(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 111797297 B
(45)授权公告日 2020.12.15

(21)申请号 202010937717.4

(22)申请日 2020.09.09

(65)同一申请的已公布的文献号
申请公布号 CN 111797297 A

(43)申请公布日 2020.10.20

(73)专利权人 平安国际智慧城市科技股份有限公司
地址 518000 广东省深圳市前海深港合作区妈湾兴海大道3048号前海自贸大厦1-34层

(72)发明人 贾波涛

(74)专利代理机构 深圳市世联合知识产权代理有限公司 44385
代理人 汪琳琳

(51)Int.Cl.
G06F 16/951(2019.01)
H04L 29/08(2006.01)
G06F 16/25(2019.01)

(56)对比文件
CN 107733986 A,2018.02.23
CN 108133041 A,2018.06.08

审查员 杨婷

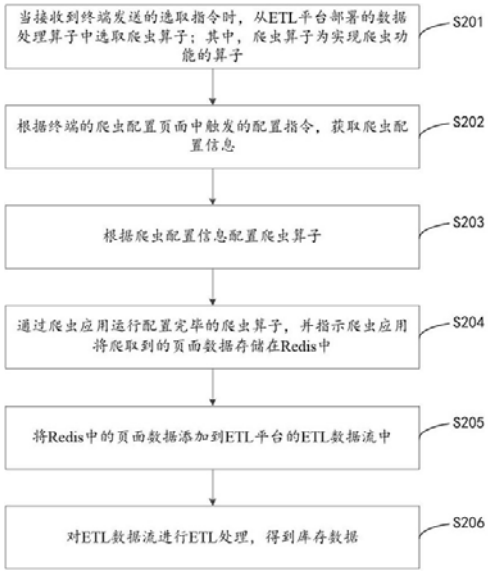
权利要求书2页 说明书12页 附图4页

(54)发明名称

页面数据处理方法、装置、计算机设备及存储介质

(57)摘要

本申请实施例属于大数据领域，应用于智慧城市领域中，涉及一种页面数据处理方法，包括：当接收到终端发送的选取指令时，从ETL平台部署的数据处理算子中选取爬虫算子；根据所述终端的爬虫配置页面中触发的配置指令，获取爬虫配置信息；根据所述爬虫配置信息配置所述爬虫算子；通过爬虫应用运行配置完毕的爬虫算子，并指示所述爬虫应用将爬取到的页面数据存储在Redis中；将所述Redis中的所述页面数据添加到所述ETL平台的ETL数据流中；对所述ETL数据流进行ETL处理，得到库存数据。本申请还提供一种页面数据处理装置、计算机设备及存储介质。此外，本申请还涉及区块链技术，库存数据可存储于区块链中。本申请提高了对页面数据的处理效率。



CN 111797297 B

1.一种页面数据处理方法,其特征在于,包括:

当接收到终端发送的选取指令时,读取ETL平台的状态标识;

当通过所述状态标识确定所述ETL平台未处于数据输出状态时,从所述ETL平台部署的数据处理算子中选取爬虫算子,并通过所述终端展示所述爬虫算子的爬虫配置页面;其中,所述爬虫算子为实现爬虫功能的算子;

根据所述终端的爬虫配置页面中触发的配置指令,获取爬虫配置信息;

根据所述爬虫配置信息配置所述爬虫算子;

通过爬虫应用运行配置完毕的爬虫算子,并指示所述爬虫应用将爬取到的页面数据存储在Redis中;

将所述Redis中的所述页面数据添加到所述ETL平台的ETL数据流中;

对所述ETL数据流进行ETL处理,得到库存数据。

2.根据权利要求1所述的页面数据处理方法,其特征在于,所述根据所述终端的爬虫配置页面中触发的配置指令,获取爬虫配置信息包括:

通过所述终端获取所述爬虫配置页面中的确认选项及文本框文本;

接收所述终端根据获取到的确认选项及文本框文本触发的配置指令;

根据所述配置指令获取爬虫配置信息。

3.根据权利要求1所述的页面数据处理方法,其特征在于,所述根据所述终端的爬虫配置页面中触发的配置指令,获取爬虫配置信息包括:

当接收到终端发送的流展示指令时,通过所述终端的爬虫配置页面展示所述ETL平台中的ETL数据流;

接收在展示的ETL数据流中选中待爬取字段触发的配置指令;

将所述配置指令中的待爬取字段添加为爬虫配置信息。

4.根据权利要求1所述的页面数据处理方法,其特征在于,所述根据所述终端的爬虫配置页面中触发的配置指令,获取爬虫配置信息,还包括:

获取所述终端的爬虫配置页面中触发的配置指令中所包含的URL;

将所述URL添加为爬虫配置信息;

或者,

当所述终端的爬虫配置页面中触发的配置指令中包含流获取指令时,从所述ETL平台的ETL数据流中查询URL标识;

读取所述URL标识所对应的ETL数据流作为爬虫配置信息。

5.根据权利要求1所述的页面数据处理方法,其特征在于,所述将所述Redis中的所述页面数据添加到所述ETL平台的ETL数据流中包括:

监测所述Redis与所述爬虫算子中的关键字;

当监测到所述Redis与所述爬虫算子中存在相同的关键字时,将所述Redis中所述关键字对应的页面数据添加到所述ETL平台的ETL数据流中。

6.根据权利要求1所述的页面数据处理方法,其特征在于,所述对所述ETL数据流进行ETL处理,得到库存数据包括:

从所述终端获取ETL设置信息;

根据所述ETL设置信息选取处理引擎对所述ETL数据流进行ETL处理;

将ETL处理后的ETL数据流进行存储,得到库存数据。

7.一种页面数据处理装置,其特征在于,包括:

算子选取模块,用于当接收到终端发送的选取指令时,读取ETL平台的状态标识;当通过所述状态标识确定所述ETL平台未处于数据输出状态时,从所述ETL平台部署的数据处理算子中选取爬虫算子,并通过所述终端展示所述爬虫算子的爬虫配置页面;其中,所述爬虫算子为实现爬虫功能的算子;

信息获取模块,用于根据所述终端的爬虫配置页面中触发的配置指令,获取爬虫配置信息;

算子配置模块,用于根据所述爬虫配置信息配置所述爬虫算子;

算子运行模块,用于通过爬虫应用运行配置完毕的爬虫算子,并指示所述爬虫应用将爬取到的页面数据存储在Redis中;

数据添加模块,用于将所述Redis中的所述页面数据添加到所述ETL平台的ETL数据流中;

数据处理模块,用于对所述ETL数据流进行ETL处理,得到库存数据。

8.一种计算机设备,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,所述处理器执行所述计算机可读指令时实现如权利要求1至6中任一项所述的页面数据处理方法。

9.一种计算机可读存储介质,其特征在于,所述计算机可读存储介质上存储有计算机可读指令,所述计算机可读指令被处理器执行时实现如权利要求1至6中任一项所述的页面数据处理方法。

页面数据处理方法、装置、计算机设备及存储介质

技术领域

[0001] 本申请涉及大数据领域，尤其涉及一种页面数据处理方法、装置、计算机设备及存储介质。

背景技术

[0002] 随着大数据技术的发展，ETL的应用也越来越广泛。ETL(Extract-Transform-Load)是将数据从来源端经过抽取(extract)、转换(transform)、加载(load)至目的端的过程。ETL的数据来源端通常是各种业务系统，目的端通常为数据仓库，但也不局限于数据仓库。ETL目的是将各种分散、零乱、标准不统一的数据整合到一起，为决策提供分析依据，ETL在商业智能中有着重要的应用。

[0003] 然而，传统的ETL工具只能从数据库或者指定的文件中获取数据，对于大量的没有存储在数据库或文件中的数据，例如页面数据等不能直接处理，使得ETL工具的数据处理效率较低。

发明内容

[0004] 本申请实施例的目的在于提出一种页面数据处理方法、装置、计算机设备及存储介质，以解决传统的ETL工具对页面数据处理效率较低的问题。

[0005] 为了解决上述技术问题，本申请实施例提供一种页面数据处理方法，采用了如下所述的技术方案：

[0006] 当接收到终端发送的选取指令时，从ETL平台部署的数据处理算子中选取爬虫算子；其中，所述爬虫算子为实现爬虫功能的算子；

[0007] 根据所述终端的爬虫配置页面中触发的配置指令，获取爬虫配置信息；

[0008] 根据所述爬虫配置信息配置所述爬虫算子；

[0009] 通过爬虫应用运行配置完毕的爬虫算子，并指示所述爬虫应用将爬取到的页面数据存储在Redis中；

[0010] 将所述Redis中的所述页面数据添加到所述ETL平台的ETL数据流中；

[0011] 对所述ETL数据流进行ETL处理，得到库存数据。

[0012] 进一步的，所述当接收到终端发送的选取指令时，从ETL平台部署的数据处理算子中选取爬虫算子包括：

[0013] 当接收到终端发送的选取指令时，读取ETL平台的状态标识；

[0014] 当通过所述状态标识确定所述ETL平台未处于数据输出状态时，从所述ETL平台部署的数据处理算子中选取爬虫算子，并通过所述终端展示所述爬虫算子的爬虫配置页面。

[0015] 进一步的，所述根据所述终端的爬虫配置页面中触发的配置指令，获取爬虫配置信息包括：

[0016] 通过所述终端获取所述爬虫配置页面中的确认选项及文本框文本；

[0017] 接收所述终端根据获取到的确认选项及文本框文本触发的配置指令；

[0018] 根据所述配置指令获取爬虫配置信息。

[0019] 进一步的,所述根据所述终端的爬虫配置页面中触发的配置指令,获取爬虫配置信息包括:

[0020] 当接收到终端发送的流展示指令时,通过所述终端的爬虫配置页面展示所述ETL平台中的ETL数据流;

[0021] 接收在展示的ETL数据流中选中待爬取字段触发的配置指令;

[0022] 将所述配置指令中的待爬取字段添加为爬虫配置信息。

[0023] 进一步的,所述根据所述终端的爬虫配置页面中触发的配置指令,获取爬虫配置信息,还包括:

[0024] 获取所述终端的爬虫配置页面中触发的配置指令中所包含的URL;

[0025] 将所述URL添加为爬虫配置信息;

[0026] 或者,

[0027] 当所述终端的爬虫配置页面中触发的配置指令中包含流获取指令时,从所述ETL平台的ETL数据流中查询URL标识;

[0028] 读取所述URL标识所对应的ETL数据流作为爬虫配置信息。

[0029] 进一步的,所述将所述Redis中的所述页面数据添加到所述ETL平台的ETL数据流中包括:

[0030] 监测所述Redis与所述爬虫算子中的关键字;

[0031] 当监测到所述Redis与所述爬虫算子中存在相同的关键字时,将所述Redis中所述关键字对应的页面数据添加到所述ETL平台的ETL数据流中。

[0032] 进一步的,所述对所述ETL数据流进行ETL处理,得到库存数据包括:

[0033] 从所述终端获取ETL设置信息;

[0034] 根据所述ETL设置信息选取处理引擎对所述ETL数据流进行ETL处理;

[0035] 将ETL处理后的ETL数据流进行存储,得到库存数据。

[0036] 为了解决上述技术问题,本申请实施例还提供一种页面数据处理装置,采用了如下所述的技术方案:

[0037] 算子选取模块,用于当接收到终端发送的选取指令时,从ETL平台部署的数据处理算子中选取爬虫算子;其中,所述爬虫算子为实现爬虫功能的算子;

[0038] 信息获取模块,用于根据所述终端的爬虫配置页面中触发的配置指令,获取爬虫配置信息;

[0039] 算子配置模块,用于根据所述爬虫配置信息配置所述爬虫算子;

[0040] 算子运行模块,用于通过爬虫应用运行配置完毕的爬虫算子,并指示所述爬虫应用将爬取到的页面数据存储在Redis中;

[0041] 数据添加模块,用于将所述Redis中的所述页面数据添加到所述ETL平台的ETL数据流中;

[0042] 数据处理模块,用于对所述ETL数据流进行ETL处理,得到库存数据。

[0043] 为了解决上述技术问题,本申请实施例还提供一种计算机设备,包括存储器和处理器,所述存储器中存储有计算机程序,所述处理器执行所述计算机程序时实现上述所述的页面数据处理方法。

[0044] 为了解决上述技术问题，本申请实施例还提供一种计算机可读存储介质，所述计算机可读存储介质上存储有计算机程序，所述计算机程序被处理器执行时实现上述所述的页面数据处理方法。

[0045] 与现有技术相比，本申请实施例主要有以下有益效果：先根据选取指令，从ETL平台中选取爬虫算子，ETL平台集成部署有包括爬虫算子在内的多种数据处理算子，能对数据进行多种处理；用户在终端的配置页面中进行配置操作触发配置指令，依据配置指令获取爬虫配置信息，简单快捷，提高了爬虫算子的配置效率；爬虫应用运行爬虫算子，从页面中爬取页面数据并存储在Redis中；Redis是一种响应快速、支持多批量数据存储的数据库，通过Redis缓存页面数据，保证了ETL平台能通过多个爬虫算子同时爬取页面数据，保证了页面数据的获取速度；最后将Redis中的页面数据添加到ETL平台的ETL数据流中，并进行ETL处理得到库存数据，使得ETL平台能够实现对页面数据的一站式处理，提高了对页面数据的处理效率。

附图说明

[0046] 为了更清楚地说明本申请中的方案，下面将对本申请实施例描述中所需要使用的附图作一个简单介绍，显而易见地，下面描述中的附图是本申请的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的附图。

[0047] 图1是本申请可以应用于其中的示例性系统架构图；

[0048] 图2是根据本申请的页面数据处理方法的一个实施例的流程图；

[0049] 图3是根据本申请的页面数据处理装置的一个实施例的结构示意图；

[0050] 图4是根据本申请的计算机设备的一个实施例的结构示意图。

具体实施方式

[0051] 除非另有定义，本文所使用的所有的技术和科学术语与属于本申请的技术领域的技术人员通常理解的含义相同；本文中在申请的说明书中所使用的术语只是为了描述具体的实施例的目的，不是旨在于限制本申请；本申请的说明书和权利要求书及上述附图说明中的术语“包括”和“具有”以及它们的任何变形，意图在于覆盖不排他的包含。本申请的说明书和权利要求书或上述附图中的术语“第一”、“第二”等是用于区别不同对象，而不是用于描述特定顺序。

[0052] 在本文中提及“实施例”意味着，结合实施例描述的特定特征、结构或特性可以包含在本申请的至少一个实施例中。在说明书中的各个位置出现该短语并不一定均是指相同的实施例，也不是与其它实施例互斥的独立的或备选的实施例。本领域技术人员显式地和隐式地理解的是，本文所描述的实施例可以与其它实施例相结合。

[0053] 为了使本技术领域的人员更好地理解本申请方案，下面将结合附图，对本申请实施例中的技术方案进行清楚、完整地描述。

[0054] 如图1所示，系统架构100可以包括终端设备101、102、103，网络104和服务器105。网络104用以在终端设备101、102、103和服务器105之间提供通信链路的介质。网络104可以包括各种连接类型，例如有线、无线通信链路或者光纤电缆等等。

[0055] 用户可以使用终端设备101、102、103通过网络104与服务器105交互，以接收或发

送消息等。终端设备101、102、103上可以安装有各种通讯客户端应用，例如网页浏览器应用、购物类应用、搜索类应用、即时通信工具、邮箱客户端、社交平台软件等。

[0056] 终端设备101、102、103可以是具有显示屏并且支持网页浏览的各种电子设备，包括但不限于智能手机、平板电脑、电子书阅读器、MP3播放器(Moving Picture Experts Group Audio Layer III，动态影像专家压缩标准音频层面3)、MP4(Moving Picture Experts Group Audio Layer IV，动态影像专家压缩标准音频层面4)播放器、膝上型便携计算机和台式计算机等等。

[0057] 服务器105可以是提供各种服务的服务器，例如对终端设备101、102、103上显示的页面提供支持的后台服务器。

[0058] 需要说明的是，本申请实施例所提供的页面数据处理方法一般由服务器执行，相应地，页面数据处理装置一般设置于服务器中。

[0059] 应该理解，图1中的终端设备、网络和服务器的数目仅仅是示意性的。根据实现需要，可以具有任意数目的终端设备、网络和服务器。

[0060] 继续参考图2，示出了根据本申请的页面数据处理方法的一个实施例的流程图。所述的页面数据处理方法，包括以下步骤：

[0061] 步骤S201，当接收到终端发送的选取指令时，从ETL平台部署的数据处理算子中选取爬虫算子；其中，爬虫算子为实现爬虫功能的算子。

[0062] 在本实施例中，页面数据处理方法运行于其上的电子设备(例如图1所示的服务器)可以通过有线连接方式或者无线连接方式于终端进行通信。需要指出的是，上述无线连接方式可以包括但不限于3G/4G连接、WiFi连接、蓝牙连接、WiMAX连接、Zigbee连接、UWB(ultra wideband)连接、以及其他现在已知或将来开发的无线连接方式。

[0063] 其中，选取指令可以是选取ETL平台中数据处理算子的指令。ETL平台可以是部署在服务器中的软件平台，可以实现ETL功能。爬虫算子可以是实现爬虫功能的算子。

[0064] 具体地，本申请中的ETL平台是自研的数据处理平台，支持可视化的触控操作。用户通过终端打开ETL平台的编辑页面，编辑页面中存在多个算子标识，每个算子标识代表不同的数据处理算子。ETL平台集成了多种数据处理算子，能够对数据进行多种处理。数据处理算子是对数据处理逻辑的打包，ETL平台的源代码中包含了多种数据处理算子的程序。用户在编辑页面中选取数据处理算子，服务器运行与数据处理算子对应的程序；数据在数据处理算子之间传递，从而实现对数据流的处理。

[0065] 用户在终端的编辑页面中选定爬虫算子，使终端触发选取指令并将选取指令发送至服务器。服务器依据选取指令从ETL平台预先部署的数据处理算子中选取爬虫算子。

[0066] 在一个实施例中，用户在编辑页面中通过光标持续作用于爬虫算子标识，将爬虫算子标识拖入编辑页面的设置区域内，触发选取指令。终端将选取指令发送至服务器，服务器根据选取指令，从ETL平台部署的数据处理算子中，选取爬虫算子。

[0067] 步骤S202，根据终端的爬虫配置页面中触发的配置指令，获取爬虫配置信息。

[0068] 其中，爬虫配置页面可以是对爬虫算子进行配置的页面。配置指令可以是用户在配置页面中进行配置操作触发的指令。爬虫配置信息用于对爬虫算子进行设置。

[0069] 具体地，用户可以点击设置区域内的爬虫算子，进入爬虫算子的配置页面。配置页面支持可视化交互的配置方式。终端记录用户在爬虫配置页面中的配置操作，当接收到配

置页面中触发的确认指令时，根据记录的配置操作生成配置指令，并将配置指令发送至服务器，服务器依据接收到的配置指令获取爬虫配置信息。

[0070] 在一个实施例中，爬虫配置信息包括URL、字段信息、xpath路径、中间件等信息，中间件等信息可以是cookie、header、proxy等。

[0071] 其中，URL(Uniform Resource Locator)为统一资源定位符，用于唯一标识信息资源在万维网上的地址。字段信息可以是URL所对应的页面中的字段。xpath即为XML路径语言(XML Path Language)，它是一种用来确定XML文档中某部分位置的语言。用户可以在浏览器中打开页面，通过键盘上的F12按键，或者在页面中点击鼠标右键并点击“检查”选项，进入开发者工具调试页面。用户在打开的页面中点击需要爬取的标签，从而在开发者工具调试页面中获取该标签的xpath路径，并将xpath路径拷贝到配置页面中。xpath路径用于指示爬虫应用爬取xpath路径所对应的页面数据。

[0072] Cookie类型为“小型文本文件”，是某些网站为了辨别用户身份，进行会话跟踪而储存在用户本地终端上的数据(通常经过加密)，由用户本地终端暂时或永久保存。Header为http请求头，通常http(超文本传输协议)消息包括终端向服务器的请求消息和服务器向终端的响应消息，这两种类型的消息中包括请求头。Proxy为代理服务器，通过给爬虫算子配置代理服务器来防止页面反爬取。

[0073] 步骤S203，根据爬虫配置信息配置爬虫算子。

[0074] 具体地，服务器根据获取到的爬虫配置信息配置爬虫算子，选取的爬虫算子可以是一段模板程序，根据模板程序中可替换变量的标注说明，将可替换变量替换为爬虫配置信息。

[0075] 在一个实施例中，爬虫算子可以是配置文件，服务器将爬虫配置信息封装进配置文件中。

[0076] 步骤S204，通过爬虫应用运行配置完毕的爬虫算子，并指示爬虫应用将爬取到的页面数据存储在Redis中。

[0077] 其中，爬虫应用可以是实现爬虫功能的应用程序。Redis是一个key-value存储系统，Redis作为缓存工具，具备高性能、高响应等特性。

[0078] 具体地，爬虫应用作为一种爬虫工具，可以独立于ETL平台之外。现有的爬虫工具需要在客户端使用，而本申请中的爬虫应用提供web页面，用户在web页面中使用爬虫应用，以提高爬虫应用的便捷性并降低使用爬虫应用的限制条件。爬虫应用可以提供接口给ETL平台，ETL平台依据爬虫配置信息对爬虫算子进行配置，由爬虫应用依据配置完毕的爬虫算子在页面中爬取页面数据。

[0079] 将爬虫应用独立于ETL平台，通过接口的方式调用爬虫应用，使得爬虫应用可以结合ETL平台实现页面数据的ETL；在不结合ETL平台时，只需要提供给爬虫应用需要的爬虫配置信息，爬虫应用便可作为单独的爬虫工具使用。

[0080] 爬虫应用可以将爬取到的页面数据存储到Redis中。在通过配置页面操作ETL平台之前，Redis已经部署完毕。一个爬虫应用可以运行多个爬虫算子，爬虫应用运行爬虫算子时爬取到的页面数据，可以存储在一个Redis中。

[0081] 步骤S205，将Redis中的页面数据添加到ETL平台的ETL数据流中。

[0082] 其中，ETL数据流可以是ETL平台中的有序的数据序列。

[0083] 具体地，服务器从Redis中读取页面数据，将页面数据加载到ETL平台的ETL数据流中。服务器可以实时监测Redis，当Redis中出现页面数据时，即可触发读取指令，依据读取指令将Redis中的页面数据添加到ETL平台的ETL数据流中。

[0084] 步骤S206，对ETL数据流进行ETL处理，得到库存数据。

[0085] 具体地，服务器通过ETL平台对ETL数据流进行ETL处理，并将处理后的ETL数据流进行存储，得到库存数据。用户可以在终端的编辑页面中对ETL处理进行设置，服务器依据设置对ETL数据流进行处理。

[0086] 需要强调的是，为进一步保证上述库存数据的私密和安全性，上述库存数据还可以存储于一区块链的节点中。

[0087] 本申请所指区块链是分布式数据存储、点对点传输、共识机制、加密算法等计算机技术的新型应用模式。区块链(Blockchain)，本质上是一个去中心化的数据库，是一串使用密码学方法相关联产生的数据块，每一个数据块中包含了一批次网络交易的信息，用于验证其信息的有效性(防伪)和生成下一个区块。区块链可以包括区块链底层平台、平台产品服务层以及应用服务层等。

[0088] 本申请可应用于智慧城市领域中，从而推动智慧城市的建设。本申请中通过ETL平台得到的库存数据可进一步通过自然语言处理，从而结合智能搜索或智能推荐。

[0089] 例如，本申请可应用于智慧政务领域中的数据治理，通过ETL平台从各政府网站爬取政策法规，在搭建的事务自助办理平台中，根据用户选择的需要办理的事务，自动推荐相关的政策法规从而对用户进行引导。或者，本申请可应用于智慧教育领域，从网上爬取到大量题目后，向使用学习类应用的学生推荐试卷或者题目。又比如，可以应用于智慧医疗领域，在爬取到各种药品的使用说明后，根据用户的搜索请求向用户展示某药品的详细信息。

[0090] 本实施例中，先根据选取指令，从ETL平台中选取爬虫算子，ETL平台集成部署有包括爬虫算子在内的多种数据处理算子，能对数据进行多种处理；用户在终端的配置页面中进行配置操作触发配置指令，依据配置指令获取爬虫配置信息，简单快捷，提高了爬虫算子的配置效率；爬虫应用运行爬虫算子，从页面中爬取页面数据并存储在Redis中；Redis是一种响应快速、支持多批量数据存储的数据库，通过Redis缓存页面数据，保证了ETL平台能通过多个爬虫算子同时爬取页面数据，保证了页面数据的获取速度；最后将Redis中的页面数据添加到ETL平台的ETL数据流中，并进行ETL处理得到库存数据，使得ETL平台能够实现对页面数据的一站式处理，提高了对页面数据的处理效率。

[0091] 进一步的，上述步骤S201可以包括：当接收到终端发送的选取指令时，读取ETL平台的状态标识；当通过状态标识确定ETL平台未处于数据输出状态时，从ETL平台部署的数据处理算子中选取爬虫算子，并通过终端展示爬虫算子的爬虫配置页面。

[0092] 其中，状态标识用于标记ETL平台当前的数据处理状态。

[0093] 具体地，ETL平台具有数据处理状态，例如，当ETL平台可以处于数据输入状态、数据抽取状态、数据转换状态或数据输出状态。不同的数据处理状态可以同时存在，比如ETL平台可以同时处于数据输入状态和数据抽取状态。ETL平台使用状态标识标记ETL平台当前所处的状态。

[0094] 当服务器接收到终端发送的选取指令时，先获取ETL平台的状态标识，由状态标识确定ETL平台当前所处的数据处理状态。ETL平台在不同的数据处理状态下能进行的操作具

有限制，当处于数据输出状态时，为了有序而准确地输出库存数据，ETL平台不允许读入新的页面数据；而当根据状态标识确定ETL平台未处于数据输出状态时，允许读取新的页面数据，爬虫算子也处于可用状态，服务器从ETL平台部署的数据处理算子中选取爬虫算子，并指示终端展示爬虫算子的爬虫配置页面。

[0095] 本实施例中，在接收到选取指令后先通过状态标识确定ETL平台的数据处理状态，仅允许在ETL平台未处于数据输出状态时选取爬虫算子，以保证能够有序且准确地输出库存数据。

[0096] 进一步的，上述步骤S202可以包括：通过终端获取爬虫配置页面中的确认选项及文本框文本；接收终端根据获取到的确认选项及文本框文本触发的配置指令；根据配置指令获取爬虫配置信息。

[0097] 具体地，配置页面支持可视化交互的配置方式，包括点击选项、文本框输入等配置方式。终端记录爬虫配置页面中被点击的选项以及文本框中输入的文本，当接收到配置页面中触发的确认指令时，根据被点击的选项以及文本框中输入的文本生成配置指令并将配置指令发送至服务器，服务器依据接收到的配置指令获取爬虫配置信息。

[0098] 确认指令可以通过点击配置页面中的虚拟确认按钮触发，也可以是在监测到完成对一处选项的选定或文本框的输入后自动触发。

[0099] 本实施例中，用以生成配置指令的确认选项以及文本框文本以可视化交互的方式在爬虫配置页面中录入，操作简便，且爬虫配置信息依据配置指令获取，提高了爬虫配置信息的获取效率。

[0100] 进一步的，上述步骤S202可以包括：当接收到终端发送的流展示指令时，通过终端的爬虫配置页面展示ETL平台中的ETL数据流；接收在展示的ETL数据流中选中待爬取字段触发的配置指令；将配置指令中的待爬取字段添加为爬虫配置信息。

[0101] 其中，流展示指令可以是指示服务器通过终端展示ETL平台中ETL数据流的指令。待爬取字段可以是ETL数据流中的字段，待爬取字段可以提供给爬虫算子进行页面数据的爬取。

[0102] 具体地，爬虫配置信息可以来自于ETL平台中的ETL数据流。当用户想对ETL数据流进行补充爬取时，可以在爬虫配置页面中点击流展示按钮，使终端触发流展示指令并将流展示指令发送至服务器。服务器接收到流展示指令后，通过终端展示当前ETL平台中的ETL数据流。

[0103] 用户可以在终端看到ETL数据流包含哪些字段，并选取字段作为待爬取字段。待爬取字段可以被封装进配置指令中发送至服务器，服务器对配置指令进行解析，将解析得到的待爬取字段作为爬虫配置信息。

[0104] 举例说明，ETL数据流中包括一个名单列表，该名单列表中记录了一些人员的姓名和职位。用户想针对名单列表中的某个人A进行爬取时，可以选中姓名“A”作为待爬取字段，“A”也将作为爬虫配置信息。当爬取到“A”的页面数据后，将爬取到的页面数据与ETL数据流进行合并。

[0105] 本实施例中，通过终端展示ETL平台中的ETL数据流，并支持从ETL数据流中选取待爬取字段作为爬虫配置信息，丰富了爬虫配置信息的配置方式。

[0106] 进一步的，上述步骤S202还包括：获取终端的爬虫配置页面中触发的配置指令中

所包含的URL;将URL添加为爬虫配置信息;或者,当终端的爬虫配置页面中触发的配置指令中包含流获取指令时,从ETL平台的ETL数据流中查询URL标识;读取URL标识所对应的ETL数据流作为爬虫配置信息。

[0107] 其中,流获取指令可以是指示服务器从ETL数据流中获取URL的指令。URL标识用于标识ETL数据流中的URL。

[0108] 具体地,爬虫配置信息中的URL可以由用户手动录入,也可以从ETL平台的ETL数据流中提取。用户在配置页面中的URL文本框中录入URL后,录入的URL被封装到配置指令中发送至服务器,服务器将配置指令中的URL作为爬虫配置信息。当用户在配置页面中点击从ETL数据流中获取URL的虚拟按钮后,触发流获取指令,流获取指令被封装进配置指令发送至服务器。服务器解析到配置指令中的流获取指令时,从ETL数据流中查找URL标识,并提取URL标识所对应的ETL数据流作为爬虫配置信息。服务器还可以通过终端展示提取到的URL。

[0109] 用户可以配置多个URL,以使爬虫算子可以爬取多个URL所对应的页面数据。

[0110] 本实施例中,可以直接从配置指令中获取录入的URL,或者从ETL数据流中获取URL,丰富了URL获取方式。

[0111] 在一个实施例中,上述步骤S205可以包括:监测Redis与爬虫算子中的关键字;当监测到Redis与爬虫算子中存在相同的关键字时,将Redis中关键字对应的页面数据添加到ETL平台的ETL数据流中。

[0112] 其中,关键字可以是Redis中的key,Redis是一种key-value数据库,key是关键字,value为值,Redis以键值对的形式存储页面数据。

[0113] 具体地,用户还可以在爬虫配置页面中录入关键字,关键字也被封装进爬虫算子中。爬虫应用爬取到页面数据后,依据爬虫算子中的关键字给页面数据添加关键字。页面数据存储在Redis的队列中。服务器实时监测Redis中的关键字,当Redis中出现与爬虫算子中的关键字相同的关键字时,从Redis中读取该爬虫算子所对应的页面数据。

[0114] Redis作为消息队列使用时可以存入多组消息,因此Redis中的队列可以存储多个关键字所对应的页面数据,保证了ETL平台可以同时运行多个爬虫算子。同时,Redis的高响应速度提高了服务器获取页面数据的速度。

[0115] 当监测到相同的关键字时,服务器即可从Redis中读取页面数据并合并到ETL数据流中,而不必等待爬取结束,对页面数据进行分散式处理,以提升ETL平台的数据处理速度。

[0116] 本实施例中,通过Redis缓存爬取到的页面数据,依据关键字从Redis中读取页面数据到ETL数据流中,使得服务器可以同时从Redis中读取多组页面数据,提高了服务器获取页面数据的速度。

[0117] 在一个实施例中,上述步骤S206可以包括:从终端获取ETL设置信息;根据ETL设置信息选取处理引擎对ETL数据流进行ETL处理;将ETL处理后的ETL数据流进行存储,得到库存数据。

[0118] 其中,设置信息用于指示对ETL数据流进行ETL处理。

[0119] 具体地,用户可以在ETL平台的编辑页面中对ETL进行设置,得到ETL设置信息。服务器依据ETL设置信息进行ETL处理。ETL设置信息中可以包括ETL时所使用的处理引擎,包括spark、kettle等,服务器从ETL平台部署的多种处理引擎中选取ETL设置信息对应的处理引擎,依据处理引擎对ETL数据流进行ETL处理。其中,Spark 是为大规模数据处理而设计的

快速通用的计算引擎;Kettle是一种可以运行于多种操作系统上的ETL工具。

[0120] ETL处理完毕后可以进行多种形式的存储,包括但不限于数据库、消息队列存储、大数据存储、文件存储,还可以以Excel、Word、JSON等格式存储,得到库存数据。ETL设置信息中可以包括存储方式,ETL平台依据ETL设置信息中的存储方式对ETL处理后的ETL数据流进行存储。

[0121] 本实施例中,依据ETL设置信息可以选择不同的处理引擎对ETL数据流进行ETL处理,保证了ETL处理可以有序地进行。

[0122] 本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程,是可以通过计算机可读指令来指令相关的硬件来完成,该计算机可读指令可存储于一计算机可读取存储介质中,该程序在执行时,可包括如上述各方法的实施例的流程。其中,前述的存储介质可为磁碟、光盘、只读存储记忆体(Read-Only Memory,ROM)等非易失性存储介质,或随机存储记忆体(Random Access Memory,RAM)等。

[0123] 应该理解的是,虽然附图的流程图中的各个步骤按照箭头的指示依次显示,但是这些步骤并不是必然按照箭头指示的顺序依次执行。除非本文中有明确的说明,这些步骤的执行并没有严格的顺序限制,其可以以其他的顺序执行。而且,附图的流程图中的至少一部分步骤可以包括多个子步骤或者多个阶段,这些子步骤或者阶段并不必然是在同一时刻执行完成,而是可以在不同的时刻执行,其执行顺序也不必然是依次进行,而是可以与其他步骤或者其他步骤的子步骤或者阶段的至少一部分轮流或者交替地执行。

[0124] 进一步参考图3,作为对上述图2所示方法的实现,本申请提供了一种页面数据处理装置的一个实施例,该装置实施例与图2所示的方法实施例相对应,该装置具体可以应用于各种电子设备中。

[0125] 如图3所示,本实施例所述的页面数据处理装置300包括:算子选取模块301、信息获取模块302、算子配置模块303、算子运行模块304、数据添加模块305以及数据处理模块306。其中:

[0126] 算子选取模块301,用于当接收到终端发送的选取指令时,从ETL平台部署的数据处理算子中选取爬虫算子;其中,爬虫算子为实现爬虫功能的算子。

[0127] 信息获取模块302,用于根据终端的爬虫配置页面中触发的配置指令,获取爬虫配置信息。

[0128] 算子配置模块303,用于根据爬虫配置信息配置爬虫算子。

[0129] 算子运行模块304,用于通过爬虫应用运行配置完毕的爬虫算子,并指示爬虫应用将爬取到的页面数据存储在Redis中。

[0130] 数据添加模块305,用于将Redis中的页面数据添加到ETL平台的ETL数据流中。

[0131] 数据处理模块306,用于对ETL数据流进行ETL处理,得到库存数据。

[0132] 本实施例中,先根据选取指令,从ETL平台中选取爬虫算子,ETL平台集成部署有包括爬虫算子在内的多种数据处理算子,能对数据进行多种处理;用户在终端的配置页面中进行配置操作触发配置指令,依据配置指令获取爬虫配置信息,简单快捷,提高了爬虫算子的配置效率;爬虫应用运行爬虫算子,从页面中爬取页面数据并存储在Redis中;Redis是一种响应快速、支持多批量数据存储的数据库,通过Redis缓存页面数据,保证了ETL平台能通过多个爬虫算子同时爬取页面数据,保证了页面数据的获取速度;最后将Redis中的页面数

据添加到ETL平台的ETL数据流中，并进行ETL处理得到库存数据，使得ETL平台能够实现对页面数据的一站式处理，提高了对页面数据的处理效率。

[0133] 在本实施例的一些可选的实现方式中，上述算子选取模块301包括：标识读取子模块和算子选取子模块，其中：

[0134] 标识读取子模块：用于当接收到终端发送的选取指令时，读取ETL平台的状态标识。

[0135] 算子选取子模块，用于当通过状态标识确定ETL平台未处于数据输出状态时，从ETL平台部署的数据处理算子中选取爬虫算子，并通过终端展示爬虫算子的爬虫配置页面。

[0136] 本实施例中，在接收到选取指令后先通过状态标识确定ETL平台的数据处理状态，仅允许在ETL平台未处于数据输出状态时选取爬虫算子，以保证能够有序且准确地输出库存数据。

[0137] 在本实施例的一些可选的实现方式中，上述信息获取模块302包括：获取子模块、触发子模块和配置获取子模块，其中：

[0138] 获取子模块，用于通过终端获取爬虫配置页面中的确认选项及文本框文本。

[0139] 触发子模块，用于接收终端根据获取到的确认选项及文本框文本触发的配置指令。

[0140] 配置获取子模块，用于根据配置指令获取爬虫配置信息。

[0141] 本实施例中，用以生成配置指令的确认选项以及文本框文本以可视化交互的方式在爬虫配置页面中录入，操作简便，且爬虫配置信息依据配置指令获取，提高了爬虫配置信息的获取效率。

[0142] 在本实施例的一些可选的实现方式中，上述信息获取模块302还包括：数据流展示子模块、指令接收子模块和字段添加子模块，其中：

[0143] 数据流展示子模块，用于当接收到终端发送的流展示指令时，通过终端的爬虫配置页面展示ETL平台中的ETL数据流。

[0144] 指令接收子模块，用于接收在展示的ETL数据流中选中待爬取字段触发的配置指令。

[0145] 字段添加子模块，用于将配置指令中的待爬取字段添加为爬虫配置信息。

[0146] 本实施例中，通过终端展示ETL平台中的ETL数据流，并支持从ETL数据流中选取待爬取字段作为爬虫配置信息，丰富了爬虫配置信息的配置方式。

[0147] 在本实施例的一些可选的实现方式中，上述信息获取模块302还包括：URL获取子模块以及URL添加子模块，或者标识查询子模块以及数据流读取子模块，其中：

[0148] URL获取子模块，用于获取终端的爬虫配置页面中触发的配置指令中所包含的URL。

[0149] URL添加子模块，用于将URL添加为爬虫配置信息。

[0150] 标识查询子模块，用于当终端的爬虫配置页面中触发的配置指令中包含流获取指令时，从ETL平台的ETL数据流中查询URL标识。

[0151] 数据流读取子模块，用于读取URL标识所对应的ETL数据流作为爬虫配置信息。

[0152] 本实施例中，可以直接从配置指令中获取录入的URL，或者从ETL数据流中获取URL，丰富了URL获取方式。

[0153] 在本实施例的一些可选的实现方式中，上述数据添加模块305包括关键字监测子模块和数据添加子模块，其中：

[0154] 关键字监测子模块，用于监测Redis与爬虫算子中的关键字。

[0155] 数据添加子模块，用于当监测到Redis与爬虫算子中存在相同的关键字时，将Redis中关键字对应的页面数据添加到ETL平台的ETL数据流中。

[0156] 本实施例中，通过Redis缓存爬取到的页面数据，依据关键字从Redis中读取页面数据到ETL数据流中，使得服务器可以同时从Redis中读取多组页面数据，提高了服务器获取页面数据的速度。

[0157] 在本实施例的一些可选的实现方式中，上述数据处理模块306包括：设置获取子模块、处理子模块以及存储子模块，其中：

[0158] 设置获取子模块，用于从终端获取ETL设置信息。

[0159] 处理子模块，用于根据ETL设置信息选取处理引擎对ETL数据流进行ETL处理。

[0160] 存储子模块，用于将ETL处理后的ETL数据流进行存储，得到库存数据。

[0161] 本实施例中，依据ETL设置信息可以选择不同的处理引擎对ETL数据流进行ETL处理，保证了ETL处理可以有序地进行。

[0162] 为解决上述技术问题，本申请实施例还提供计算机设备。具体请参阅图4，图4为本实施例计算机设备基本结构框图。

[0163] 所述计算机设备4包括通过系统总线相互通信连接存储器41、处理器42、网络接口43。需要指出的是，图中仅示出了具有组件41-43的计算机设备4，但是应理解的是，并不要求实施所有示出的组件，可以替代的实施更多或者更少的组件。其中，本技术领域技术人员可以理解，这里的计算机设备是一种能够按照事先设定或存储的指令，自动进行数值计算和/或信息处理的设备，其硬件包括但不限于微处理器、专用集成电路(Application Specific Integrated Circuit，ASIC)、可编程门阵列(Field－Programmable Gate Array，FPGA)、数字处理器 (Digital Signal Processor，DSP)、嵌入式设备等。

[0164] 所述计算机设备可以是桌上型计算机、笔记本、掌上电脑及云端服务器等计算设备。所述计算机设备可以与用户通过键盘、鼠标、遥控器、触摸板或声控设备等方式进行人机交互。

[0165] 所述存储器41至少包括一种类型的可读存储介质，所述可读存储介质包括闪存、硬盘、多媒体卡、卡型存储器(例如，SD或DX存储器等)、随机访问存储器(RAM)、静态随机访问存储器(SRAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、可编程只读存储器(PROM)、磁性存储器、磁盘、光盘等。在一些实施例中，所述存储器41可以是所述计算机设备4的内部存储单元，例如该计算机设备4的硬盘或内存。在另一些实施例中，所述存储器41也可以是所述计算机设备4的外部存储设备，例如该计算机设备4上配备的插接式硬盘，智能存储卡(Smart Media Card, SMC)，安全数字(Secure Digital, SD)卡，闪存卡(Flash Card)等。当然，所述存储器41还可以既包括所述计算机设备4的内部存储单元也包括其外部存储设备。本实施例中，所述存储器41通常用于存储安装于所述计算机设备4的操作系统和各类应用软件，例如页面数据处理方法的计算机可读指令等。此外，所述存储器41还可以用于暂时地存储已经输出或者将要输出的各类数据。

[0166] 所述处理器42在一些实施例中可以是中央处理器(Central Processing Unit，

CPU)、控制器、微控制器、微处理器、或其他数据处理芯片。该处理器42通常用于控制所述计算机设备4的总体操作。本实施例中，所述处理器42用于运行所述存储器41中存储的计算机可读指令或者处理数据，例如运行所述页面数据处理方法的计算机可读指令。

[0167] 所述网络接口43可包括无线网络接口或有线网络接口，该网络接口43通常用于在所述计算机设备4与其他电子设备之间建立通信连接。

[0168] 本实施例中提供的计算机设备可以执行上述页面数据处理方法。此处页面数据处理方法可以是上述各个实施例的页面数据处理方法。

[0169] 本实施例中，先根据选取指令，从ETL平台中选取爬虫算子，ETL平台集成部署有包括爬虫算子在内的多种数据处理算子，能对数据进行多种处理；用户在终端的配置页面中进行配置操作触发配置指令，依据配置指令获取爬虫配置信息，简单快捷，提高了爬虫算子的配置效率；爬虫应用运行爬虫算子，从页面中爬取页面数据并存储在Redis中；Redis是一种响应快速、支持多批量数据存储的数据库，通过Redis缓存页面数据，保证了ETL平台能通过多个爬虫算子同时爬取页面数据，保证了页面数据的获取速度；最后将Redis中的页面数据添加到ETL平台的ETL数据流中，并进行ETL处理得到库存数据，使得ETL平台能够实现对页面数据的一站式处理，提高了对页面数据的处理效率。

[0170] 本申请还提供了另一种实施方式，即提供一种计算机可读存储介质，所述计算机可读存储介质存储有计算机可读指令，所述计算机可读指令可被至少一个处理器执行，以使所述至少一个处理器执行如上述的页面数据处理方法。

[0171] 本实施例中，先根据选取指令，从ETL平台中选取爬虫算子，ETL平台集成部署有包括爬虫算子在内的多种数据处理算子，能对数据进行多种处理；用户在终端的配置页面中进行配置操作触发配置指令，依据配置指令获取爬虫配置信息，简单快捷，提高了爬虫算子的配置效率；爬虫应用运行爬虫算子，从页面中爬取页面数据并存储在Redis中；Redis是一种响应快速、支持多批量数据存储的数据库，通过Redis缓存页面数据，保证了ETL平台能通过多个爬虫算子同时爬取页面数据，保证了页面数据的获取速度；最后将Redis中的页面数据添加到ETL平台的ETL数据流中，并进行ETL处理得到库存数据，使得ETL平台能够实现对页面数据的一站式处理，提高了对页面数据的处理效率。

[0172] 通过以上的实施方式的描述，本领域的技术人员可以清楚地了解到上述实施例方法可借助软件加必需的通用硬件平台的方式来实现，当然也可以通过硬件，但很多情况下前者是更佳的实施方式。基于这样的理解，本申请的技术方案本质上或者说对现有技术做出贡献的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(如ROM/RAM、磁碟、光盘)中，包括若干指令用以使得一台终端设备(可以是手机，计算机，服务器，空调器，或者网络设备等)执行本申请各个实施例所述的方法。

[0173] 显然，以上所描述的实施例仅仅是本申请一部分实施例，而不是全部的实施例，附图中给出了本申请的较佳实施例，但并不限制本申请的专利范围。本申请可以以许多不同的形式来实现，相反地，提供这些实施例的目的是使对本申请的公开内容的理解更加透彻全面。尽管参照前述实施例对本申请进行了详细的说明，对于本领域的技术人员来而言，其依然可以对前述各具体实施方式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效替换。凡是利用本申请说明书及附图内容所做的等效结构，直接或间接运用在其他相关的技术领域，均同理在本申请专利保护范围之内。

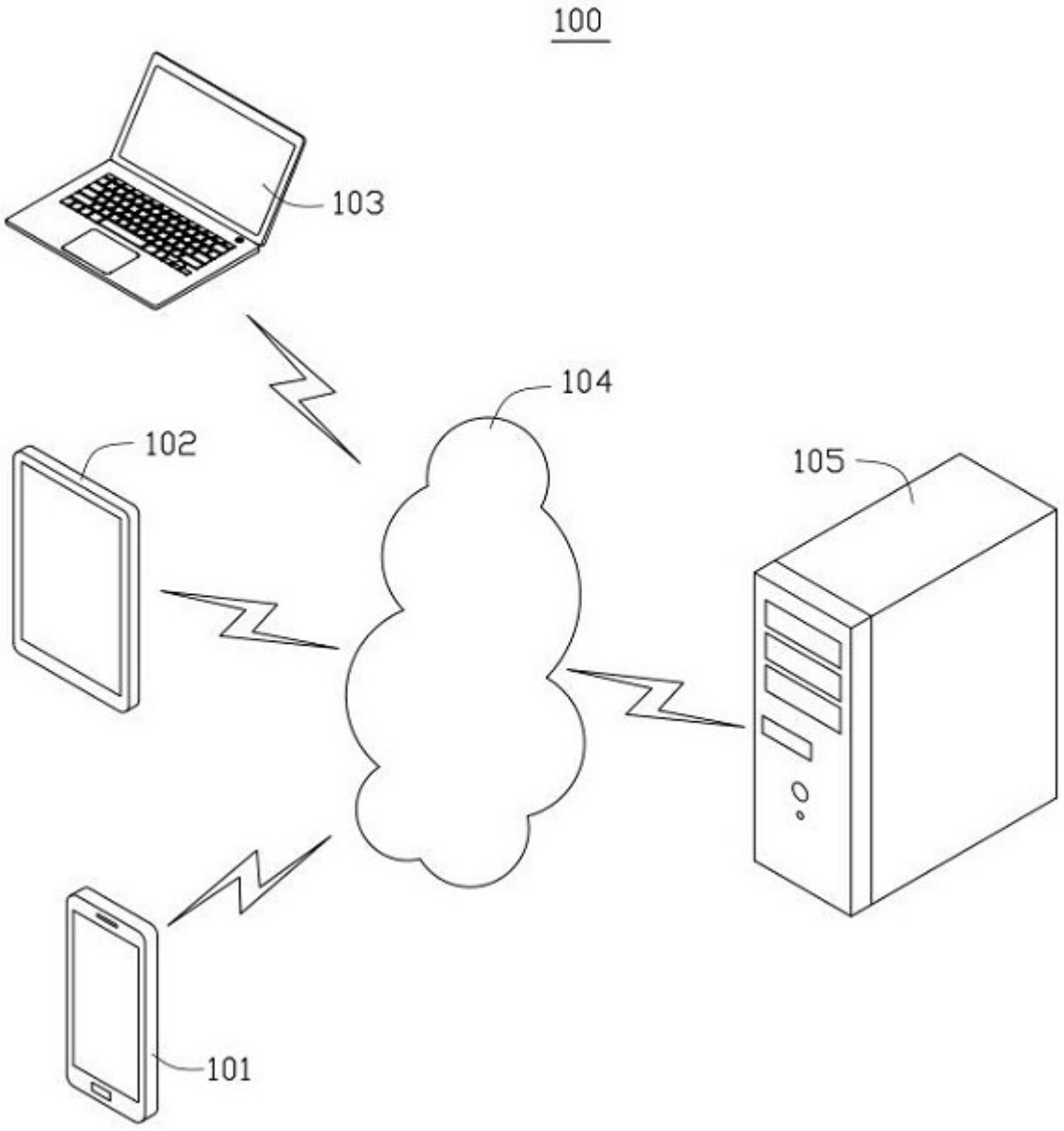


图1

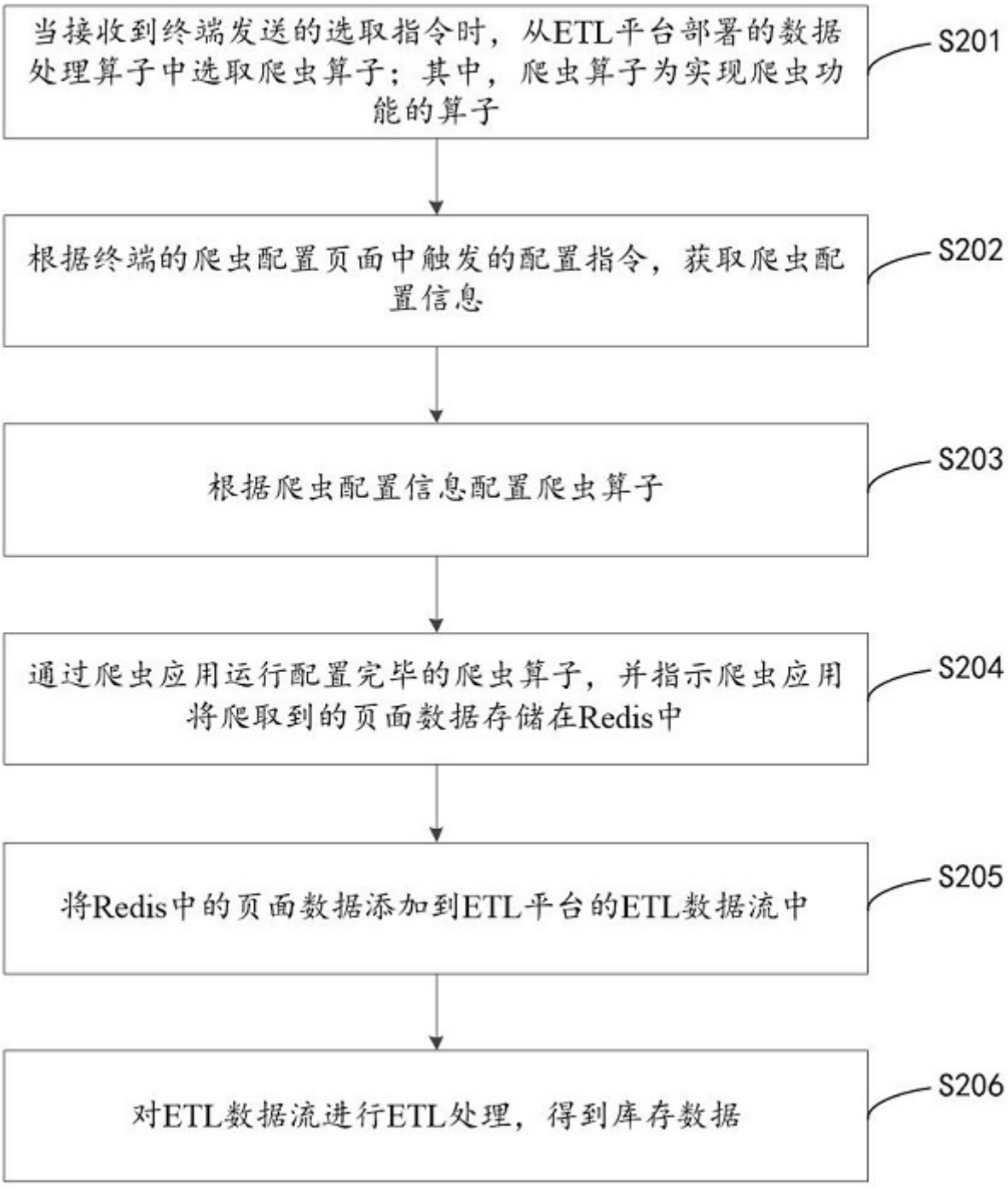

当接收到终端发送的选取指令时，从ETL平台部署的数据处理算子中选取爬虫算子；其中，爬虫算子为实现爬虫功能的算子
S201
根据终端的爬虫配置页面中触发的配置指令，获取爬虫配置信息
S202
根据爬虫配置信息配置爬虫算子
S203
通过爬虫应用运行配置完毕的爬虫算子，并指示爬虫应用将爬取到的页面数据存储在Redis中
S204
将Redis中的页面数据添加到ETL平台的ETL数据流中
S205
对ETL数据流进行ETL处理，得到库存数据
S206

图2

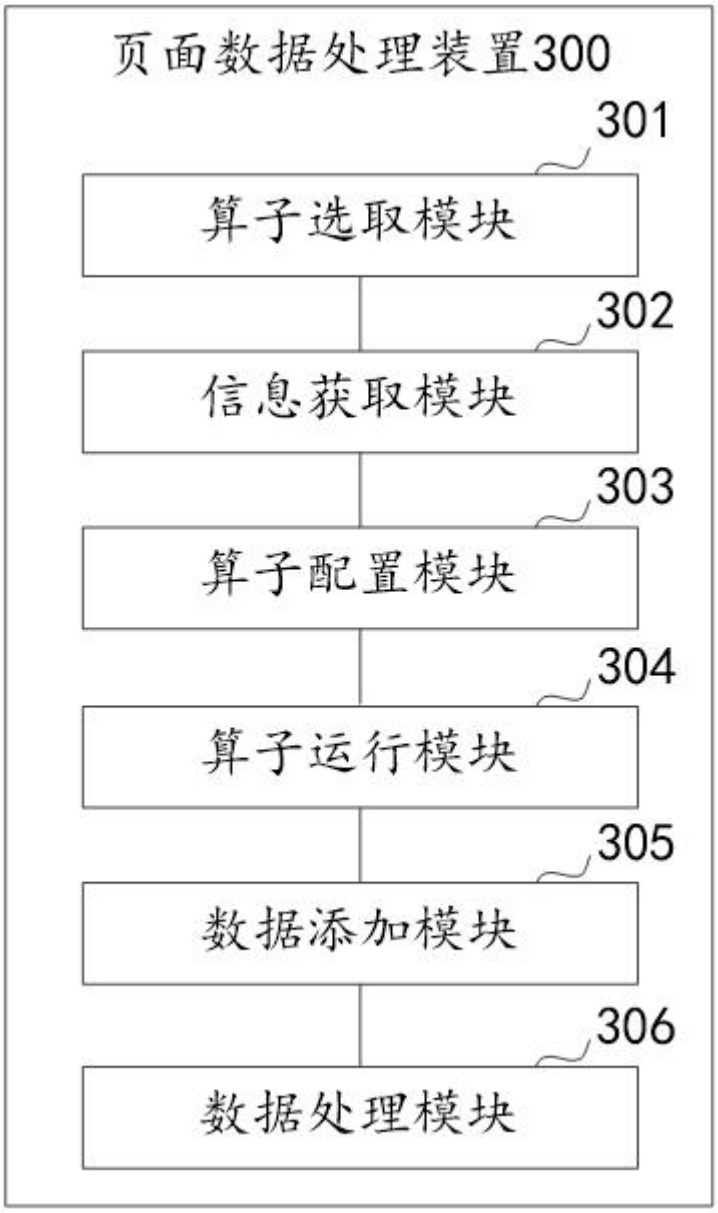

页面数据处理装置300
301
算子选取模块
302
信息获取模块
303
算子配置模块
304
算子运行模块
305
数据添加模块
306
数据处理模块

图3

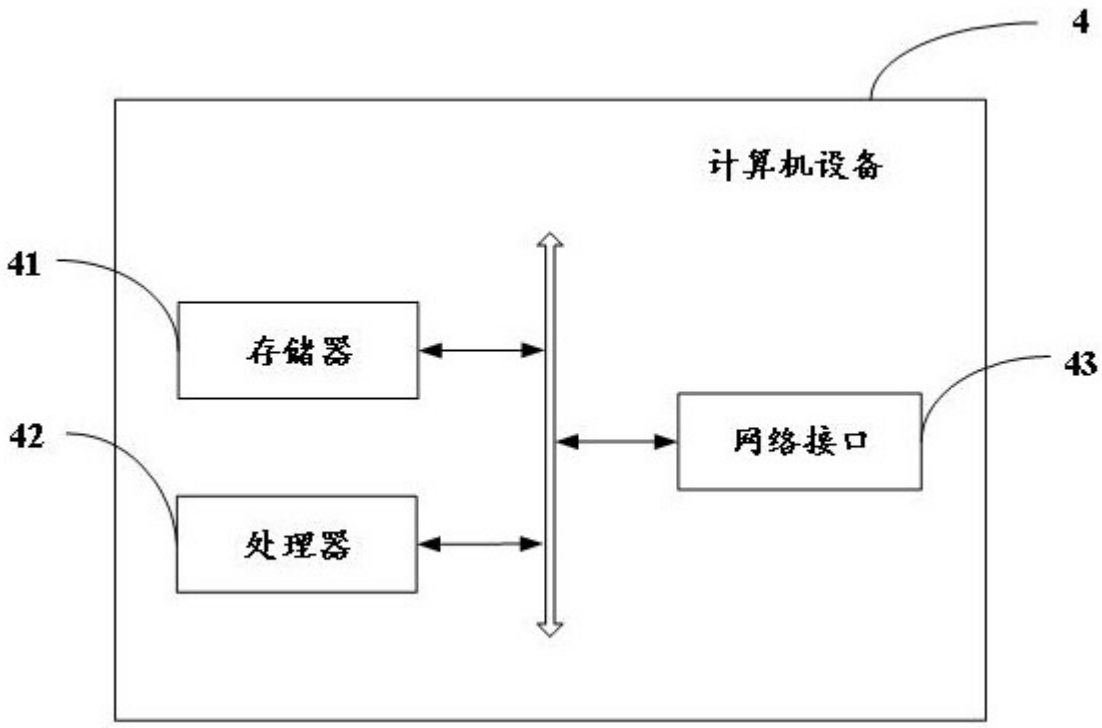


图4