(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 112650570 B
(45)授权公告日 2024.09.20

(21)申请号 202011598949.8
(22)申请日 2020.12.29
(65)同一申请的已公布的文献号
申请公布号 CN 112650570 A
(43)申请公布日 2021.04.13
(73)专利权人 百果园技术(新加坡)有限公司
地址 新加坡巴西班让路枫树商业城30号楼15层31A
(72)发明人 陈志坚
(74)专利代理机构 北京泽方誉航专利代理事务所(普通合伙) 11884
专利代理师 陈照辉
(51)Int.Cl.
G06F 9/48(2006.01)
G06F 9/54(2006.01)
G06F 16/951(2019.01)

(56)对比文件
CN 107908698 A,2018.04.13
CN 108334585 A,2018.07.27
CN 106168985 A,2016.11.30
CN 108520024 A,2018.09.11

审查员 王灿

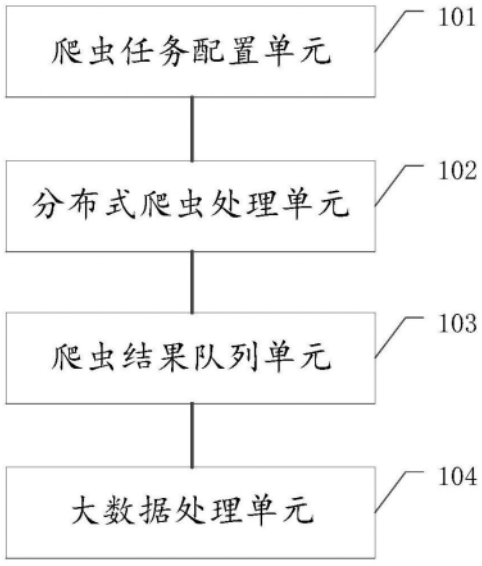
权利要求书2页 说明书9页 附图11页

(54)发明名称

可动态扩展的分布式爬虫系统、数据处理方法及装置

(57)摘要

本申请实施例公开了可动态扩展的分布式爬虫系统、数据处理方法及装置。本申请实施例提供的技术方案通过优化系统,在爬虫任务配置中隔离爬虫任务配置和条件配置,在执行爬虫任务时对不同任务信息的爬虫任务列入不同的任务队列等待,并且多线程分别执行不停任务队列的爬虫任务;通过降低系统内部的耦合度、提高了可动态扩展性能,避免内存泄露,大大提高开发效率,以及提高系统的吞吐量。



CN 112650570 B

1.可动态扩展的分布式爬虫系统,其特征在于,包括依次连接的爬虫任务配置单元、分布式爬虫处理单元、爬虫结果队列单元和大数据处理单元;

所述爬虫任务配置单元用于供用户配置爬虫任务和爬取内容限定条件,所述爬虫任务配置单元包括相互连接的任务配置模块和条件配置模块,所述任务配置模块用于供用户配置多个分别适用于不同平台的爬虫任务,所述条件配置模块用于供用户配置爬取内容限定条件,所述爬虫任务包括任务信息,所述任务信息包括平台信息和渠道信息;所述分布式爬虫处理单元用于接收爬虫任务,并根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列,从多个爬虫任务队列中分别获取爬虫任务并根据该爬虫任务对应的爬取内容限定条件获取目标资源;所述爬虫结果队列单元用于接收所述目标资源并将所述目标资源进行缓存;所述大数据处理单元用于从爬虫结果队列单元读取目标资源,并根据所述目标资源执行目标资源处理动作,以获得与目标资源对应的目标数据并存储所述目标数据。

2.根据权利要求1所述的分布式爬虫系统,其特征在于,所述分布式爬虫处理单元包括调度模块和多个执行模块,多个执行模块均与所述调度模块连接,所述调度模块与爬虫任务配置单元连接,多个所述执行模块均与爬虫结果队列单元连接;

所述调度模块用于接收爬虫任务,并根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列;多个所述执行模块用于从不同的爬虫任务队列中分别获取爬虫任务,根据爬虫任务的爬取内容限定条件获取目标资源。

3.根据权利要求1至2任一项所述的分布式爬虫系统,其特征在于,所述大数据处理单元包括多个处理模块,每一个所述处理模块包括相互连接的读取模块和下载模块,所述读取模块与爬虫结果队列单元连接;

所述读取模块用于从爬虫结果队列单元读取目标资源;所述下载模块用于根据所述目标资源下载与目标资源对应的目标数据,并将下载的目标数据进行存储;其中,所述目标资源包含下载地址。

4.根据权利要求3所述的分布式爬虫系统,其特征在于,所述处理模块还包括与下载模块连接的上传模块,所述上传模块用于获取存储的目标数据,将所述目标数据上传至内网服务平台中。

5.根据权利要求4所述的分布式爬虫系统,其特征在于,所述分布式爬虫系统还包括数据库,所述处理模块还包括与上传模块连接的发送模块,发送模块与数据库连接,所述发送模块用于将来自上传模块的目标数据发送至数据库;其中,多个所述处理模块之间并发执行。

6.根据权利要求1所述的分布式爬虫系统,其特征在于,所述目标资源还包含资源信息,所述资源信息由爬取渠道编码、爬虫类型和唯一识别码依次组成,所述唯一识别码为16位唯一识别码,所述16位唯一识别码由10位时间戳和6位唯一码构成。

7.根据权利要求1所述的分布式爬虫系统,其特征在于,还包括可视化单元,所述爬虫任务配置单元、所述分布式爬虫处理单元、所述爬虫结果队列单元均与所述可视化单元连接,所述可视化单元用于根据爬虫结果队列单元中的目标资源生成爬虫结果报表,当爬虫结果队列单元中每新增有目标资源时针对新增的目标资源对所述爬虫结果报表中的内容进行添加;每隔预设时长将爬虫结果报表进行展示;并接收用户的爬取结果查询指令,根据

该爬取结果查询指令获取爬虫任务对应的爬取结果。

8.一种基于分布式爬虫的数据处理方法,其特征在于,包括:

接收用户配置的多个分别适用于不同平台的爬虫任务,以及爬取内容限定条件,所述爬虫任务包括任务信息,所述任务信息包括平台信息和渠道信息;

根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列;

从爬虫任务队列中获取爬虫任务并根据该爬虫任务的爬取内容限定条件获取目标资源;

根据所述目标资源执行目标资源处理动作,以获得与目标资源对应的目标数据,并存储所述目标数据。

9.根据权利要求8所述的数据处理方法,其特征在于,根据所述目标资源执行目标资源处理动作,包括:

根据所述目标资源下载与目标资源对应的目标数据,将所述目标数据进行存储;所述目标资源包含下载地址;

获取存储的目标数据,将所述目标数据上传至内网服务平台和数据库中。

10.根据权利要求8所述的数据处理方法,其特征在于,还包括:

根据目标资源生成爬虫结果报表,当爬虫结果队列单元中每新增有目标资源时针对新增的目标资源对所述爬虫结果报表中的内容进行添加;

每隔预设时长将爬虫结果报表进行展示;

接收用户的爬取结果查询指令,所述爬取结果查询指令包括用于指示对应的爬取任务的任务信息;

根据该爬取结果查询指令获取爬取任务对应的爬取结果。

11.一种基于分布式爬虫的数据处理装置,其特征在于,包括:

任务接收模块:用于接收用户配置的多个分别适用于不同平台的爬虫任务,以及爬取内容限定条件,所述爬虫任务包括任务信息,所述任务信息包括平台信息和渠道信息;

任务分配模块:用于根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列;

结果获取模块:用于从爬虫任务队列中获取爬虫任务并根据该爬虫任务的爬取内容限定条件获取目标资源;

数据传输模块:用于根据所述目标资源执行目标资源处理动作,以获得与目标资源对应的目标数据,并将存储所述目标数据。

12.一种数据处理设备,其特征在于,包括:存储器以及一个或多个处理器;

所述存储器,用于存储一个或多个程序;

当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现如权利要求8-10任一所述的基于分布式爬虫的数据处理方法。

13.一种包含计算机可执行指令的存储介质,其特征在于,所述计算机可执行指令在由计算机处理器执行时用于执行如权利要求8-10任一所述的基于分布式爬虫的数据处理方法。

可动态扩展的分布式爬虫系统、数据处理方法及装置

技术领域

[0001] 本申请实施例涉及数据爬虫技术领域,尤其涉及一种可动态扩展的分布式爬虫系统、数据处理方法、数据处理装置、数据处理设备及存储介质。

背景技术

[0002] 随着国内智能手机快速普及和移动网络飞速发展,人们可以享受各种各样的信息流,特别是以短视频为首的一批App深受欢迎。如何深挖短视频带来的商业价值,给公司业务带来变现价值成为了一个热门话题。要获取这些媒体内容需要依赖爬虫技术,所以开发一个强大的爬虫系统是这一切业务的基础。

[0003] 目前短视频内容来自于不同平台,爬虫的方法各有不同,使用到的工具也多种多样。目前业界的爬虫系统没有统一的标准,系统的耦合度较高,不便于扩展,增加第三方工具的时候改动较大,会带来额外的工作量。另外一方面,当视频等内容获取后,这些富媒体内容往往比较大,无法采取和普通文本一样的方式进行处理。目前业界的爬虫系统都是负责爬取的模块,并不能处理大量的富媒体文件。

发明内容

[0004] 本申请实施例提供一种可动态扩展的分布式爬虫系统、数据处理方法、装置、设备及存储介质,以实现系统的耦合度低、易扩展的效果。

[0005] 在第一方面,本申请实施例提供了可动态扩展的分布式爬虫系统,包括依次连接的爬虫任务配置单元、分布式爬虫处理单元、爬虫结果队列单元和大数据处理单元;

[0006] 所述爬虫任务配置单元用于供用户配置爬虫任务和爬取内容限定条件,所述爬虫任务包括任务信息;所述分布式爬虫处理单元用于接收爬虫任务,并根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列,从多个爬虫任务队列中分别获取爬虫任务并根据该爬虫任务对应的爬取内容限定条件获取目标资源;所述爬虫结果队列单元用于接收所述目标资源并将所述目标资源进行缓存;所述大数据处理单元用于从爬虫结果队列单元读取目标资源,并根据所述目标资源执行目标资源处理动作,以获得与目标资源对应的目标数据并存储所述目标数据。

[0007] 在第二方面,本申请实施例提供了一种基于分布式爬虫的数据处理方法,包括:

[0008] 接收用户配置的爬虫任务和爬取内容限定条件,所述爬虫任务包括任务信息;

[0009] 根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列;

[0010] 从爬虫任务队列中获取爬虫任务并根据该爬虫任务的爬取内容限定条件获取目标资源;

[0011] 根据所述目标资源执行目标资源处理动作,以获得与目标资源对应的目标数据,并存储所述目标数据。

[0012] 在第三方面,本申请实施例提供了一种基于分布式爬虫的数据处理装置,包括:

[0013] 任务接收模块:用于接收用户配置的爬虫任务和爬取内容限定条件,所述爬虫任

务包括任务信息;

[0014] 任务分配模块:用于根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列;

[0015] 结果获取模块:用于从爬虫任务队列中获取爬虫任务并根据该爬虫任务的爬取内容限定条件获取目标资源;

[0016] 数据传输模块:用于根据所述目标资源执行目标资源处理动作,以获得与目标资源对应的目标数据,并将存储所述目标数据。在第四方面,本申请实施例提供了一种数据处理设备,包括:存储器以及一个或多个处理器;

[0017] 所述存储器,用于存储一个或多个程序;

[0018] 当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现如第一方面所述的基于分布式爬虫的数据处理方法。

[0019] 在第四方面,本申请实施例提供了一种包含计算机可执行指令的存储介质,所述计算机可执行指令在由计算机处理器执行时用于执行如第一方面所述的基于分布式爬虫的数据处理方法。

[0020] 本申请实施例通过优化系统,在爬虫任务配置中隔离爬虫任务配置和条件配置,在执行爬虫任务时对不同任务信息的爬虫任务列入不同的任务队列等待,并且多线程分别执行不停任务队列的爬虫任务;通过降低系统内部的耦合度、提高了可动态扩展性能,避免内存泄露,大大提高开发效率,以及提高系统的吞吐量。

附图说明

[0021] 图1是本申请实施例提供的一种可动态扩展的分布式爬虫系统的结构示意图;

[0022] 图2是本申请实施例提供的另一种可动态扩展的分布式爬虫系统的结构示意图;

[0023] 图3是本申请实施例提供的另一种可动态扩展的分布式爬虫系统的结构示意图;

[0024] 图4是本申请实施例提供的另一种可动态扩展的分布式爬虫系统的结构示意图;

[0025] 图5是本申请实施例提供的另一种可动态扩展的分布式爬虫系统的结构示意图;

[0026] 图6是本申请实施例提供的另一种可动态扩展的分布式爬虫系统的结构示意图;

[0027] 图7是本申请实施例提供的一种基于分布式爬虫的数据处理方法的流程图;

[0028] 图8是本申请实施例提供的另一种基于分布式爬虫的数据处理方法的流程图;

[0029] 图9是本申请实施例提供的另一种基于分布式爬虫的数据处理方法的流程图;

[0030] 图10是本申请实施例提供的另一种基于分布式爬虫的数据处理方法的流程图;

[0031] 图11是本申请实施例提供的一种基于分布式爬虫的数据处理方法的整体流程示意图;

[0032] 图12是本申请实施例提供的一种基于分布式爬虫的数据处理装置的结构示意图;

[0033] 图13是本申请实施例提供的一种数据处理设备的结构示意图。

具体实施方式

[0034] 为了使本申请的目的、技术方案和优点更加清楚,下面结合附图对本申请具体实施例作进一步的详细描述。可以理解的是,此处所描述的具体实施例仅仅用于解释本申请,而非对本申请的限定。另外还需要说明的是,为了便于描述,附图中仅示出了与本申请相关

的部分而非全部内容。在更加详细地讨论示例性实施例之前应当提到的是,一些示例性实施例被描述成作为流程图描绘的处理或方法。虽然流程图将各项操作(或步骤)描述成顺序的处理,但是其中的许多操作可以被并行地、并发地或者同时实施。此外,各项操作的顺序可以被重新安排。当其操作完成时所述处理可以被终止,但是还可以具有未包括在附图中的附加步骤。所述处理可以对应于方法、函数、规程、子例程、子程序等等。

[0035] 本申请实施例提供了可动态扩展的分布式爬虫系统、数据处理方法、数据处理装置、数据处理设备及存储介质。本申请实施例通过优化系统,在爬虫任务配置中隔离爬虫任务配置和条件配置,在执行爬虫任务时对不同任务信息的爬虫任务列入不同的任务队列等待,并且多线程分别执行不停任务队列的爬虫任务;通过降低系统内部的耦合度、提高了可动态扩展性能,避免内存泄露,大大提高开发效率,以及提高系统的吞吐量。

[0036] 下面分别进行详细说明。

[0037] 图1给出了本申请实施例提供的可动态扩展的分布式爬虫系统的结构示意图。如图1所示,一种可动态扩展的分布式爬虫系统包括爬虫任务配置单元101、分布式爬虫处理单元102、爬虫结果队列单元103和大数据处理单元104,其中爬虫任务配置单元101、分布式爬虫处理单元102、爬虫结果队列单元103和大数据处理单元104依次连接。

[0038] 本实施例中,所述爬虫任务配置单元101用于供用户配置爬虫任务和爬取内容限定条件,所述爬虫任务包括任务信息。本申请实施例在系统的上游设置任务配置系统,提供可供用户进行操作和配置的界面。爬虫任务配置单元101提供给用户分别配置爬虫任务和爬取内容限定条件的框架,使得爬虫任务和限定条件之间两个行为解耦,能够动态筛选出符合行业需求的数据。

[0039] 结合图2所示,图2示出了更进一步的可动态扩展的分布式爬虫系统的结构示意图。如图2所示,在本实施例中,爬虫任务配置单元101包括相互连接的任务配置模块1011和条件配置模块1012,所述任务配置模块1011用于供用户配置多个分别适用于不同平台的爬虫任务,所述条件配置模块1012用于供用户配置爬取内容限定条件。

[0040] 在本申请实施例中,根据业务需求,用户在任务配置单元101的任务配置模块1011中,配置适用于不同平台的爬虫任务。开发人员可以独立开发不同平台的爬虫脚本,易于扩展不同平台的爬虫。容易理解的是,一个平台通常包含有多个不同的爬取渠道,比如通过个人页、hashtag和关键词等搜索方式,均是可以实施的爬取渠道。作为一种应用场景,本申请实施例用户在爬虫任务配置单元101中配置了应用在短视频播放平台、论文发表平台的两种不同的爬虫任务。对应于短视频播放平台的爬虫任务,又分为获取短视频播放平台中数据不同渠道的爬虫任务,例如短视频播放平台的搜索渠道、短视频播放平台的评论渠道。同样的,对应于论文发表平台的爬虫任务,又根据论文发表平台的不同获取渠道不同而有区别。每一个爬虫任务都包含任务信息。任务信息包括平台信息和渠道信息,平台信息用于指示不同的平台,渠道信息用于指示不同的渠道。

[0041] 在本申请实施例中,根据情景需要,需要配置给爬虫任务的爬取内容限定条件。用户可通过页面直接配置,可以控制爬虫的爬取行为。由于任务配置单元101包括任务配置模块1011和条件配置模块1012,也即是将爬取任务的配置和爬取内容限定条件的配置两种行为解耦开。在一种实施方式中,用户可配置了爬虫任务之后,又通过条件配置模块1012继续配置与先前所配置的爬虫任务相关、匹配的爬取内容限定条件。在另外一个实施示例中,由

于先前已经在条件配置模块配置了大量的爬取内容限定条件,这些爬取内容限定条件中包含了适用于多种平台、多种渠道的不同爬取任务的爬取内容限定条件。用户在配置了爬虫任务之后,可以不通过条件配置模块1012配置爬取内容限定条件,而是通过条件配置模块1012本身存在的多个爬取内容限定条件中获取到匹配的爬取内容限定条件。

[0042] 本实施例中的分布式爬虫处理单元102具体用于接收爬虫任务,并根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列,从多个爬虫任务队列中分别获取爬虫任务并根据该爬虫任务对应的爬取内容限定条件获取目标资源。

[0043] 本申请实施例的分布式爬虫处理单元102从接收到爬虫任务,将爬虫任务队列分发。第一步为接收爬虫任务,并根据爬虫任务的任务信息的不同,将爬虫任务分发到不同的任务队列。例如当前共有5组爬虫任务为新配置,其中爬虫任务1的的任务信息指示为A平台,B渠道,爬虫任务2的任务信息指示为A平台,C渠道,爬虫任务3的任务信息指示为B平台,A渠道,爬虫任务4的任务信息指示为C平台,B渠道,爬虫任务5的任务信息指示为A平台,B渠道。则将该爬虫任务1和爬虫任务5都分发到与A平台B渠道对应的任务队列,而爬虫任务2分发A平台、C渠道的任务队列,爬虫任务3分发到B平台、A渠道的任务队列,爬虫任务4分发到C平台、B渠道的任务队列。可见,虽然两个爬虫任务的任务信息都指向同一个平台,但是如果爬取渠道不同,也分发到不停的任务队列,如果仅仅爬取渠道相同,但对应不同平台,也同样分发到不同的任务队列。只有当平台和渠道都完全一致时,才分发在同一个任务队列,如上述示例爬取任务1和爬取任务5。

[0044] 不同平台和不停渠道的爬虫会依赖各种第三方插件。现有使用的爬虫框架耦合度高,不易于加入第三方插件。本申请实施例所提供的分布式爬虫处理单元102使用分布式任务调度框架,只提供调度爬虫任务的能力,没有与任何第三方耦合,爬虫的依赖统一为开源的爬虫发行版本,各个业务爬虫可以根据自己所需而增加依赖。通过任务分布式调度和依赖环境的解耦,可以让爬虫系统快速扩展第三方工具,从而避免烦人的环境依赖冲突。

[0045] 在爬虫任务执行方面,本申请实施例中将各个不同的爬虫任务分发给单独的执行器执行,执行器的数量可以根据任务数量调整,适应不同渠道的任务负载均衡。

[0046] 图3示出了另一种可动态扩展的分布式爬虫系统的结构示意图。本实施例中结合图1和图3,在本实施例中,分布式爬虫处理单元102包括调度模块1021和多个执行模块1022,多个执行模块均与所述调度模块连接,所述调度模块与爬虫任务配置单元连接,多个所述执行模块均与爬虫结果队列单元连接。调度模块1021用于接收爬虫任务,并根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列;多个所述执行模块1022用于从不同的爬虫任务队列中分别获取爬虫任务,根据爬虫任务的爬取内容限定条件获取目标资源。

[0047] 执行模块1022即为前述的执行器。本申请实施例中,调度模块1021只接收爬虫任务,对爬虫任务进行任务队列的分发,而不对爬虫任务进行执行。在整个分布式爬虫处理单元102中,可能同时存在数量较多的任务队列,因此配置有多个执行模块1022。多个执行模块同时执行多个不同的爬虫任务。在本实施例中,执行模块1022的数量可能与任务队列的数量相同,可能大于任务队列的数量,可能小于任务队列的数量。当执行模块1022的数量与任务队列的数量相同,表明执行模块1022被配置为与任务队列一一对应,此时例如有任务队列5个,则执行模块1022也有五个,五个执行模块1022同时执行,在不同的进程并发工作。

当执行模块1022的数量大于任务队列的数量,意味着可能多个同一个位于任务队列中的爬虫任务对应一个执行模块1022,这种情况通常可能是某一个任务队列有较多的爬虫任务,爬虫任务数量多,执行压力大,因此配置多个执行模块1022来执行该任务队列的爬虫任务,缓解该爬虫队列的任务,并保持各任务队列之间的负载均衡。例如,任务队列A当前排队的爬虫任务有25组,而任务队列B当前排队的爬虫任务有10组,任务队列C当前排队的爬虫任务有8组,任务队列A的爬虫任务数量比其他两组任务队列中爬虫任务数量的两倍还要多,此时可以在任务队列B和任务队列C均是分配给一个执行模块1022的情况下,优先给任务队列A分发三个执行模块1022。当执行模块1022的数量小于任务队列的数量,表明有些任务队列中可能爬虫任务数量少,执行压力小,并且其他任务队列的爬虫任务数量也不大,整体爬虫压力较小,可以选择等一个任务队列的爬虫任务执行后再执行另一个任务队列的爬虫任务,或者几个任务队列的爬虫任务交替间隔的执行。例如当前分布式爬虫处理单元102中分配有3个执行模块1022,同时存在的任务队列有5个,任务队列1有2个爬虫任务,其他四个任务队列都是1个爬虫任务,可见爬虫压力小。

[0048] 本申请实施例每隔执行模块1022都有进程监控工具进行监控,保证执行模块1022持续运作。在执行完任务后,使用同一的协议格式,将目标资源发送至缓存队列等待下一步的处理。本实施例将任务调度行为抽象为公共平台,而平台本身不负责业务逻辑。调度模块1021只负责管理调度信息,支持可视化和简单动态管理调度信息。执行模块1022负责执行任务逻辑。执行1022模块专注于任务的执行,开发和维护更加简单高效。调度模块1021和执行模块1022分开解耦,能提高框架的稳定性和可扩展性。

[0049] 本申请实施例的所述爬虫结果队列单元103用于接收所述目标资源并将所述目标资源进行缓存。本实施例中应用到消息队列,作为一种进程间或者线程间的通信方式,消息发送方将一份数据发送到消息队列,然后消费方再从消息队列取走该份数据。当上下游的消费能力不一致时,能作为一个消息的缓冲区,等下游有能力处理时再将消息取走。本申请在爬虫结果队列单元103使用消息队列,将爬取的结果缓存起来,等大数据处理单元104陆续取走爬虫数据,并进行下一步的处理。

[0050] 本申请实施例所提供的大数据处理单元104用于从爬虫结果队列单元读取目标资源,并根据所述目标资源执行目标资源处理动作,以获得与目标资源对应的目标数据并存储所述目标数据。

[0051] 数据作为事件流,无边界数据集的抽象,并会随着时间源源不断地加入进来。流式处理就是指实时地处理一个或多个事件流,读取数据集并对它们处理并生成结果。整个过程是持续不断的。流式处理框架提供嵌入到数据流中的数据接口,用户可以通过它自由的处理单流或者多流,并保持一致性和容错。同时用户可以注册事件时间和处理时间的回调处理,以实现复杂的计算逻辑。本实施的大数据处理单元104中提供流式数据处理框架。结合图1和图4,本申请实施例所提供的另一种可动态扩展的分布式爬虫系统包括爬虫任务配置单元101、分布式爬虫处理单元102、爬虫结果队列单元103和大数据处理单元104,其中爬虫任务配置单元101、分布式爬虫处理单元102、爬虫结果队列单元103和大数据处理单元104依次连接。

[0052] 其中,参考图4,所述大数据处理单元104包括多个处理模块,每个所述处理模块包括相互连接的读取模块1041和下载模块1042,所述读取模块1041与爬虫结果队列单元103

连接;

[0053] 所述读取模块1041用于从爬虫结果队列单元103读取目标资源;所述下载模块1042用于根据所述目标资源下载与目标资源对应的目标数据,并将下载的目标数据进行存储;其中,所述目标资源包含下载地址。

[0054] 目标资源为原始资源,如果不将目标资源进行下载,将目标资源放在网络平台,往往有时效性,有的甚至几个小时就失效了。为了防止资源失效,因此本实施例中将目标资源进行下载。目标资源包含下载地址,因此根据下载地址,能够准确获取与目标资源对应的目标数据。

[0055] 作为优选的实施方式,大数据处理单元104的处理模块还包括与下载模块1042连接的上传模块1043,所述上传模块1043用于获取存储的目标数据,将所述目标数据上传至内网服务平台中。内网服务平台是服务于设置爬虫任务的公司内部服务平台,可以对目标数据永久性存储,并且可以方便内部处理、方便公司的业务线重复利用目标数据。

[0056] 如图5所示,本实施例提供了另一种可动态扩展的分布式爬虫系统的结构示意图,本实施例中,所述分布式爬虫系统还包括数据库105,所述大数据处理单元104的处理模块还包括与上传模块1043连接的发送模块1044,发送模块1044与数据库105连接,所述发送模块1044用于将来自上传模块1043的目标数据发送至数据库105;其中,下载模块1042、上传模块1043和发送模块1044之间顺序执行,但不同的处理模块之间并发执行。将大数据处理单元104包含多个处理模块,实际上是为了形成多条下载、上传、发送的线程。

[0057] 上传模块1043将下载模块1042所下载的目标数据进行上传,而发送模块1044将目标数据推送至数据库105。更为具体的,实际上将发送模块1044发送至数据库105之前,还会先经过一个审核接口,通过审核接口对目标数据进行审核,审核合格之后再传输至数据库105。

[0058] 作为优选的实施方式,在本申请实施例中,所述目标资源还包含资源信息,所述资源信息由爬取渠道编码、爬虫类型和唯一识别码依次组成,所述唯一识别码由时间戳和唯一码构成。

[0059] 为了保证目标资源不会重复下载而浪费资源空间,所以将目标资源设置为包含资源信息用于对目标资源进行身份信息的识别,且作为唯一身份标志。本申请设置资源信息由爬取渠道编码、爬虫类型和唯一识别码依次组成,即是爬取渠道编码+爬虫类型+唯一识别码的组成方式构成了本申请的资源信息。作为优选的唯一识别码为16位自增id码,唯一识别码进一步由时间戳和唯一码构成。时间戳是指获取目标资源当前时间戳,由16位自增id码中的前10位构成,后六位是自增id,能保证每日新增爬取量小于一百万的情况下,每隔爬虫的目标资源的资源信息都是唯一的。

[0060] 结合图1和图6所示,本实施例提供的可动态扩展的分布式爬虫系统还包括可视化单元106,所述爬虫任务配置单元101、所述分布式爬虫处理单元102、所述爬虫结果队列单元103均与所述可视化单元106连接,所述可视化单元106用于根据爬虫结果队列单元103中的目标资源生成爬虫结果报表,当爬虫结果队列单元103中每新增有目标资源时针对新增的目标资源对所述爬虫结果报表中的内容进行添加;每隔预设时长将爬虫结果报表进行展示;并接收用户的爬取结果查询指令,根据该爬取结果查询指令获取爬虫任务对应的爬取结果。由于获取爬取任务的爬取结果的过程中,爬取结果包括用户配置了爬虫任务之后,对

应的爬虫任务所处在的任意一个节点,包括处于任务队列等待被执行,或者已经下载了目标资源。另外,用户的爬取结果查询指令可通过爬虫任务配置单元101提供用户的输入端口进行输入。本申请以爬虫任务为维度,提供任务级别的数据查阅,用于通过点击查询某个爬虫任务,能快速响应,从而得知该任务对应的爬取结果。

[0061] 图11示出了本申请实施例提供的基于分布式爬虫的数据处理方法的整体流程示意图。包括用户登录爬虫任务后台,在爬虫任务配置单元根据不同的平台和渠道配置爬虫任务、爬虫内容限定条件。之后将爬虫任务推送至任务消费队列。在推送过程中可能推送成功也可能推送失败,推送成功则直接推送至任务消费队列,推送失败则返回上一步,继续接收用户的配置。这里的任务消费队列是指分布式爬虫处理单元的调度模块,并且分布式爬虫处理单元设置有多个执行模块作为爬虫单位分别执行不同的爬虫任务。执行爬虫任务之后获得对应的目标资源均存储在缓存队列,也即是存放在爬虫结果队列单元中。之后大数据处理单元主动获取目标资源,根据目标资源进行下载、上传、推送的一系列动作。在大数据处理单元中,对目标资源在同一个线程中进行下载、上传、推送是顺序执行的,但在不同的线程之间是并发执行的。也即是,假设当前时间戳有目标资源1、2、3,大数据处理单元对目标资源1、2、3分别进行下载,下一个时间戳又有新的目标资源4,此时对又对目标资源1、2、3分别进行上传,同时对目标资源4进行下载。也即是,在大数据处理单元中,对下载、上传、推送可同时并行为上千个任务。

[0062] 具体请参考图7至图10,图7至图10给出了本申请实施例提供的基于分布式爬虫的数据处理方法的流程图,本申请实施例提供的基于分布式爬虫的数据处理方法可以由可动态扩展的分布式爬虫系统或者基于分布式爬虫的数据处理装置来执行,该于分布式爬虫的数据处理装可以通过硬件和/或软件的方式实现,并集成在计算机设备中。

[0063] 如图7所示,该基于分布式爬虫的数据处理方法包括:

[0064] 701:接收用户配置的爬虫任务和爬取内容限定条件,所述爬虫任务包括任务信息。

[0065] 702:根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列。

[0066] 703:从爬虫任务队列中获取爬虫任务并根据该爬虫任务的爬取内容限定条件获取目标资源。

[0067] 704:根据所述目标资源执行目标资源处理动作,以获得与目标资源对应的目标数据,并存储所述目标数据。

[0068] 本申请实施例可以配置不同平台不同渠道的爬虫任务,以及可以配置给爬虫任务爬取内容限定条件。

[0069] 作为更优选的实施方式,图8示出了另外一种基于分布式爬虫的数据处理方法,包括:

[0070] 801:接收用户配置的爬虫任务和爬取内容限定条件,所述爬虫任务包括任务信息。

[0071] 802根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列。

[0072] 803:从爬虫任务队列中获取爬虫任务并根据该爬虫任务的爬取内容限定条件获

取目标资源;将所述目标资源进行缓存。

[0073] 804:从缓存中读取目标资源。

[0074] 805:根据所述目标资源下载与目标资源对应的目标数据,将所述目标数据进行存储;所述目标资源包含下载地址。

[0075] 806:获取存储的目标数据,将所述目标数据上传至内网服务平台和数据库中。

[0076] 本实施例中,对获取的目标资源先进行队列缓存,以适应数据量大的情况,等消费方有能力时再取走目标资源。

[0077] 本实施例公开了对目标数据的处理包括下载、上传,并推送至数据库中,其中下载、上传和推送是并发执行的流程。

[0078] 本实施例中,所述目标资源还包含资源信息,所述资源信息由爬取渠道编码、爬虫类型和唯一识别码依次组成,所述唯一识别码为16位唯一识别码,所述16位唯一识别码由10位时间戳和6位唯一码构成。

[0079] 为了保证目标资源不会重复下载而浪费资源空间,所以将目标资源设置为包含资源信息用于对目标资源进行身份信息的识别,且作为唯一身份标志。本申请设置资源信息由爬取渠道编码、爬虫类型和唯一识别码依次组成,即是爬取渠道编码+爬虫类型+唯一识别码的组成方式构成了本申请的资源信息。作为优选的唯一识别码为16位唯一识别码,也即是16位自增id码,唯一识别码进一步由时间戳和唯一码构成。时间戳是指获取目标资源当前时间戳,由16位自增id码中的前10位构成,后6位是自增id。

[0080] 图9示出了另外一种基于分布式爬虫的数据处理方法,如图9所示,该数据处理方法包括:

[0081] 901:接收用户配置的爬虫任务和爬取内容限定条件,所述爬虫任务包括任务信息。

[0082] 902:根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列。

[0083] 903:从爬虫任务队列中获取爬虫任务并根据该爬虫任务的爬取内容限定条件获取目标资源。

[0084] 904:根据所述目标资源执行目标资源处理动作,以获得与目标资源对应的目标数据,并存储所述目标数据。

[0085] 905:根据目标资源生成爬虫结果报表,当爬虫结果队列单元中每新增有目标资源时针对新增的目标资源对所述爬虫结果报表中的内容进行添加。

[0086] 906:每隔预设时长将爬虫结果报表进行展示。

[0087] 本实施例相当于提供给用户可视化的结果展示,预设时长例如为一天,或者半天,或者两小时,等,将预设时长内的爬虫结果统计为一个报表,方便后续查询和直观的展示。

[0088] 进一步的,如图10所示,基于分布式爬虫的数据处理方法还可进一步包括:

[0089] 1007:接收用户的爬取结果查询指令,所述爬取结果查询指令包括用于指示对应的爬取任务的任务信息;

[0090] 1008:根据该爬取结果查询指令获取爬取任务对应的爬取结果。

[0091] 本申请以爬虫任务为维度,提供任务级别的数据查阅,用于通过点击查询某个爬虫任务,能快速响应,从而得知该任务对应的爬取结果。

[0092] 本申请实施例所提供的基于分布式爬虫的数据处理方法实质流程与本申请提出的可动态拓展的分布式爬虫系统相同,因此本实施中对数据处理方法的原理等实质性介绍较为省略,具体可参考本申请实施例所提供的可动态拓展的分布式爬虫系统。

[0093] 如图12所示,本申请实施例还提供一种基于分布式爬虫的数据处理装置,包括任务接收模块1201、任务分配模块1202、结果获取模块1203和数据传输模块1204。

[0094] 其中,任务接收模块1201用于接收用户配置的爬虫任务和爬取内容限定条件,所述爬虫任务包括任务信息。任务分配模块1202用于根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列。结果获取模块1203用于从爬虫任务队列中获取爬虫任务并根据该爬虫任务的爬取内容限定条件获取目标资源。数据传输模块1203用于根据所述目标资源执行目标资源处理动作,以获得与目标资源对应的目标数据,并将存储所述目标数据。

[0095] 如图13所示,本申请实施例还提供一种数据处理设备,包括:存储器1301以及一个或多个处理器1302;所述存储器1301,用于存储一个或多个程序;当所述一个或多个程序被所述一个或多个处理器1302执行,使得所述一个或多个处理器实现如本申请所述的基于分布式爬虫的数据处理方法。

[0096] 本申请实施例还提供一种包含计算机可执行指令的存储介质,所述计算机可执行指令在由计算机处理器执行时用于执行如上述实施例提供的基于分布式爬虫的数据处理方法。

[0097] 当然,本申请实施例所提供的一种包含计算机可执行指令的存储介质,其计算机可执行指令不限于如上所述的基于分布式爬虫的数据处理方法,还可以执行本申请任意实施例所提供的基于分布式爬虫的数据处理方法中的相关操作。

[0098] 上述仅为本申请的较佳实施例及所运用的技术原理。本申请不限于这里所述的特定实施例,对本领域技术人员来说能够进行的各种明显变化、重新调整及替代均不会脱离本申请的保护范围。因此,虽然通过以上实施例对本申请进行了较为详细的说明,但是本申请不仅仅限于以上实施例,在不脱离本申请构思的情况下,还可以包括更多其他等效实施例,而本申请的范围由权利要求的范围决定。

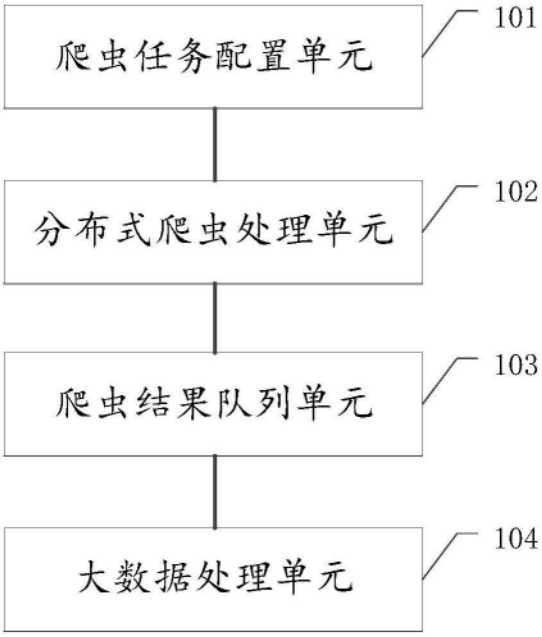

爬虫任务配置单元
101
分布式爬虫处理单元
102
爬虫结果队列单元
103
大数据处理单元
104

图1

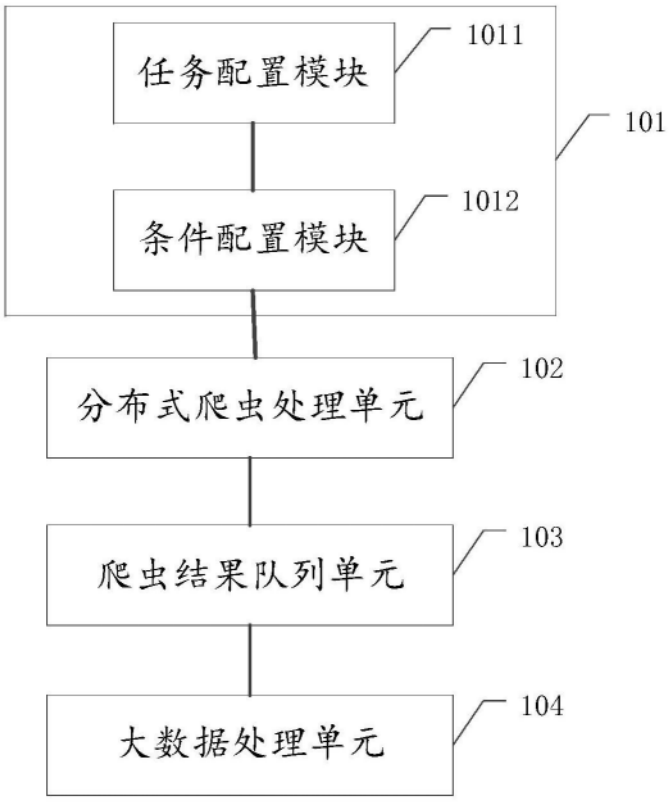

任务配置模块
1011
条件配置模块
1012
101
分布式爬虫处理单元
102
爬虫结果队列单元
103
大数据处理单元
104

图2

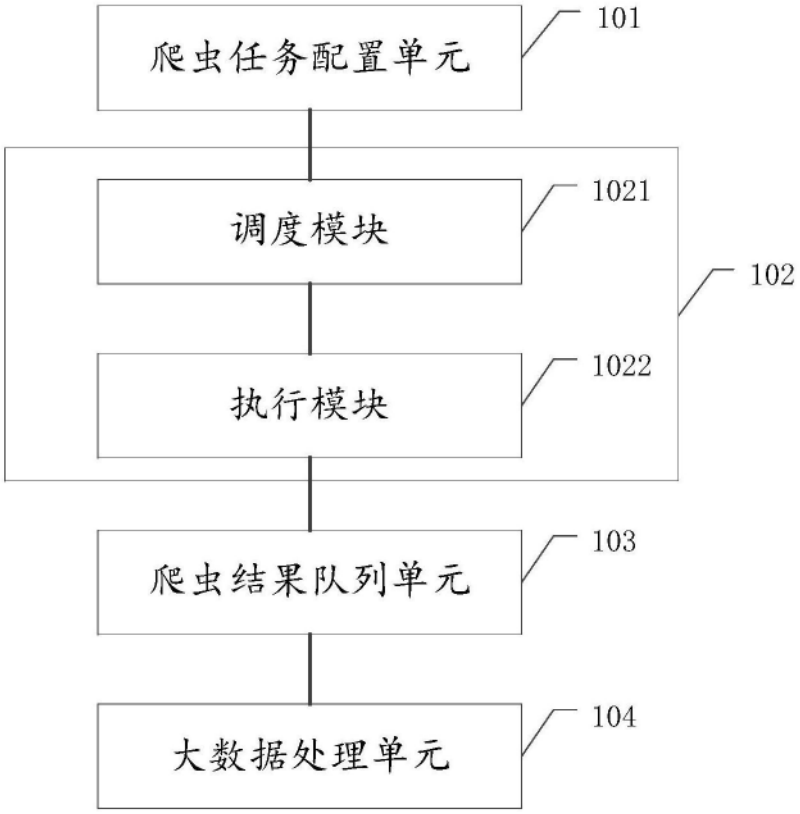

爬虫任务配置单元
101
调度模块
1021
102
执行模块
1022
爬虫结果队列单元
103
大数据处理单元
104

图3

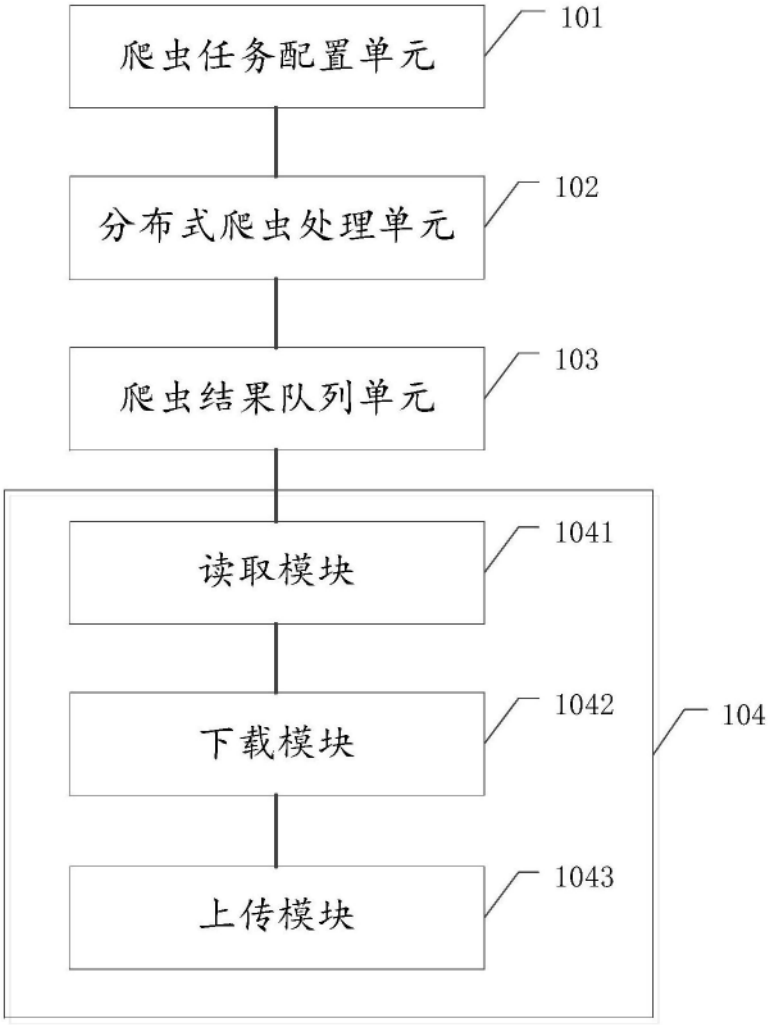

爬虫任务配置单元
101
分布式爬虫处理单元
102
爬虫结果队列单元
103
读取模块
1041
下载模块
1042
上传模块
1043
104

图4

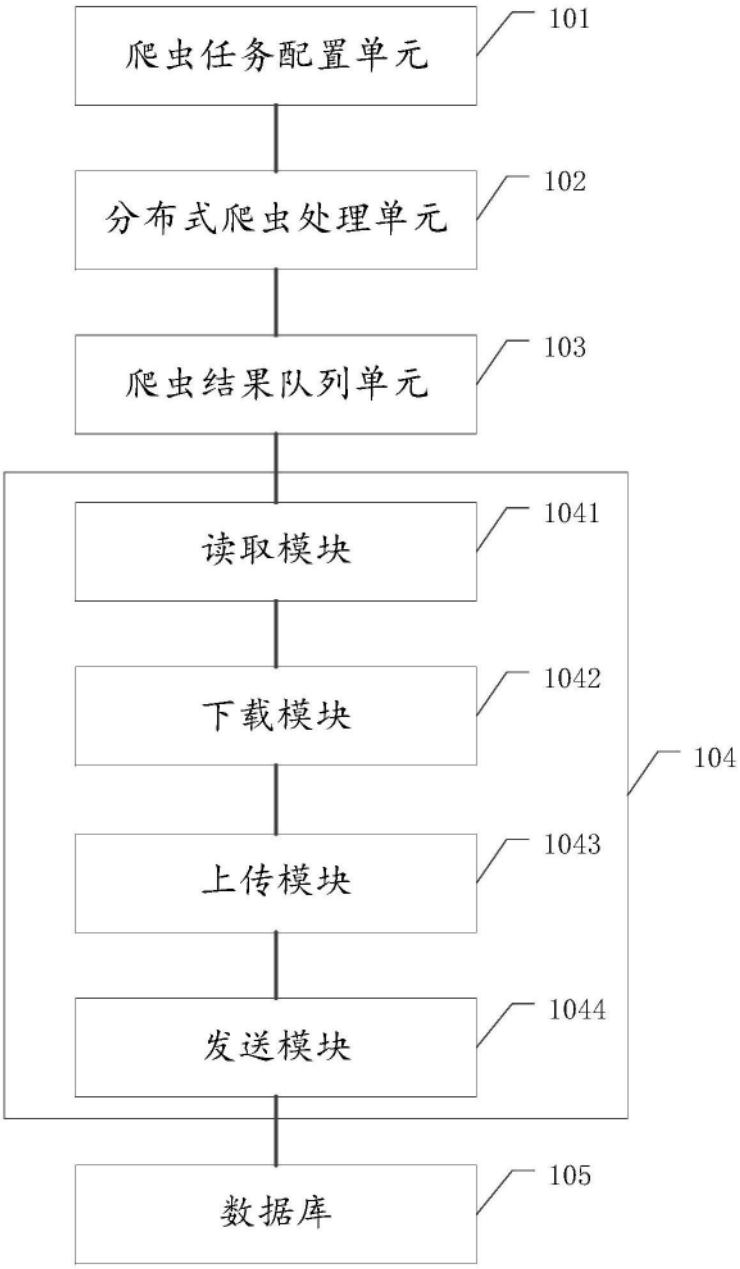

爬虫任务配置单元
101
分布式爬虫处理单元
102
爬虫结果队列单元
103
读取模块
1041
下载模块
1042
上传模块
1043
发送模块
1044
104
数据库
105

图5

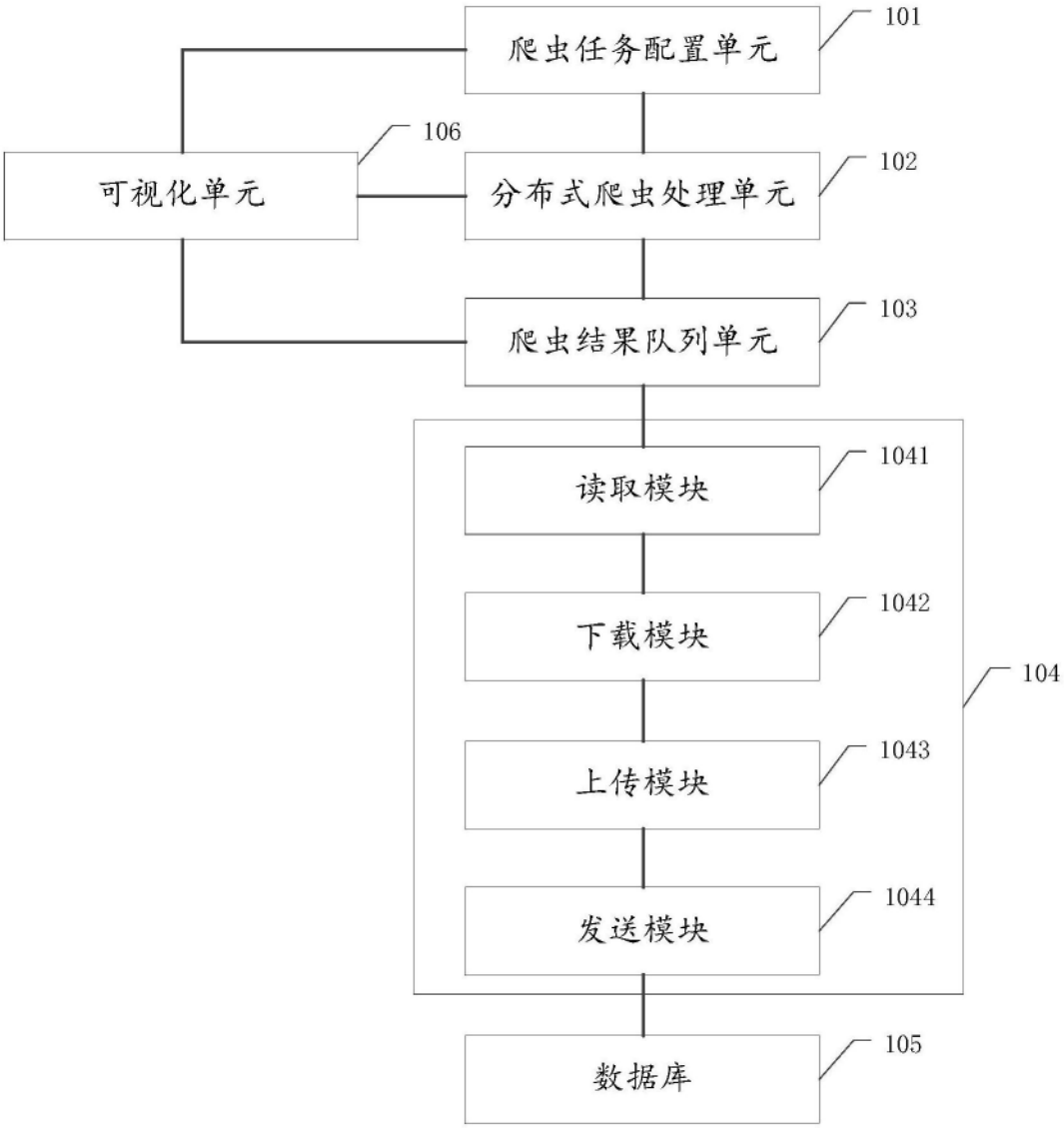

爬虫任务配置单元
101
可视化单元
106
分布式爬虫处理单元
102
爬虫结果队列单元
103
读取模块
1041
下载模块
1042
上传模块
1043
发送模块
1044
104
数据库
105

图6

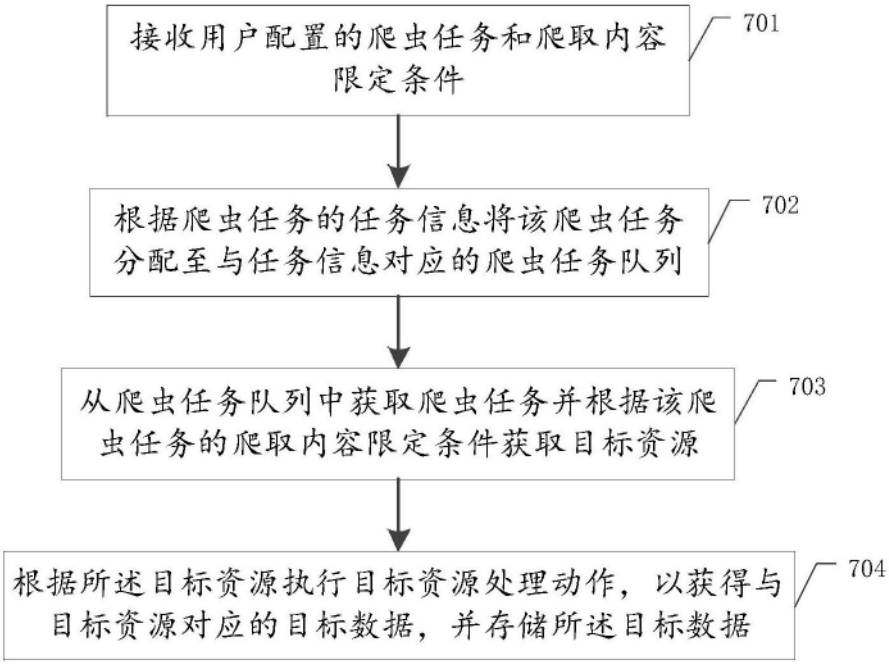

接收用户配置的爬虫任务和爬取内容限定条件
701
根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列
702
从爬虫任务队列中获取爬虫任务并根据该爬虫任务的爬取内容限定条件获取目标资源
703
根据所述目标资源执行目标资源处理动作，以获得与目标资源对应的目标数据，并存储所述目标数据
704

图7

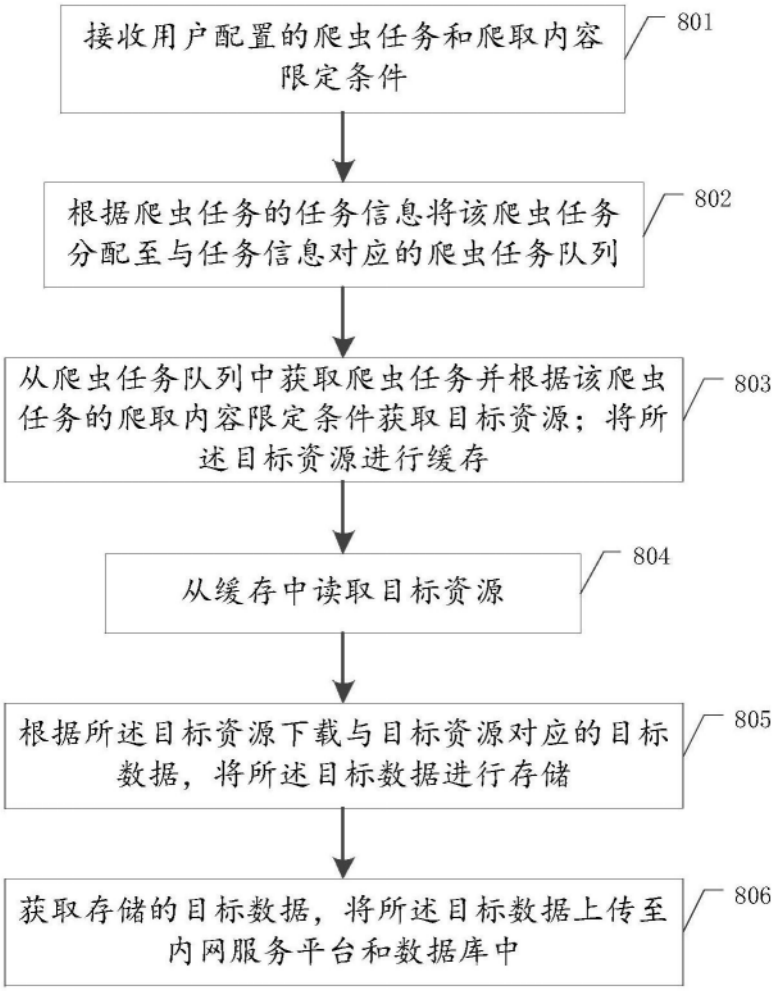

接收用户配置的爬虫任务和爬取内容限定条件
801
根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列
802
从爬虫任务队列中获取爬虫任务并根据该爬虫任务的爬取内容限定条件获取目标资源；将所述目标资源进行缓存
803
从缓存中读取目标资源
804
根据所述目标资源下载与目标资源对应的目标数据，将所述目标数据进行存储
805
获取存储的目标数据，将所述目标数据上传至内网服务平台和数据库中
806

图8

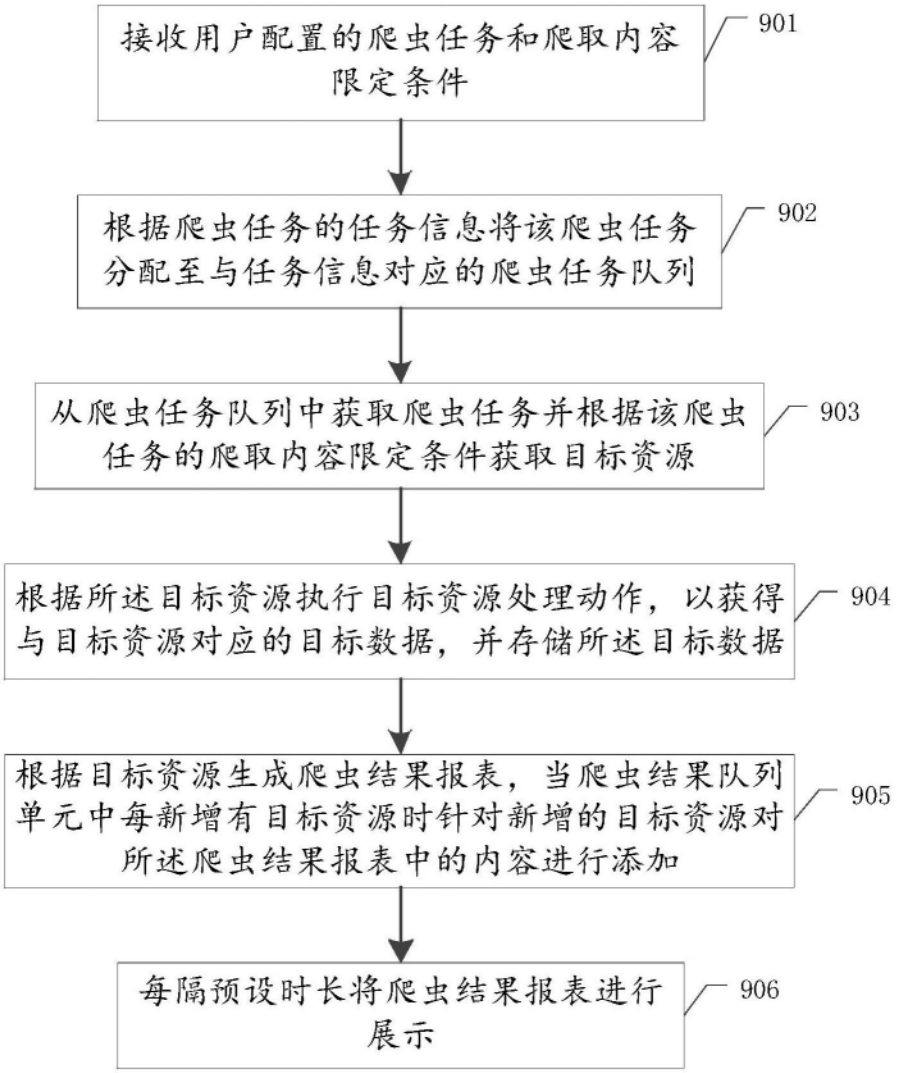

接收用户配置的爬虫任务和爬取内容限定条件
901
根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列
902
从爬虫任务队列中获取爬虫任务并根据该爬虫任务的爬取内容限定条件获取目标资源
903
根据所述目标资源执行目标资源处理动作，以获得与目标资源对应的目标数据，并存储所述目标数据
904
根据目标资源生成爬虫结果报表，当爬虫结果队列单元中每新增有目标资源时针对新增的目标资源对所述爬虫结果报表中的内容进行添加
905
每隔预设时长将爬虫结果报表进行展示
906

图9

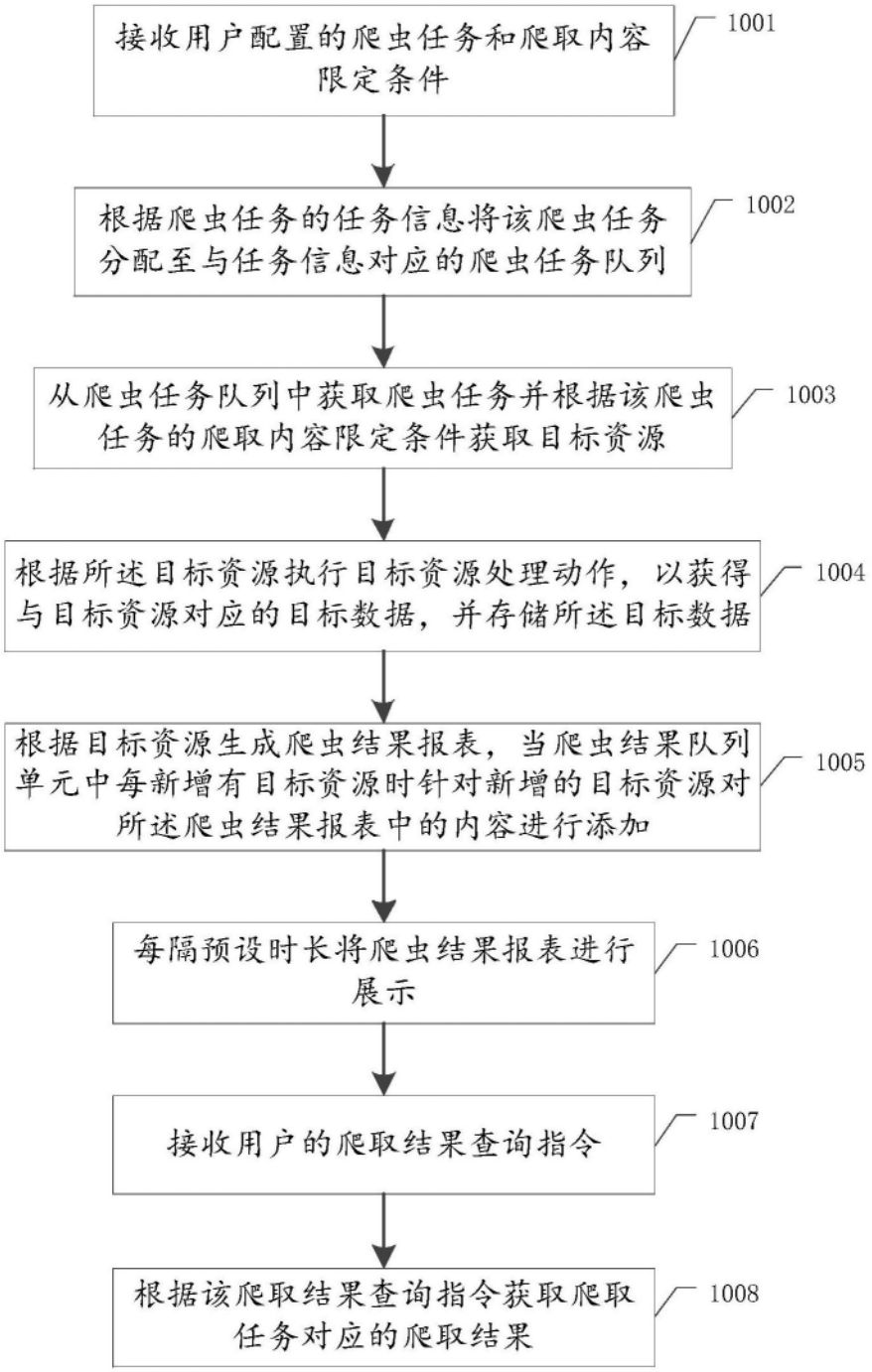

接收用户配置的爬虫任务和爬取内容限定条件
1001
根据爬虫任务的任务信息将该爬虫任务分配至与任务信息对应的爬虫任务队列
1002
从爬虫任务队列中获取爬虫任务并根据该爬虫任务的爬取内容限定条件获取目标资源
1003
根据所述目标资源执行目标资源处理动作，以获得与目标资源对应的目标数据，并存储所述目标数据
1004
根据目标资源生成爬虫结果报表，当爬虫结果队列单元中每新增有目标资源时针对新增的目标资源对所述爬虫结果报表中的内容进行添加
1005
每隔预设时长将爬虫结果报表进行展示
1006
接收用户的爬取结果查询指令
1007
根据该爬取结果查询指令获取爬取任务对应的爬取结果
1008

图10

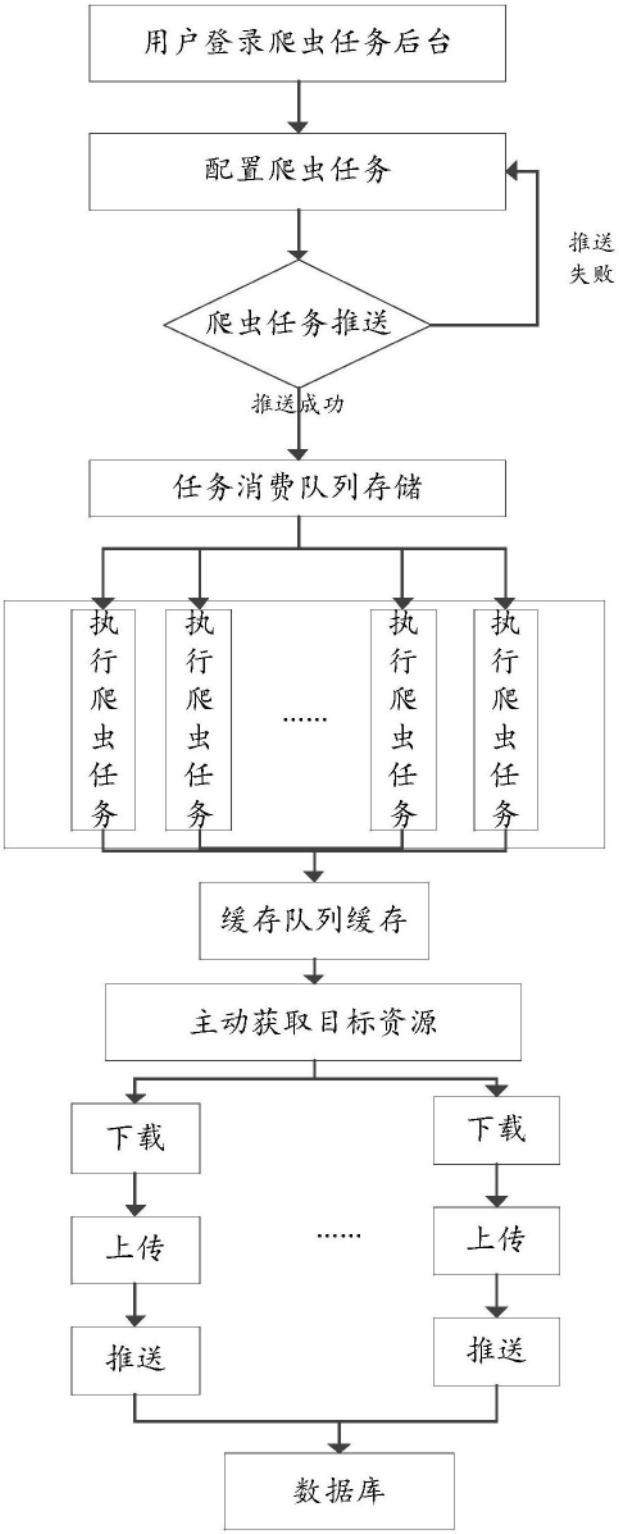

用户登录爬虫任务后台
配置爬虫任务
推送失败
爬虫任务推送
推送成功
任务消费队列存储
执行爬虫任务
执行爬虫任务
.....
执行爬虫任务
执行爬虫任务
缓存队列缓存
主动获取目标资源
下载
上传
推送
.....
下载
上传
推送
数据库

图11

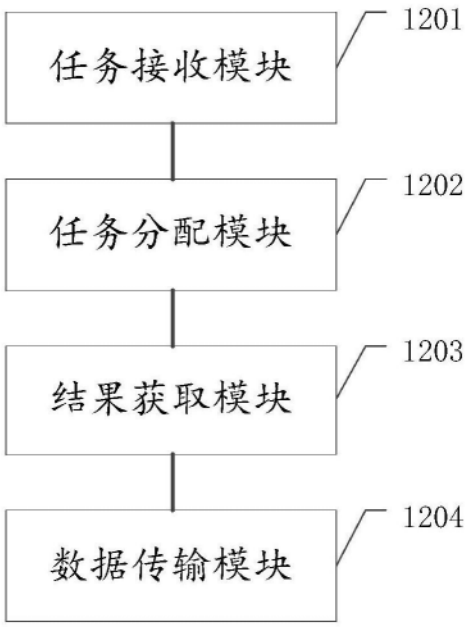

任务接收模块
1201
任务分配模块
1202
结果获取模块
1203
数据传输模块
1204

图12

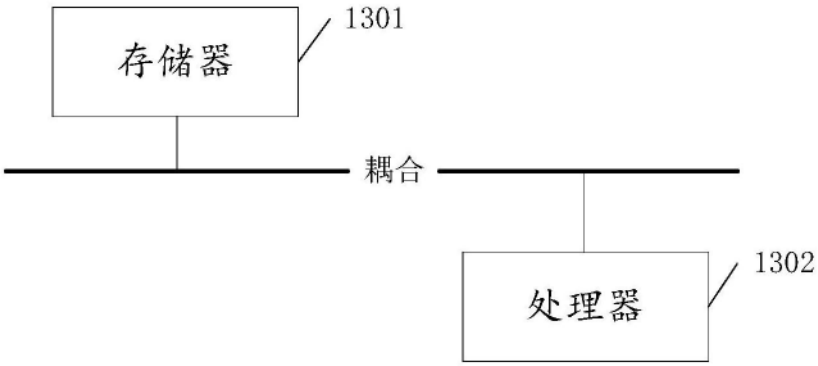

存储器
1301
耦合
处理器
1302

图13