



(12) 发明专利

(10) 授权公告号 CN 110389976 B

(45) 授权公告日 2022. 04. 12

(21) 申请号 201810331883.2

G06F 16/2453 (2019.01)

(22) 申请日 2018.04.13

G06F 9/48 (2006.01)

(65) 同一申请的已公布的文献号
申请公布号 CN 110389976 A

(56) 对比文件

CN 107657425 A, 2018.02.02

EP 1081593 A2, 2001.03.07

(43) 申请公布日 2019.10.29

WO 2004079911 A3, 2005.04.28

(73) 专利权人 北京京东乾石科技有限公司
地址 100176 北京市大兴区北京经济技术
开发区科创十一街18号院2号楼19层
A1905室

CN 107436844 A, 2017.12.05

宋浩. 异构网络资源调度方法的设计与实现.《中国优秀硕士学位论文全文数据库》.2015,
(第4期),

(72) 发明人 童月红 殷伟

审查员 刘高鸣

(74) 专利代理机构 中原信达知识产权代理有限
责任公司 11219
代理人 张一军 张效荣

(51) Int. Cl.

G06F 16/25 (2019.01)

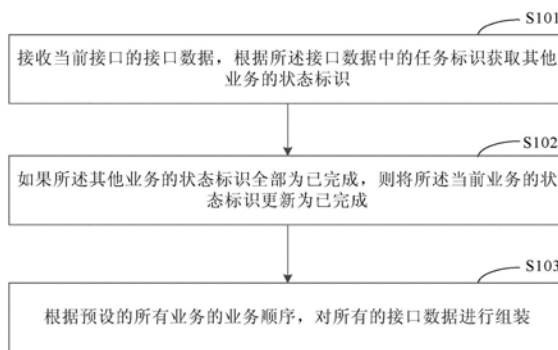
权利要求书2页 说明书7页 附图4页

(54) 发明名称

一种多接口数据的调度方法和装置

(57) 摘要

本发明公开了一种多接口数据的调度方法和装置,涉及计算机技术领域。该方法的一具体实施方式包括:接收当前接口的接口数据,根据接口数据中的任务标识获取其他业务的状态标识;其中,其他业务为任务标识对应的所有业务中,除当前接口对应的当前业务之外的业务;如果其他业务的状态标识全部为已完成,则将当前业务的状态标识更新为已完成;根据预设的所有业务的业务顺序,对所有的接口数据进行组装。本申请接收无序接口数据,在任务标识对应的其他业务的状态标识全部为已完成时,更新当前业务的状态标识,并对所有接口数据进行组装,既能兼容无序接口调用,又能按照业务顺序进行调度处理,减轻了数据库查询的压力,降低了网络资源消耗。



1. 一种多接口数据的调度方法,其特征在于,包括:

接收当前接口的接口数据,根据所述接口数据中的任务标识获取其他业务的状态标识;其中,所述其他业务为所述任务标识对应的所有业务中,除所述当前接口对应的当前业务之外的业务;

如果所述其他业务的状态标识全部为已完成,则将所述当前业务的状态标识更新为已完成;根据预设的所有业务的业务顺序,对所有的接口数据进行组装;

如果所述其他业务的状态标识存在未完成,则将所述当前业务的状态标识更新为已完成;将下一接口的接口数据作为所述当前接口的接口数据,以接收所述当前接口的接口数据。

2. 根据权利要求1所述的方法,其特征在于,所述根据所述接口数据中的任务标识获取其他业务的状态标识的步骤之前,还包括:

构建任务主档数据;其中所述任务主档数据包括所述任务标识和对应的所有业务的状态标识;

所述根据所述接口数据中的任务标识获取其他业务的状态标识,包括:

根据所述接口数据中的任务标识查询任务主档数据,以获取其他业务的状态标识。

3. 根据权利要求2所述的方法,其特征在于,所述根据所述接口数据中的任务标识查询任务主档数据的步骤,包括:

根据所述接口数据中的任务标识和所述当前业务的业务类型,生成待汇总对象;

根据所述待汇总对象的任务标识查询任务主档数据。

4. 一种多接口数据的调度装置,其特征在于,包括:

获取模块,用于接收当前接口的接口数据,根据所述接口数据中的任务标识获取其他业务的状态标识;其中,所述其他业务为所述任务标识对应的所有业务中,除所述当前接口对应的当前业务之外的业务;

更新模块,用于如果所述其他业务的状态标识全部为已完成,则将所述当前业务的状态标识更新为已完成;

组装模块,用于根据预设的所有业务的业务顺序,对所有的接口数据进行组装;

更新重复模块,用于如果所述其他业务的状态标识存在未完成,则将所述当前业务的状态标识更新为已完成;以及将下一接口的接口数据作为所述当前接口的接口数据,以接收所述当前接口的接口数据。

5. 根据权利要求4所述的装置,其特征在于,所述装置还包括:构建模块,用于构建任务主档数据;其中所述任务主档数据包括所述任务标识和对应的所有业务的状态标识;

所述获取模块,还用于:根据所述接口数据中的任务标识查询任务主档数据,以获取其他业务的状态标识。

6. 根据权利要求5所述的装置,其特征在于,所述获取模块,还用于:

根据所述接口数据中的任务标识和所述当前业务的业务类型,生成待汇总对象;以及根据所述待汇总对象的任务标识查询任务主档数据。

7. 一种电子设备,其特征在于,包括:

一个或多个处理器;

存储装置,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1-3中任一所述的方法。

8.一种计算机可读介质,其上存储有计算机程序,其特征在于,所述程序被处理器执行时实现如权利要求1-3中任一所述的方法。

一种多接口数据的调度方法和装置

技术领域

[0001] 本发明涉及计算机领域,尤其涉及一种多接口数据的调度方法和装置。

背景技术

[0002] 一个系统的业务处理需要依赖于多个外部接口的接口数据,并且对各个接口的接口数据的反馈顺序有一定要求。在收到多个无序的接口数据后,需要按照顺序对接口数据进行处理。比如无人仓的机器人码垛任务,任务中心需要按顺序判断码垛预处理任务完成、发票打印成功和商品到达码垛位后,下发机器人码垛任务。

[0003] 由于网络异常、并发或外界各系统的性能不同,即使接口调用的顺序正常,在到达系统时也可能出现顺序错乱,在该情况下,系统需要兼容任何一个接口数据的无序到达,并按照系统要求的顺序进行处理。现有技术中有两种处理方式,方式一:按顺序接收来自外部接口的接口数据,若后发送的接口数据先到达系统,则系统不接受并抛出异常,等待下次重新调用该接口;方式二:无序接收来自各接口的接口数据并保存,使用定时器或者Worker(任务对象)异步扫描保存的数据,直到接收了该任务的所有接口数据后,定时器的定时任务才完成。

[0004] 在实现本发明过程中,发明人发现现有技术中至少存在如下问题:方式一当前置接口未到达,针对后置接口抛出异常,从而触发后置接口的重新调用,由于重复调用的频率不同,调用次数也不定,在一定程度上增加了网络资源的无效消耗,降低了资源利用率,接口反馈不够友好;方式二需要用定时器或者线程频繁扫描数据库,以确认多个接口的接口数据已到达,增加了数据库的读写负担,延长了业务处理时间。

发明内容

[0005] 有鉴于此,本发明实施例提供一种多接口数据的调度方法和装置,通过接收无序的接口数据,在任务标识对应的其他业务的状态标识全部为已完成时,更新当前业务的状态标识,并对所有接口数据进行组装,使得本申请既能够兼容无序接口调用又能按照业务顺序进行调度处理,减少了无效重复处理,减轻了数据库查询的压力,降低了网络资源的消耗。

[0006] 实现上述目的,根据本发明实施例的一个方面,提供了一种多接口数据的调度方法。

[0007] 本发明实施例的一种多接口数据的调度方法,包括:接收当前接口的接口数据,根据所述接口数据中的任务标识获取其他业务的状态标识;其中,所述其他业务为所述任务标识对应的所有业务中,除所述当前接口对应的当前业务之外的业务;如果所述其他业务的状态标识全部为已完成,则将所述当前业务的状态标识更新为已完成;根据预设的所有业务的业务顺序,对所有的接口数据进行组装。

[0008] 可选地,所述根据所述接口数据中的任务标识获取其他业务的状态标识的步骤之前,还包括:构建任务主档数据;其中所述任务主档数据包括所述任务标识和对应的所有业

务的状态标识;所述根据所述接口数据中的任务标识获取其他业务的状态标识,包括:根据所述接口数据中的任务标识查询任务主档数据,以获取其他业务的状态标识。

[0009] 可选地,所述根据所述接口数据中的任务标识查询任务主档数据的步骤,包括:根据所述接口数据中的任务标识和所述当前业务的业务类型,生成待汇总对象;根据所述待汇总对象的任务标识查询任务主档数据。

[0010] 可选地,所述方法还包括:如果所述其他业务的状态标识存在未完成,则将所述当前业务的状态标识更新为已完成;将下一接口的接口数据作为所述当前接口的接口数据,以接收所述当前接口的接口数据。

[0011] 为实现上述目的,根据本发明实施例的一个方面,提供了一种多接口数据的调度装置。

[0012] 本发明实施例的一种多接口数据的调度装置,包括:获取模块,用于接收当前接口的接口数据,根据所述接口数据中的任务标识获取其他业务的状态标识;其中,所述其他业务为所述任务标识对应的所有业务中,除所述当前接口对应的当前业务之外的业务;更新模块,用于如果所述其他业务的状态标识全部为已完成,则将所述当前业务的状态标识更新为已完成;组装模块,用于根据预设的所有业务的业务顺序,对所有的接口数据进行组装。

[0013] 可选地,所述装置还包括:构建模块,用于构建任务主档数据;其中所述任务主档数据包括所述任务标识和对应的所有业务的状态标识;所述获取模块,还用于:根据所述接口数据中的任务标识查询任务主档数据,以获取其他业务的状态标识。

[0014] 可选地,所述获取模块,还用于:根据所述接口数据中的任务标识和所述当前业务的业务类型,生成待汇总对象;以及根据所述待汇总对象的任务标识查询任务主档数据。

[0015] 可选地,所述装置还包括:更新重复模块,用于如果所述其他业务的状态标识存在未完成,则将所述当前业务的状态标识更新为已完成;以及将下一接口的接口数据作为所述当前接口的接口数据,以接收所述当前接口的接口数据。

[0016] 为实现上述目的,根据本发明实施例的再一方面,提供了一种电子设备。

[0017] 本发明实施例的一种电子设备,包括:一个或多个处理器;存储装置,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本发明实施例的一种多接口数据的调度方法。

[0018] 为实现上述目的,根据本发明实施例的再一方面,提供了一种计算机可读介质。

[0019] 本发明实施例的一种计算机可读介质,其上存储有计算机程序,所述程序被处理器执行时实现本发明实施例的一种多接口数据的调度方法。

[0020] 上述发明中的一个实施例具有如下优点或有益效果:通过接收无序的接口数据,在接口数据中的任务标识对应的其他业务的状态标识全部为已完成时,更新当前业务的状态标识,并对所有接口数据进行组装,使得本申请在正常或者高并发场景下,既能够兼容无序接口调用又能按照业务顺序进行调度处理,减轻了数据库查询的压力,降低了网络资源的消耗,接口反馈友好;N个接口调用分别对应不同接口数据和不同的状态标识,有效防止在高并发下对相同数据进行更新的风险;N个接口调用,最多N次调度即可判定任务是否已完成,减少了无效重复处理。

[0021] 上述的非惯用的可选方式所具有的进一步效果将在下文中结合具体实施方式加

以说明。

附图说明

- [0022] 附图用于更好地理解本发明,不构成对本发明的不当限定。其中:
- [0023] 图1是根据本发明实施例的多接口数据的调度方法的主要步骤的示意图;
- [0024] 图2是根据本发明实施例的多接口数据的调度方法的主要流程示意图;
- [0025] 图3是根据本发明实施例的多接口数据的调度装置的主要单元的示意图;
- [0026] 图4是本发明实施例可以应用于其中的示例性系统架构图;
- [0027] 图5是适用于来实现本发明实施例的电子设备的计算机装置的结构示意图。

具体实施方式

[0028] 以下结合附图对本发明的示范性实施例做出说明,其中包括本发明实施例的各种细节以助于理解,应当将它们认为仅仅是示范性的。因此,本领域普通技术人员应当认识到,可以对这里描述的实施例做出各种改变和修改,而不会背离本发明的范围和精神。同样,为了清楚和简明,以下的描述中省略了对公知功能和结构的描述。

[0029] 图1是根据本发明实施例的多接口数据的调度方法的主要步骤的示意图。如图1所示,本发明实施例的多接口数据的调度方法,主要包括如下步骤:

[0030] 步骤S101:接收当前接口的接口数据,根据所述接口数据中的任务标识获取其他业务的状态标识;其中,所述其他业务为所述任务标识对应的所有业务中,除所述当前接口对应的当前业务之外的业务。一个任务包括至少一个业务,各业务通过接口实现,预先为该任务分配唯一的任务标识。系统向外部接口发送调用请求,并接收接口反馈的接口数据,该接口数据中包括任务标识。

[0031] 步骤S102:如果所述其他业务的状态标识全部为已完成,则将所述当前业务的状态标识更新为已完成。每个任务的任务标识和该任务包含的所有业务的状态标识保存在任务主档数据中,根据任务标识查询任务主档数据,判断所述其他业务的状态标识是否全部为已完成,如果全部已完成,则将所述当前业务的状态标识更新为已完成。

[0032] 步骤S103:根据预设的所有业务的业务顺序,对所有的接口数据进行组装。所述业务顺序根据业务需求来确定,根据业务顺序和该任务所需求的数据,从所有接口的接口数据中获取部分数据,并汇总组装成该任务所需的数据字段和数据报文。

[0033] 图2是根据本发明实施例的多接口数据的调度方法的主要步骤的示意图。如图2所示,本发明实施例的多接口数据的调度方法,主要包括如下步骤:

[0034] 步骤S201:发送调用请求到当前接口,接收来自当前接口的接口数据,对所述当前接口的接口数据进行校验,将检验通过的接口数据存储到数据库。其中,所述接口数据中包括任务标识。当接收到以数据报文形式发送来的接口数据时,首先进行防重处理,即根据数据报文的UUID(Universally Unique Identifier,通用唯一识别码)判断是否已经接收过该接口数据,如果没有接收过,再对接口数据进行校验;如果已经接收过,则不接收该接口数据,并反馈数据重复信息。对接口数据进行校验可以是对接口数据是否异常的判定,如果校验通过,则保存接口数据;如果校验未通过,则反馈接口数据异常的信息。

[0035] 步骤S202:根据所述任务标识和所述当前接口对应的当前业务类型,生成待汇总

对象。所述待汇总对象是带有任务标识和当前业务类型的中间消息对象,该对象用于关联整个任务,作为后续查询的触发条件。生成待汇总对象后,反馈接口调用成功的信息。

[0036] 步骤S203:根据所述待汇总对象的任务标识查询任务主档数据,以判断所述任务标识对应的其他业务的状态标识是否全部已完成;如果存在未完成的状态标识,则执行步骤S204;如果全部已完成,则执行步骤S205。其中,所述任务主档数据需预先构建,包括任务标识和所述任务标识对应的所有业务的状态标识。如果系统已经收到某个接口的接口数据,则会将该接口对应业务的状态标识更改为已完成。

[0037] 步骤S204:将所述任务主档数据中,所述当前业务对应的状态标识更新为已完成,将下一接口的接口数据作为当前接口的接口数据,执行步骤S201。如果任务主档数据中其他业务的状态标识存在未完成,说明还未收到该未完成的业务所对应接口的接口数据,则需要等待接收该接口的接口数据,并将该接口数据作为当前接口数据,从步骤S201开始重复执行本发明实施例的多接口数据的调度方法。

[0038] 步骤S205:将所述任务主档数据中,所述当前业务对应的状态标识更新为已完成,执行步骤S206。如果任务主档数据中其他业务的状态标识均为已完成,说明系统已经接收到该任务的多个接口的接口数据,则更新当前业务对应的状态标识,后续即可汇总组装所有的接口数据。

[0039] 步骤S206:根据预设的所有业务的业务顺序,对存储的所有的接口数据进行组装。该业务顺序根据业务需求来确定,得到的组装结果用于下一数据处理流程。对存储的所有的接口数据进行组装的具体过程为:根据该任务所需求的数据,从所有接口的接口数据中获取部分数据,并汇总组装成该任务所需的数据字段和数据报文。

[0040] 以机器人码垛任务为例,系统调用判断码垛预处理任务完成、发票打印成功和商品到达码垛位这三个业务的接口的先后顺序不定,每次收到接口数据后,修改该接口对应业务的状态标识为已完成,N次调用最多生成N个待汇总数据对象,触发N次对其他业务的状态标识是否全部已完成的判定。当最后一个接口的状态标识更改为已完成时,标志该任务终结。

[0041] 通过本发明实施例的多接口数据的调度方法可以看出,通过接收无序的接口数据,在接口数据中的任务标识对应的其他业务的状态标识全部为已完成时,更新当前业务的状态标识,并对所有接口数据进行组装,使得本申请在正常或者高并发场景下,既能够兼容无序接口调用又能按照业务顺序进行调度处理,减轻了数据库查询的压力,降低了网络资源的消耗,接口反馈友好;N个接口调用分别对应不同接口数据和不同的状态标识,有效防止在高并发下对相同数据进行更新的风险;N个接口调用,最多N次调度即可判定任务是否已完成,减少了无效重复处理。

[0042] 图3是根据本发明实施例的多接口数据的调度装置的主要模块的示意图。如图3所示,本发明实施例的多接口数据的调度装置300,主要包括:

[0043] 获取模块301,用于接收当前接口的接口数据,根据所述接口数据中的任务标识获取其他业务的状态标识;其中,所述其他业务为所述任务标识对应的所有业务中,除所述当前接口对应的当前业务之外的业务。一个任务包括至少一个业务,各业务通过接口实现,预先为该任务分配唯一的任务标识。系统向外部接口发送调用请求,并接收接口反馈的接口数据,该接口数据中包括任务标识。

[0044] 更新模块302,用于如果所述其他业务的状态标识全部为已完成,则将所述当前业务的状态标识更新为已完成。每个任务的任务标识和该任务包含的所有业务的状态标识保存在任务主档数据中,根据任务标识查询任务主档数据,判断所述其他业务的状态标识是否全部为已完成,如果全部已完成,则将所述当前业务的状态标识更新为已完成。

[0045] 组装模块303,用于根据预设的所有业务的业务顺序,对所有的接口数据进行组装。所述业务顺序根据业务需求来确定,根据业务顺序和该任务所需求的数据,从所有接口的接口数据中获取部分数据,并汇总组装成该任务所需的数据字段和数据报文。

[0046] 另外,本发明实施例的多接口数据的调度装置300还可以包括:构建模块和更新重复模块(图3中未示出)。其中,构建模块,用于构建任务主档数据;其中,所述任务主档数据包括所述任务标识和对应的所有业务的状态标识。更新重复模块,用于如果所述其他业务的状态标识存在未完成,则将所述当前业务的状态标识更新为已完成;以及将下一接口的接口数据作为所述当前接口的接口数据,以接收所述当前接口的接口数据。

[0047] 从以上描述可以看出,通过接收无序的接口数据,在接口数据中的任务标识对应的其他业务的状态标识全部为已完成时,更新当前业务的状态标识,并对所有接口数据进行组装,使得本申请在正常或者高并发场景下,既能够兼容无序接口调用又能按照业务顺序进行调度处理,减轻了数据库查询的压力,降低了网络资源的消耗,接口反馈友好;N个接口调用分别对应不同接口数据和不同的状态标识,有效防止在高并发下对相同数据进行更新的风险;N个接口调用,最多N次调度即可判定任务是否已完成,减少了无效重复处理。

[0048] 图4示出了可以应用本发明实施例的多接口数据的调度方法或多接口数据的调度装置的示例性系统架构400。

[0049] 如图4所示,系统架构400可以包括终端设备401、402、403,网络404和服务器405。网络404用以在终端设备401、402、403和服务器405之间提供通信链路的介质。网络404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。

[0050] 用户可以使用终端设备401、402、403通过网络404与服务器405交互,以接收或发送消息等。终端设备401、402、403上可以安装有各种通讯客户端应用,例如购物类应用、网页浏览器应用、搜索类应用、即时通信工具、邮箱客户端、社交平台软件等。

[0051] 终端设备401、402、403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、膝上型便携计算机和台式计算机等等。

[0052] 服务器405可以是提供各种服务的服务器,例如对用户利用终端设备401、402、403所产生的点击事件提供支持的后台管理服务器。后台管理服务器可以对接收到的点击数据、文本内容等数据进行分析等处理,并将处理结果(例如目标推送信息、产品信息--仅为示例)反馈给终端设备。

[0053] 需要说明的是,本申请实施例所提供的多接口数据的调度方法一般由服务器405执行,相应地,多接口数据的调度装置一般设置于服务器405中。

[0054] 应该理解,图4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。

[0055] 根据本发明的实施例,本发明还提供了一种电子设备和一种计算机可读介质。

[0056] 本发明的电子设备包括:一个或多个处理器;存储装置,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

现本发明实施例的一种多接口数据的调度方法。

[0057] 本发明的计算机可读介质,其上存储有计算机程序,所述程序被处理器执行时实现本发明实施例的一种多接口数据的调度方法。

[0058] 下面参考图5,其示出了适用于来实现本发明实施例的电子设备的计算机系统500的结构示意图。图5示出的电子设备仅仅是一个示例,不应对本发明实施例的功能和使用范围带来任何限制。

[0059] 如图5所示,计算机系统500包括中央处理单元(CPU)501,其可以根据存储在只读存储器(ROM)502中的程序或者从存储部分508加载到随机访问存储器(RAM)503中的程序而执行各种适当的动作和处理。在RAM 503中,还存储有计算机系统500操作所需的各种程序和数据。CPU 501、ROM 502以及RAM 503通过总线504彼此相连。输入/输出(I/O)接口505也连接至总线504。

[0060] 以下部件连接至I/O接口505:包括键盘、鼠标等的输入部分506;包括诸如阴极射线管(CRT)、液晶显示器(LCD)等以及扬声器等的输出部分507;包括硬盘等的存储部分508;以及包括诸如LAN卡、调制解调器等的网络接口卡的通信部分509。通信部分509经由诸如因特网的网络执行通信处理。驱动器510也根据需要连接至I/O接口505。可拆卸介质511,诸如磁盘、光盘、磁光盘、半导体存储器等等,根据需要安装在驱动器510上,以便于从其上读出的计算机程序根据需要被安装入存储部分508。

[0061] 特别地,根据本发明公开的实施例,上文主要步骤图描述的过程可以被实现为计算机软件程序。例如,本公开的实施例包括一种计算机程序产品,其包括承载在计算机可读介质上的计算机程序,该计算机程序包含用于执行主要步骤图所示的方法的程序代码。在这样的实施例中,该计算机程序可以通过通信部分509从网络上被下载和安装,和/或从可拆卸介质511被安装。在该计算机程序被中央处理单元(CPU)501执行时,执行本发明的系统中限定的上述功能。

[0062] 需要说明的是,本发明所示的计算机可读介质可以是计算机可读信号介质或者计算机可读存储介质或者是上述两者的任意组合。计算机可读存储介质例如可以是——但不限于——电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者任意以上的组合。计算机可读存储介质的更具体的例子可以包括但不限于:具有一个或多个导线的电连接、便携式计算机磁盘、硬盘、随机访问存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑磁盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。在本发明中,计算机可读存储介质可以是任何包含或存储程序的有形介质,该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。而在本发明中,计算机可读的信号介质可以包括在基带中或者作为载波一部分传播的数据信号,其中承载了计算机可读的程序代码。这种传播的数据信号可以采用多种形式,包括但不限于电磁信号、光信号或上述的任意合适的组合。计算机可读的信号介质还可以是计算机可读存储介质以外的任何计算机可读介质,该计算机可读介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。计算机可读介质上包含的程序代码可以用任何适当的介质传输,包括但不限于:无线、电线、光缆、RF等等,或者上述的任意合适的组合。

[0063] 附图中的流程图和框图,图示了按照本发明各种实施例的系统、方法和计算机程

序产品的可能实现的体系架构、功能和操作。在这点上,流程图或框图中的每个方框可以代表一个模块、程序段、或代码的一部分,上述模块、程序段、或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意,在有些作为替换的实现中,方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如,两个接连地表示的方框实际上可以基本并行地执行,它们有时也可以按相反的顺序执行,这依所涉及的功能而定。也要注意的是,框图或流程图中的每个方框、以及框图或流程图中的方框的组合,可以用执行规定的功能或操作的专用的基于硬件的系统来实现,或者可以用专用硬件与计算机指令的组合来实现。

[0064] 描述于本发明实施例中涉及到的模块可以通过软件的方式实现,也可以通过硬件的方式来实现。所描述的模块也可以设置在处理器中,例如,可以描述为:一种处理器包括获取模块、更新模块和组装模块。其中,这些模块的名称在某种情况下并不构成对该模块本身的限定,例如,获取模块还可以被描述为“接收当前接口的接口数据,根据所述接口数据中的任务标识获取其他业务的状态标识的模块”。

[0065] 作为另一方面,本发明还提供了一种计算机可读介质,该计算机可读介质可以是上述实施例中描述的设备中所包含的;也可以是单独存在,而未装配入该设备中。上述计算机可读介质承载有一个或者多个程序,当上述一个或者多个程序被一个该设备执行时,使得该设备包括:接收当前接口的接口数据,根据所述接口数据中的任务标识获取其他业务的状态标识;其中,所述其他业务为所述任务标识对应的所有业务中,除所述当前接口对应的当前业务之外的业务;如果所述其他业务的状态标识全部为已完成,则将所述当前业务的状态标识更新为已完成;根据预设的所有业务的业务顺序,对所有的接口数据进行组装。

[0066] 从以上描述可以看出,通过接收无序的接口数据,在接口数据中的任务标识对应的其他业务的状态标识全部为已完成时,更新当前业务的状态标识,并对所有接口数据进行组装,使得本申请在正常或者高并发场景下,既能够兼容无序接口调用又能按照业务顺序进行调度处理,减轻了数据库查询的压力,降低了网络资源的消耗,接口反馈友好;N个接口调用分别对应不同接口数据和不同的状态标识,有效防止在高并发下对相同数据进行更新的风险;N个接口调用,最多N次调度即可判定任务是否已完成,减少了无效重复处理。

[0067] 上述产品可执行本发明实施例所提供的方法,具备执行方法相应的功能模块和有益效果。未在本实施例中详尽描述的技术细节,可参见本发明实施例所提供的方法。

[0068] 上述具体实施方式,并不构成对本发明保护范围的限制。本领域技术人员应该明白的是,取决于设计要求和因素,可以发生各种各样的修改、组合、子组合和替代。任何在本发明的精神和原则之内所作的修改、等同替换和改进等,均应包含在本发明保护范围之内。

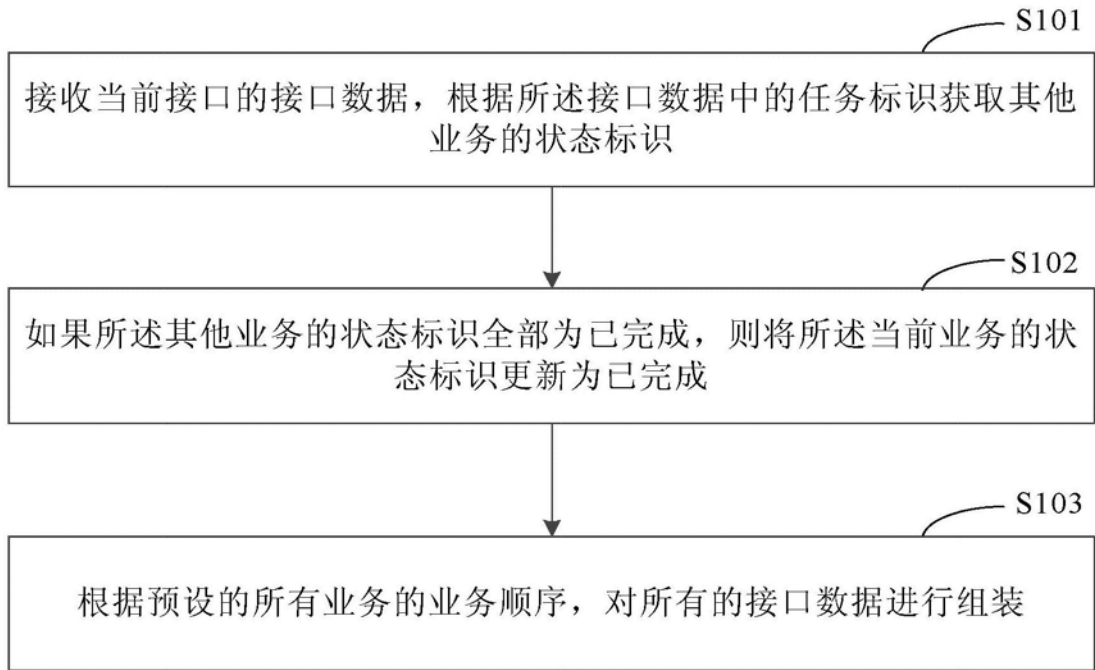


图1

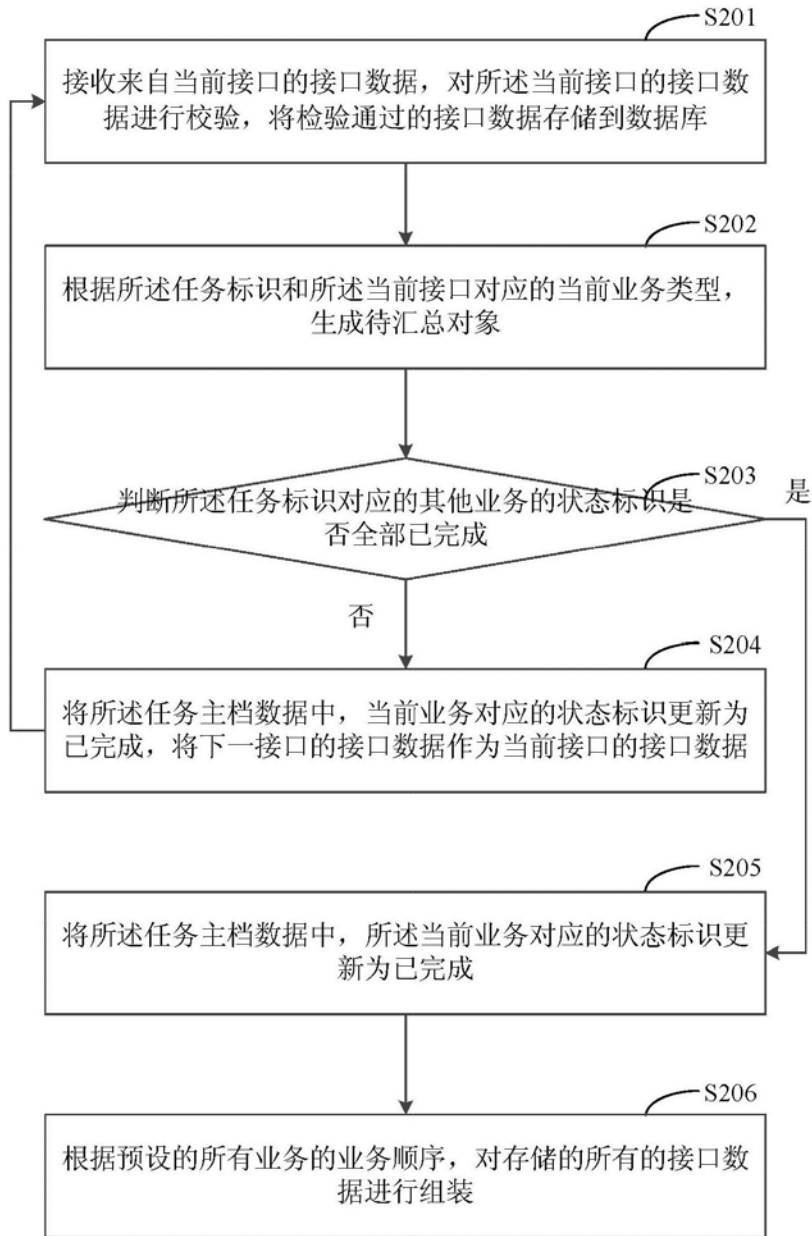


图2



图3

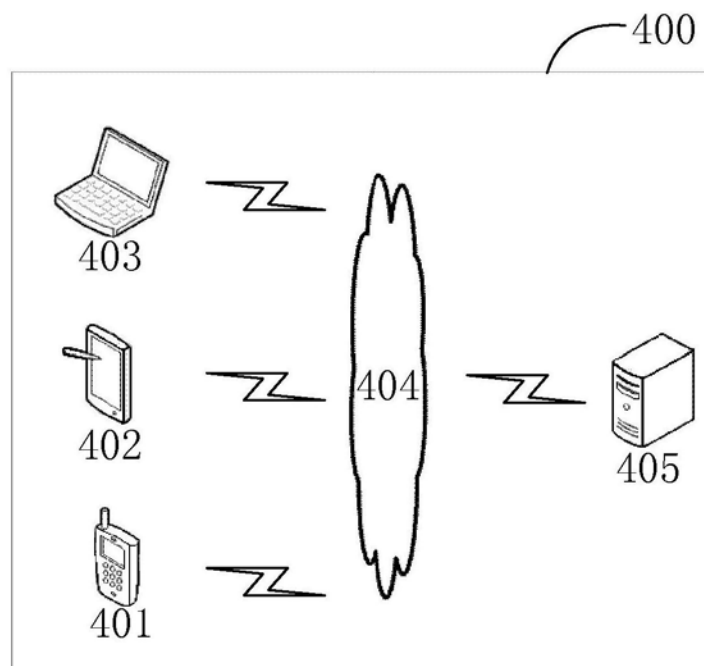


图4

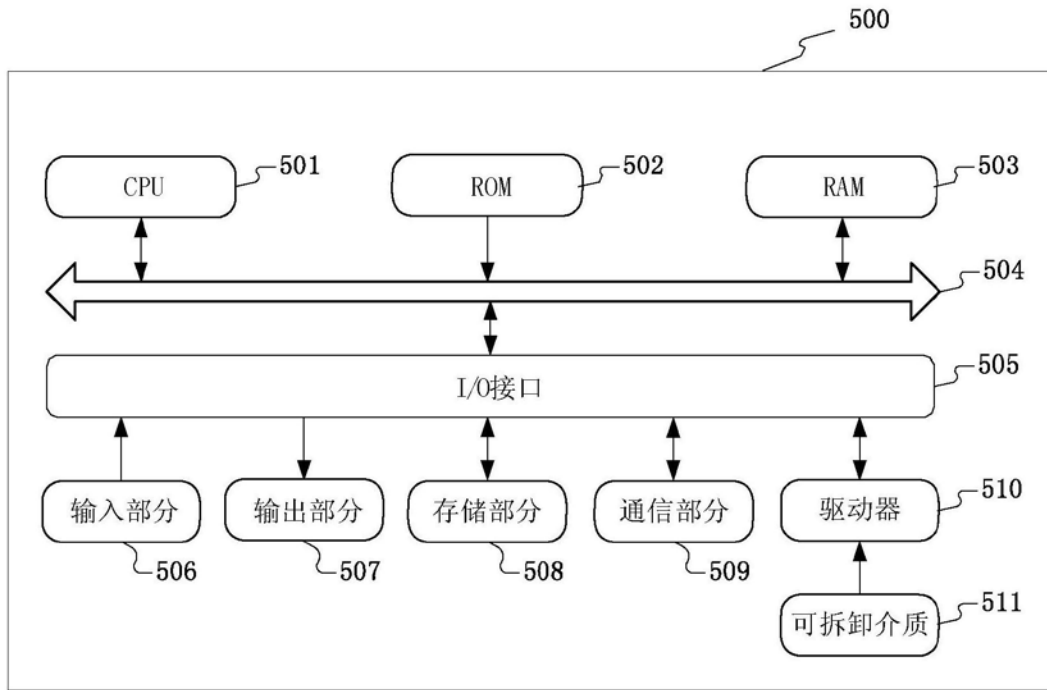


图5