据
S202

图 3

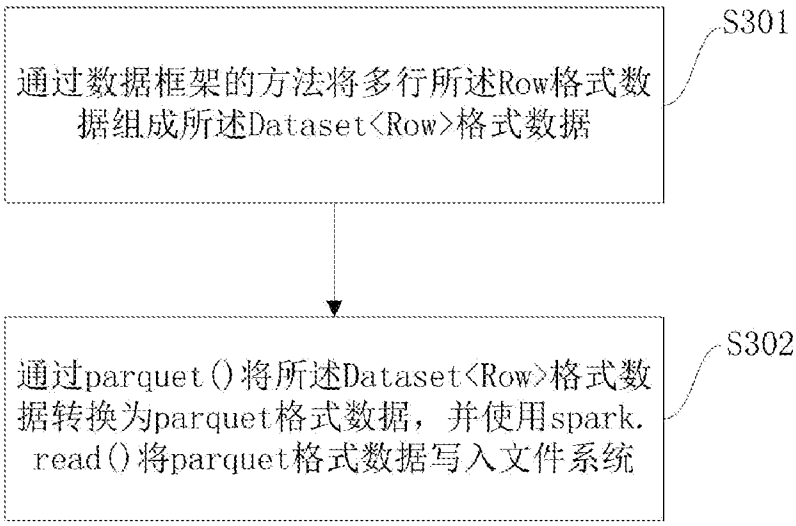

通过数据框架的方法将多行所述Row格式数据组成所述Dataset<Row>格式数据
S301
通过parquet()将所述Dataset<Row>格式数据转换为parquet格式数据，并使用spark.read()将parquet格式数据写入文件系统
S302

图 4

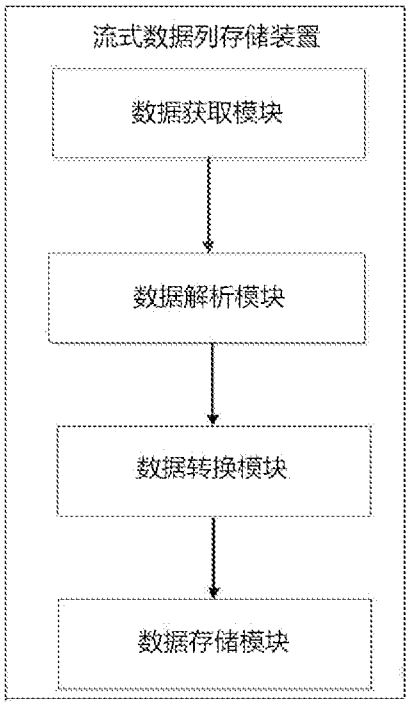


图 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2019/092893

A. CLASSIFICATION OF SUBJECT MATTER

G06F 16/22(2019.01)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

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)

VEN; CNABS; CNTXT; CNKI: 流式数据, 列存储, 计算框架, 解析, 结构化, streaming, store, stor+, column, data, spark, row

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN 109542889 A (PING AN TECHNOLOGY (SHENZHEN) CO., LTD.) 29 March 2019 (2019-03-29) entire document	1-20
A	CN 108255855 A (BEIJING GRIDSUM TECHNOLOGY CO., LTD.) 06 July 2018 (2018-07-06) entire document	1-20
A	US 2009171999 A1 (CLOUDSCALE INC.) 02 July 2009 (2009-07-02) entire document	1-20

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"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)

"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

"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

"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

"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

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
26 August 2019	**10 September 2019**

Name and mailing address of the ISA/CN	Authorized officer
China National Intellectual Property Administration (ISA/CN) **No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088** **China**	
Facsimile No. **(86-10)62019451**	Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2019/092893

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN 109542889 A	29 March 2019	None	
CN 108255855 A	06 July 2018	None	
US 2009171999 A1	02 July 2009	US 8069190 B2	29 November 2011

国际检索报告

国际申请号
PCT/CN2019/092893

A. 主题的分类

G06F 16/22(2019.01)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)

G06F

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）)

VEN;CNABS;CNTXT;CNKI:流式数据，列存储，计算框架，解析，结构化，streaming，store，stor+，column，data，spark，row

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
PX	CN 109542889 A (平安科技深圳有限公司) 2019年 3月 29日 (2019-03-29) 全文	1-20
A	CN 108255855 A (北京国双科技有限公司) 2018年 7月 6日 (2018-07-06) 全文	1-20
A	US 2009171999 A1 (CLOUDSCALE INC) 2009年 7月 2日 (2009-07-02) 全文	1-20

☐ 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	国际检索报告邮寄日期
2019年 8月 26日	2019年 9月 10日

ISA/CN的名称和邮寄地址	受权官员
中国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088	白雪涛
传真号 (86-10)62019451	电话号码 62412069

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2019/092893

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 109542889 A	2019年 3月 29日	无	
CN 108255855 A	2018年 7月 6日	无	
US 2009171999 A1	2009年 7月 2日	US 8069190 B2	2011年 11月 29日