(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 9 月 8 日 (08.09.2017)

WIPO | PCT

(10) 国际公布号
WO 2017/148268 A1

(51) 国际专利分类号:
G06F 9/48 (2006.01)

(21) 国际申请号: PCT/CN2017/073721

(22) 国际申请日: 2017 年 2 月 16 日 (16.02.2017)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201610112497.5 2016 年 2 月 29 日 (29.02.2016) CN

(71) 申请人: **阿里巴巴集团控股有限公司 (ALIBABA GROUP HOLDING LIMITED)** [—/CN]; 开曼群岛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: **方正 (FANG, Zheng)**; 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)。 **刘少伟 (LIU, Shaowei)**; 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, 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, 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,

[见续页]

(54) **Title**: TASK PROCESSING METHOD, APPARATUS AND SYSTEM BASED ON DISTRIBUTED SYSTEM

(54) **发明名称**: 基于分布式系统的任务处理方法、装置和系统

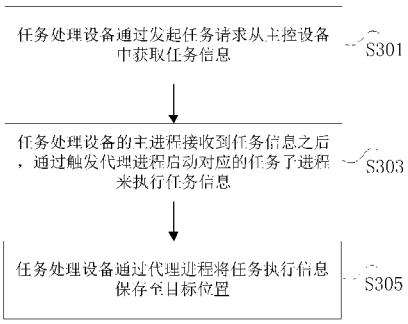


图 3

S301 A TASK PROCESSING DEVICE OBTAINS TASK INFORMATION FROM A MASTER CONTROL DEVICE BY INITIATING A TASK REQUEST
S303 AFTER A MAIN PROCESS OF THE TASK PROCESSING DEVICE RECEIVES THE TASK INFORMATION, THE MAIN PROCESS STARTS, BY TRIGGERING AN AGENT PROCESS, A CORRESPONDING TASK SUBPROCESS TO EXECUTE THE TASK INFORMATION
S305 THE TASK PROCESSING DEVICE STORES TASK EXECUTION INFORMATION TO A TARGET LOCATION BY MEANS OF THE AGENT PROCESS

(57) **Abstract**: A task processing method, apparatus and system based on a distributed system. The method comprises: a task processing device obtains task information from a master control device by initiating a task request (S301), task information corresponding to the task request being stored in the master control device; after a main process of the task processing device receives the task information, the main process starts, by triggering an agent process, a corresponding task subprocess to execute the task information (S303); and the task processing device stores task execution information to a target location by means of the agent process (S305), the task execution information comprising any one or more of the following information: a task identifier of the task information, process information of the task subprocess, log information generated when the task subprocess executes the task information, and an exit code. The method, the apparatus and the system resolve the technical problem in the prior art of task loss caused when a distributed system is established in a specific application environment and a task processing device cannot successfully execute task information.

(57) **摘要**:

[见续页]

WO 2017/148268 A1

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))。

一种基于分布式系统的任务处理方法、装置和系统。其中，该方法包括：任务处理设备通过发起任务请求从主控设备中获取任务信息（S301），其中，在主控设备中保存任务请求对应的任务信息；任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息（S303）；任务处理设备通过代理进程将任务执行信息保存至目标位置（S305），其中，任务执行信息包括如下任意一个或多个信息：任务信息的任务标识、任务子进程的进程信息、任务子进程执行任务信息所生成的日志信息和退出码。所述方法、装置和系统解决了现有技术中分布式执行系统建立在特定的应用环境中，任务处理设备无法成功执行任务信息，导致任务丢失的技术问题。

基于分布式系统的任务处理方法、装置和系统

本申请要求 2016 年 02 月 29 日递交的申请号为 201610112497.5、发明名称为“基于分布式系统的任务处理方法、装置和系统”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。

技术领域

本发明涉及分布式系统领域，具体而言，涉及一种基于分布式系统的任务处理方法、装置和系统。

背景技术

随着微处理机技术的发展，分布式系统可以潜在地得到比单个大型集中式系统更好的性价比，可以通过较低廉的价格实现相似的功能，因此，分布式系统成为越来越普遍使用的系统架构。

但是，现有技术中的分布式执行系统是在特定应用环境中设计的分布式执行系统，其架构如图 1 所示，系统架构分为层：资源管理层和任务执行层，其中资源管理层作为整个系统的中控设备，可以称之为 driver，负责对任务的分配、资源的分配、客户端的 api 接口等功能；任务执行层的任务处理设备可以实现具体的任务执行逻辑，称之为 node，每个 node 部署在一台机器上（gateway），node 是可水平扩展。任务的执行流程图如图 1 所示：

客户端通过 driver 提供的 HTTP 接口提交执行任务；Driver 基于内部的资源管理模型，将任务分发到指定的 node 上；Node 读取到分配到 node 上的任务；Node 获取到任务后，将任务信息注册到内存中；启动任务执行进程，提交到计算引擎执行；保存任务执行日志以及进程 pid 到本地；任务执行的状态（执行中、执行完毕），将任务信息反馈到 driver；Driver 在将任务状态回调给客户端。

通过上述流程可以发现，上述系统中至少存在如下三个问题：

1、如果 driver 与其中某些 node 网络不通，则 driver 发起的任务信息将无法成功发给 node，也无法获知哪些任务信息没有发送成功；

2、如果任务处理设备处于特殊的应用环境，例如在任务处理设备 node 进行升级的过程中，或者进入中断状态下，由于需要重启设备，因此重启过程中会导致任务信息无

法执行等问题；

3、若存在 driver 重复发了多次任务到一台 node 上，node 存在重复执行相同的任务信息的问题，导致资源浪费。

针对现有技术中分布式执行系统建立在特定的应用环境中，任务处理设备无法成功执行任务信息，导致任务丢失的技术问题，目前尚未提出有效的解决方案。

发明内容

本发明实施例提供了一种基于分布式系统的任务处理方法、装置和系统，以至少解决现有技术中分布式执行系统建立在特定的应用环境中，任务处理设备无法成功执行任务信息，导致任务丢失的技术问题。

根据本发明实施例的一个方面，提供了一种基于分布式系统的任务处理方法，包括：任务处理设备通过发起任务请求从主控设备中获取任务信息，其中，在主控设备中保存任务请求对应的任务信息；任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息；任务处理设备通过代理进程将任务执行信息保存至目标位置，其中，任务执行信息包括如下任意一个或多个信息：任务信息的任务标识、任务子进程的进程信息、任务子进程执行任务信息所生成的日志信息和退出码。

根据本发明实施例的另一方面，还提供了一种基于分布式系统的任务处理装置，包括：第一获取模块，用于任务处理设备通过发起任务请求从主控设备中获取任务信息，其中，在主控设备中保存任务请求对应的任务信息；第一执行模块，用于任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息；第一保存模块，用于任务处理设备通过代理进程将任务执行信息保存至目标位置，其中，任务执行信息包括如下任意一个或多个信息：任务信息的任务标识、任务子进程的进程信息、任务子进程执行任务信息所生成的日志信息和退出码。

根据本发明实施例的另一方面，还提供了一种基于分布式系统的任务处理系统，包括：主控设备，用于存储任务信息；任务处理设备，与主控设备进行通讯，用于通过发起任务请求从主控设备中获取任务信息，其中，在主控设备中保存任务请求对应的任务信息；任务处理设备还用于任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息；任务处理设备还用于通过代理进程将任务执行信息保存至目标位置，其中，任务执行信息包括如下任意一个或多个信息：任务信息的任务标识、任务子进程的进程信息、任务子进程执行任务信息所生成的日志信息和

退出码。

在本发明实施例中，任务处理设备通过发起任务请求从主控设备中获取主控设备中保存的任务信息，任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息，任务处理设备通过代理进程将任务执行信息保存至目标位置。

容易注意到，由于任务处理设备中主进程和任务子进程通过代理进程进行隔离，使得主进程停止或者被重启，代理进程及任务子进程都不感知，可以正常执行任务信息，从而实现正在执行的进程在主进程启动过程中不受影响。并且由于任务处理设备主动向主控设备获取任务信息，即任务处理设备单向主控设备访问，因此通过改变 http 方式，利用数据库同步存储数据，从而实现任务处理设备、主控设备不在同一个网段环境下分布式系统的部署问题。能够支持百万级的任务执行，并且通过 HTTP 的方式，可以随时部署 node，大大提升了系统的环境适应能力和可扩展能力。

由此，本发明提供的实施例解决了现有技术中分布式执行系统建立在特定的应用环境中，任务处理设备无法成功执行任务信息，导致任务丢失的技术问题。

附图说明

此处所说明的附图用来提供对本申请的进一步理解，构成本申请的一部分，本申请的示意性实施例及其说明用于解释本申请，并不构成对本申请的不当限定。在附图中：

图 1 是根据现有技术的一种分布式执行系统的任务处理的流程图；

图 2 是根据本申请实施例的一种基于分布式系统的任务处理方法的计算机终端的硬件结构框图；

图 3 是根据本申请实施例的一种基于分布式系统的任务处理方法的流程图；

图 4 是根据本申请实施例的一种分布式执行系统的结构示意图；

图 5 是根据本申请实施例的一种可选的任务处理设备的示意图；

图 6 是根据本申请实施例的一种可选的任务处理设备执行任务的流程图；

图 7 是根据本申请实施例的一种可选的任务处理设备重启后恢复任务信息的流程图；

图 8 是根据本申请实施例的基于分布式系统的任务处理方法的交互图；

图 9 是根据本申请实施例的基于分布式系统的任务处理装置的示意图；

图 10 是根据本申请实施例的可选的基于分布式系统的任务处理装置的示意图；

图 11 是根据本申请实施例的可选的基于分布式系统的任务处理装置的示意图；

图 12 是根据本申请实施例的可选的基于分布式系统的任务处理装置的示意图；

图 13 是根据本申请实施例的可选的基于分布式系统的任务处理装置的示意图；

图 14 是根据本申请实施例的可选的基于分布式系统的任务处理装置的示意图；

图 15 是根据本申请实施例的可选的基于分布式系统的任务处理装置的示意图；

图 16 是根据本申请实施例的基于分布式系统的任务处理系统的示意图；以及

图 17 是根据本申请实施例的一种计算机终端的结构框图。

具体实施方式

为了使本技术领域的人员更好地理解本申请方案，下面将结合本申请实施例中的附图，对本申请实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本申请一部分的实施例，而不是全部的实施例。基于本申请中的实施例，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，都应当属于本申请保护的范围。

需要说明的是，本申请的说明书和权利要求书及上述附图中的术语“第一”、“第二”等是用于区别类似的对象，而不必用于描述特定的顺序或先后次序。应该理解这样使用的数据在适当情况下可以互换，以便这里描述的本申请的实施例能够以除了在这里图示或描述的那些以外的顺序实施。此外，术语“包括”和“具有”以及他们的任何变形，意图在于覆盖不排他的包含，例如，包含了一系列步骤或单元的过程、方法、系统、产品或设备不必限于清楚地列出的那些步骤或单元，而是可包括没有清楚地列出的或对于这些过程、方法、产品或设备固有的其它步骤或单元。

首先，在对本申请实施例进行描述的过程中出现的部分名词或术语适用于如下解释：

分布式系统：distributed system，是建立在网络上的软件系统，具有高度的内聚性和透明性。内聚性是指每一个数据库分布节点高度自治，有本地的数据库管理系统。透明性是指每一个数据库分布节点对用户的应用来说都是透明的，看不出是本地还是远程。

driver：主控设备，分布式执行系统中的服务层（中控），负责资源管理。

node：任务处理设备，分布式执行系统中的集群的机器节点，也称之为 gateway，具体负责任务的执行。

pid：任务进程编号，启动代理进程时进行保存。

实施例 1

根据本申请实施例，还提供了一种基于分布式系统的任务处理方法实施例，需要说明的是，在附图的流程图示出的步骤可以在诸如一组计算机可执行指令的计算机系统中执行，并且，虽然在流程图中示出了逻辑顺序，但是在某些情况下，可以以不同于此处的顺序执行所示出或描述的步骤。

本申请实施例一所提供的方法实施例可以在移动终端、计算机终端或者类似的运算装置中执行。以运行在计算机终端上为例，图 2 是根据本申请实施例的一种基于分布式系统的任务处理方法的计算机终端的硬件结构框图。如图 2 所示，计算机终端 20 可以包括一个或多个（图中仅示出一个）处理器 202（处理器 202 可以包括但不限于微处理器 MCU 或可编程逻辑器件 FPGA 等的处理装置）、用于存储数据的存储器 204、以及用于通信功能的传输模块 206。本领域普通技术人员可以理解，图 2 所示的结构仅为示意，其并不对上述电子装置的结构造成限定。例如，计算机终端 20 还可包括比图 2 中所示更多或者更少的组件，或者具有与图 2 所示不同的配置。

存储器 204 可用于存储应用软件的软件程序以及模块，如本申请实施例中的基于分布式系统的任务处理方法对应的程序指令/模块，处理器 202 通过运行存储在存储器 204 内的软件程序以及模块，从而执行各种功能应用以及数据处理，即实现上述的基于分布式系统的任务处理方法。存储器 204 可包括高速随机存储器，还可包括非易失性存储器，如一个或者多个磁性存储装置、闪存、或者其他非易失性固态存储器。在一些实例中，存储器 204 可进一步包括相对于处理器 202 远程设置的存储器，这些远程存储器可以通过网络连接至计算机终端 20。上述网络的实例包括但不限于互联网、企业内部网、局域网、移动通信网及其组合。

传输装置 206 用于经由一个网络接收或者发送数据。上述的网络具体实例可包括计算机终端 20 的通信供应商提供的无线网络。在一个实例中，传输装置 206 包括一个网络适配器（Network Interface Controller，NIC），其可通过基站与其他网络设备相连从而可与互联网进行通讯。在一个实例中，传输装置 206 可以为射频（Radio Frequency，RF）模块，其用于通过无线方式与互联网进行通讯。

在上述运行环境下，本申请提供了如图 3 所示的基于分布式系统的任务处理方法。图 3 是根据本申请实施例的一种基于分布式系统的任务处理方法的流程图，如图 3 所示的方法可以包括如下步骤 S301 至步骤 S305：

步骤 S301，任务处理设备通过发起任务请求从主控设备中获取任务信息，其中，在主控设备中保存任务请求对应的任务信息。

可选地，上述任务处理设备可以是部署在分布式系统中的集群的机器节点 node，上述任务请求可以是 http 请求，上述主控设备可以是 driver，主控设备的数据库中保存有任务处理设备需要的任务信息。

需要说明的是，分布式系统中任务都在资源管理层 Driver 层做分发，driver 管理 node、任务。每台 node 上正在执行的任务列表都存储在 driver 的 DB 中，存储周期为任务生命周期。

图 4 是根据本申请实施例的一种分布式执行系统的结构示意图，如图 4 所示，在一种可选的方案中，任务处理设备主动发送 http 请求给主控设备，主控设备接收到 http 请求之后，提取 http 请求对应的任务信息，通过接口将该任务信息发送给任务处理设备。

步骤 S303，任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息。

可选地，上述代理进程可以是代理脚本，代理脚本可以作为主进程与任务子进程的通道。

图 5 是根据本申请实施例的一种可选的任务处理设备的示意图，如图 5 所示，任务处理设备 node 中的主进程（AlisaNode）与任务子进程隔离，代理进程（Python 进程）关联任务子进程，并且读写日志。图 6 是根据本申请实施例的一种可选的任务处理设备执行任务的流程图，结合图 5 和图 6 可知，在一种可选的方案中，node 从 driver 中获取到任务之后，启动代理进程，传入任务执行需要的参数，代理进程启动具体的任务子进程执行任务信息。

步骤 S305，任务处理设备通过代理进程将任务执行信息保存至目标位置，其中，任务执行信息包括如下任意一个或多个信息：任务信息的任务标识、任务子进程的进程信息、任务子进程执行任务信息所生成的日志信息和退出码。

可选地，上述目标位置可以是代理进程的本地文件，路径可以是任务执行路径，上述任务信息的任务标识可以是任务 id，上述任务子进程的进程信息可以是任务子进程的进程号 pid。

需要说明的是，上述任务子进程执行任务信息所生成的日志信息为完整的日志信息，是指日志动作与任务处理设备重启不相关，并且只要代理进程存在，日志就可以正常读写。

如图 6 所示，在一种可选的方案中，任务子进程将所有的日志信息通过流方式返回给代理服务器，代理进程将接收到的日志信息、子进程的进程号、任务信息标号和退出

码保存到本地文件中。

本申请上述实施例一公开的方案中，任务处理设备通过发起任务请求从主控设备中获取主控设备中保存的任务信息，任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息，任务处理设备通过代理进程将任务执行信息保存至目标位置。

容易注意到，由于任务处理设备中主进程和任务子进程通过代理进程进行隔离，使得主进程停止或者被重启，代理进程及任务子进程都不感知，可以正常执行任务信息，从而实现正在执行的进程在主进程启动过程中不受影响。并且由于任务处理设备主动向主控设备获取任务信息，即任务处理设备单向主控设备访问，因此通过改变 http 方式，利用数据库同步存储数据，从而实现任务处理设备、主控设备不在同一个网段环境下分布式系统的部署问题。能够支持百万级的任务执行，并且通过 HTTP 的方式，可以随时部署 node，大大提升了系统的环境适应能力和可扩展能力。

由此，本申请提供的上述实施例一的方案解决了现有技术中分布式执行系统建立在特定的应用环境中，任务处理设备无法成功执行任务信息，导致任务丢失的技术问题。

在本申请上述实施例中，在步骤 S303 主进程接收到任务信息之后，上述方法还包括如下步骤 S307：

步骤 S307，如果主进程被重启，则通过代理进程离线启动任务子进程来执行任务信息，并离线保存任务执行信息。

在一种可选的方案中，如果 node 主进程被重启，代理进程离线启动任务子进程执行任务信息，并将任务子进程发送的任务执行信息通过离线存储方式储存在本地文件中。

通过上述步骤 S307，主进程被重启，代理进程通过离线存储任务子进程的任务执行信息的方式保护任务状态，从而实现任务处理设备热重启不影响任务执行、不丢失日志信息的目的。

在本申请上述实施例中，在步骤 S307 主进程被重启之后，上述方法还包括如下步骤 S311 至步骤 S313：

步骤 S311，任务处理设备通过访问主控设备，来获取当前正在执行的至少一个任务信息，任务信息包括：任务标识和任务执行路径。

可选地，上述任务执行路径可以是全局唯一的关键路径，这个路径可以维持整个任务的生命周期，任务执行路径的生成方式为：客户端名称+日期+随机数+任务编号。

图 7 是根据本申请实施例的一种可选的任务处理设备重启后恢复任务信息的流程

图，如图 7 所示，在一种可选的方案中，node 上保留任务信息都在内存中，node 重启后任务信息丢失。但是任务子进程与 node 主进程分离，node 重启，任务正常执行。待 node 恢复后，调用恢复（recover）接口，获取 driver 中找到当前正在执行的任务列表（即上述的当前正在执行的至少一个任务信息），每个任务包含有任务编号和任务执行路径。

步骤 S313，任务处理设备通过检测在目标位置中记录的任务执行信息，来确定在主进程被重启时至少一个任务信息中已经被执行或未被执行的任务。

可选地，通过查询该任务执行路径下的日志文件，可以得到日志信息接口；通过查询该任务执行路径下的任务子进程文件，可以得到该任务信息是否已经被任务子进程执行；通过查询该任务执行路径下的退出码文件，可以得到该任务信息是否已经被任务子进程执行结束。

如图 7 所示，在一种可选的方案中，任务处理设备对每个任务基于任务执行路径长训是否存在对应的子进程的进程号，如果存在对应的子进程的进程号，则说明该任务已经被执行；如果不存在对应的子进程的进程号，则说明该任务为被执行。

通过上述步骤 S311 至步骤 S313，任务处理设备主进程被重启后，从主控设备中获取当前正在执行的至少一个任务信息，并检测在目标位置中记录的任务执行信息，确定至少一个任务信息是否已经被执行或未被执行，从而实现主进程重启后，任务子进程执行任务信息的一致性。

在本申请上述实施例中，步骤 S313 任务处理设备通过检测在目标位置中记录的任务执行信息，来确定至少一个任务信息中已经被执行或未被执行的任务，包括如下步骤 S3133 至步骤 S3135：

步骤 S3131，任务处理设备在目标位置进行本地查询，确定是否已经存在与至少一个任务信息中任意一个任务标识相匹配的进程信息。

可选地，上述进程信息可以是子进程的进程号，每一个任务子进程在启动执行任务信息时，都会将自己的进程号对应的存放在该任务信息中。

在一种可选的方案中，任务处理设基于任务执行路径进行查询，根据任务标识确定至少一个任务信息中任意一个任务标识是否存在对应的子进程的进程号。

步骤 S3133，如果存在，则确定在主进程被重启时，任务信息已经被对应的任务子进程启动执行，任务处理设备仅在内存中注册该任务信息。

如图 7 所示，在一种可选的方案中，如果存在子进程的进程号，说明该任务信息已经被相应的任务子进程启动执行，则任务处理设备重新恢复任务信息（即上述的在内存

中注册该任务信息），并读取该任务的任务进程信息（即上述的任务执行信息）。

步骤 S3135，如果不存在，则确定在主进程被重启时，任务信息未被对应的任务子进程所执行，任务处理设备在内存中注册该任务信息，并启动对应的任务子进程执行任务信息。

如图 7 所示，在一种可选的方案中，如果不存在子进程的进程号，说明该任务信息未被相应的任务子进程执行，则任务处理设备在内存中保存该任务信息，通过启动代理进程来启动相应的任务子进程执行任务信息。

通过上述步骤 S3133 至步骤 S3135，任务处理设备从在目标位置进行本地查询，确定是否存在相应的子进程的进程号，如果存在则重新记录任务信息；如果不存在，则通过代理进程启动任务子进程执行任务信息，从而实现主进程重启后，任务子进程执行任务信息的一致性。

在本申请上述实施例中，在步骤 S3133 任务处理设备仅在内存中注册该任务信息之前，上述方法还包括如下步骤 S317 至步骤 S319：

步骤 S315，任务处理设备在目标位置进行本地查询，确定是否已经存在与已经被对应的任务子进程启动执行的任务信息相匹配的退出码。

可选地，上述退出码（returncode）用于表征该任务信息已经执行结束。

步骤 S317，如果存在，则确定在主进程被重启时，任务信息已经被对应的任务子进程执行结束，任务处理设备终止在内存中注册该任务信息，直接将对应的任务日志反馈给主控设备。

步骤 S319，如果不存在，则确定在主进程被重启时，任务信息未被对应的任务子进程执行完毕，任务处理设备在内存中注册该任务信息，并启动对应的任务子进程重新执行任务信息。

在一种可选的方案中，如果任务处理设备确定存在对应的子进程的进程号，则判断是否存在任务信息对应的退出码，如果存在，则说明该任务信息已经被执行完毕，那么就不需要在内存中注册该任务信息，直接将相应的任务日志反馈给主控设备；如果不存在，则说明该任务未被执行完毕，需要通过代理进程启动相应的任务子进程重新执行该任务信息。

通过上述步骤 S317 至步骤 S319，任务处理设备在目标位置进行本地查询，确定是否已经存在对应的退出码，如果存在则反馈任务日志，如果不存在则启动相应的任务子进程重新执行该任务信息，从而实现主进程重启后，恢复正在执行的任务信息，保证了

任务信息的状态。

在本申请上述实施例中，在步骤 S301 任务处理设备通过发起任务请求从主控设备中获取任务信息之后，上述方法还包括如下步骤 S321 至步骤 S323：

步骤 S321，从任务信息中提取路径参数，并根据路径参数生成任务执行路径。

在一种可选的方案中，任务处理设备在获取到任务信息时，会从任务信息中获取到一条路径参数，任务处理设备将该参数作为当前的任务执行路径，任务的执行日志、进程号、退出码都保存在该路径下。

步骤 S323，代理进程根据任务执行路径将任务标识保存至目标位置。

在一种可选的方案中，任务子进程在执行启动时会将当前子进程的进程号保存到本地文件中，保存的路径为任务执行路径。

通过上述步骤 S321 至步骤 S323，任务处理设备从任务信息中提取路径参数，生成任务执行路径，代理进程根据任务执行路径将任务标识保存在目标位置，从而保证主进程重启前后任务日志的完整性和可查性。

在本申请上述实施例中，上述方法还包括如下步骤 S325：

步骤 S325，任务处理设备的主进程根据任务标识监听任务信息是否被启动执行，并根据退出码监听任务信息是否执行结束。

通过上述步骤 S325，任务处理设备可以通过任务标识和退出码监听任务信息是否被执行，以及是否执行结束，从而确保任务信息全部被执行且全部执行完毕。

下面结合图 5、图 7 和图 8 详细介绍本申请的一种优选实施例。

如图 8 所示，以任务处理设备包括主进程、代理进程和任务子进程为应用场景，提供了一种可选的基于分布式系统的任务处理方法，该方法可以包括如下步骤 S81 至步骤 S87：

S801，客户终端 81 提交任务。

可选地，客户可以通过客户终端 81 发布任务信息。

S802，任务处理设备 163 的主进程获取主控设备 161 中保存的任务信息。

可选地，任务处理设备 163 的主进程通过 http 请求从主控设备 161 中获取任务信息，任务信息保存在主控设备中。

在该实施例中，上述步骤 S802 的实现方式与本申请上述实施例中的步骤 S301 的实现方式一致，在此不再赘述。

S803，任务处理设备 163 通过代理进程启动任务子进程执行任务信息。

可选地，任务处理设备 163 的主进程收到任务信息之后，通过触发代理进程启动任务子进程执行任务信息，如图 5 所示，任务处理设备 node 中的主进程（AlisaNode）与任务子进程隔离，代理进程（Python 进程）关联任务子进程，并且读写日志。

在该实施例中，上述步骤 S803 的实现方式与本申请上述实施例中的步骤 S303 的实现方式一致，在此不再赘述。

S804，任务处理设备 163 通过代理进程保存任务执行信息。

可选地，任务处理设备 163 在获取到任务信息时，从任务信息中获取路径参数，生成任务执行路径，代理进程将任务信息的任务标识、任务子进程的进程信息、日志信息和退出码保存在该任务执行路径对应的目标位置。

在该实施例中，上述步骤 S804 的实现方式与本申请上述实施例中的步骤 S305 的实现方式一致，在此不再赘述。

S805，主进程重启，任务处理设备 163 通过理进程离线启动任务子进程执行任务信息，并离线保存任务执行信息。

可选地，如果主进程被重启，代理进程离线启动任务子进程来执行任务信息，并将任务子进程执行任务信息生成的任务执行信息通过离线存储方式保存在目标位置。主进程被重启后，任务处理设备 163 从主控设备 161 中获取正在执行的任务信息，并从目标位置中查询任务执行信息，确定主进程被重启时该任务信息是否已经被执行或未被执行。

如图 7 所示，在一种可选的方案中，主进程被重启后，任务处理设备 163 调用恢复接口获取主控设备 161 中保存的任务列表，检测目标位置中是否存在对应的任务子进程的进程号，如果不存在，说明该任务信息未被相应的任务子进程执行，则任务处理设备 163 在内存中保存该任务信息，通过启动代理进程来启动相应的任务子进程执行任务信息；如果存在，说明该任务信息已经被相应的任务子进程启动执行，则任务处理设备 163 检测目标位置中是否存在对应的退出码，如果不存在，则通过代理进程启动对应的任务子进程重新执行该任务信息，如果存在，则返回任务日志给主控设备 161。

S806，任务处理设备 163 返回任务状态给主控设备 161。

可选地，主进程根据任务标识监听任务信息是否被启动执行，根据退出码监听任务信息是否执行结束，如果主进程监听该任务已经被执行，则反馈任务执行信息给主控设备 161，如果监听该任务信息已经执行结束，则反馈任务日志给主控设备 161。

在该实施例中，上述步骤 S806 的实现方式与本申请上述实施例中的步骤 S325 的实现方式一致，在此不再赘述。

S807，主控设备 161 将任务日志反馈给客户终端 81。

可选地，主控设备 161 将执行结束的任务信息的任务日志反馈给客户端，经正在执行的任务信息的任务执行信息给客户终端 81。

需要说明的是，对于前述的各方法实施例，为了简单描述，故将其都表述为一系列的动作组合，但是本领域技术人员应该知悉，本申请并不受所描述的动作顺序的限制，因为依据本申请，某些步骤可以采用其他顺序或者同时进行。其次，本领域技术人员也应该知悉，说明书中所描述的实施例均属于优选实施例，所涉及的动作和模块并不一定是本申请所必须的。

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根据上述实施例的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，或者网络设备等）执行本申请各个实施例所述的方法。

实施例 2

根据本申请实施例，还提供了一种用于实施上述基于分布式系统的任务处理方法的基于分布式系统的任务处理装置，如图 9 所示，该装置包括：第一获取模块 91，第一执行模块 93 和第一保存模块 95。

其中第一获取模块 91 用于任务处理设备通过发起任务请求从主控设备中获取任务信息，其中，在主控设备中保存任务请求对应的任务信息。第一执行模块 93 用于任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息。第一保存模块 95 用于任务处理设备通过代理进程将任务执行信息保存至目标位置，其中，任务执行信息包括如下任意一个或多个信息：任务信息的任务标识、任务子进程的进程信息、任务子进程执行任务信息所生成的日志信息和退出码。

可选地，上述任务处理设备可以是部署在分布式系统中的集群的机器节点 node，上述任务请求可以是 http 请求，上述主控设备可以是 driver，主控设备的数据库中保存有任务处理设备需要的任务信息。上述代理进程可以是代理脚本，代理脚本可以作为主进程与任务子进程的通道。上述目标位置可以是代理进程的本地文件，路径可以是任务执行路径，上述任务信息的任务标识可以是任务 id，上述任务子进程的进程信息可以是任务子进程的进程号 pid。

需要说明的是，分布式系统中任务都在资源管理层 Driver 层做分发，driver 管理 node、任务。每台 node 上正在执行的任务列表都存储在 driver 的 DB 中，存储周期为任务生命周期。上述任务子进程执行任务信息所生成的日志信息为完整的日志信息，是指日志动作与任务处理设备重启不相关，并且只要代理进程存在，日志就可以正常读写。

此处需要说明的是，上述第一获取模块 91，第一执行模块 93 和第一保存模块 95 对应于实施例一中的步骤 S301 至步骤 S305，三个模块与对应的步骤所实现的实例和应用场景相同，但不限于上述实施例一所公开的内容。需要说明的是，上述模块作为装置的一部分可以运行在实施例一提供的计算机终端 20 中。

本申请上述实施例二公开的方案中，任务处理设备通过发起任务请求从主控设备中获取主控设备中保存的任务信息，任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息，任务处理设备通过代理进程将任务执行信息保存至目标位置。

容易注意到，由于任务处理设备中主进程和任务子进程通过代理进程进行隔离，使得主进程停止或者被重启，代理进程及任务子进程都不感知，可以正常执行任务信息，从而实现正在执行的进程在主进程启动过程中不受影响。并且由于任务处理设备主动向主控设备获取任务信息，即任务处理设备单向主控设备访问，因此通过改变 http 方式，利用数据库同步存储数据，从而实现任务处理设备、主控设备不在同一个网段环境下分布式系统的部署问题。能够支持百万级的任务执行，并且通过 HTTP 的方式，可以随时部署 node，大大提升了系统的环境适应能力和可扩展能力。

由此，本申请提供的上述实施例二的方案解决了现有技术中分布式执行系统建立在特定的应用环境中，任务处理设备无法成功执行任务信息，导致任务丢失的技术问题。

在本申请上述实施例中，如图 10 所示，上述装置还包括：第二执行模块 101。

其中，第二执行模块 101 用于在主进程接收到任务信息之后，如果主进程被重启，则通过代理进程离线启动任务子进程来执行任务信息，并离线保存任务执行信息。

此处需要说明的是，上述第二执行模块 101 对应于实施例一中的步骤 S307，上述模块与对应的步骤所实现的实例和应用场景相同，但不限于上述实施例一所公开的内容。需要说明的是，上述模块作为装置的一部分可以运行在实施例一提供的计算机终端 20 中。

在本申请上述实施例中，如图 11 所示，上述装置还包括：第二获取模块 111 和第一确定模块 113。

其中。第二获取模块 111 用于任务处理设备通过访问主控设备，来获取当前正在执行的至少一个任务信息，任务信息包括：任务标识和任务执行路径。第一确定模块 113 用于任务处理设备通过检测在目标位置中记录的任务执行信息，来确定在主进程被重启时至少一个任务信息中已经被执行或未被执行的任务。

可选地，上述任务执行路径可以是全局唯一的关键路径，这个路径可以维持整个任务的生命周期，任务执行路径的生成方式为：客户端名称+日期+随机数+任务编号。通过查询该任务执行路径下的日志文件，可以得到日志信息接口；通过查询该任务执行路径下的任务子进程文件，可以得到该任务信息是否已经被任务子进程执行；通过查询该任务执行路径下的退出码文件，可以得到该任务信息是否已经被任务子进程执行结束。

此处需要说明的是，上述第二获取模块 111 和第一确定模块 113 对应于实施例一中的步骤 S311 至步骤 S313，两个模块与对应的步骤所实现的实例和应用场景相同，但不限于上述实施例一所公开的内容。需要说明的是，上述模块作为装置的一部分可以运行在实施例一提供的计算机终端 20 中。

在本申请上述实施例中，如图 12 所示，第一确定模块 113 包括：第二确定模块 121，第一注册模块 123 和第二注册模块 125。

其中，第二确定模块 121 用于任务处理设备在目标位置进行本地查询，确定是否已经存在与至少一个任务信息中任意一个任务标识相匹配的进程信息。第一注册模块 123 用于如果存在，则确定在主进程被重启时，任务信息已经被对应的任务子进程启动执行，任务处理设备仅在内存中注册该任务信息。第二注册模块 125 用于如果不存在，则确定在主进程被重启时，任务信息未被对应的任务子进程所执行，任务处理设备在内存中注册该任务信息，并启动对应的任务子进程执行任务信息。

可选地，上述进程信息可以是子进程的进程号，每一个任务子进程在启动执行任务信息时，都会将自己的进程号对应的存放在该任务信息中。

此处需要说明的是，上述第二确定模块 121，第一注册模块 123 和第二注册模块 125 对应于实施例一中的步骤 S3133 至步骤 S3135，三个模块与对应的步骤所实现的实例和应用场景相同，但不限于上述实施例一所公开的内容。需要说明的是，上述模块作为装置的一部分可以运行在实施例一提供的计算机终端 20 中。

在本申请上述实施例中，如图 13 所示，上述装置还包括：第三确定模块 131，反馈模块 133 和第三注册模块 135。

其中，第三确定模块 131 用于任务处理设备在目标位置进行本地查询，确定是否已

经存在与已经被对应的任务子进程启动执行的任务信息相匹配的退出码。反馈模块 133 用于如果存在，则确定在主进程被重启时，任务信息已经被对应的任务子进程执行结束，任务处理设备终止在内存中注册该任务信息，直接将对应的任务日志反馈给主控设备。第三注册模块 135 用于如果不存在，则确定在主进程被重启时，任务信息未被对应的任务子进程执行完毕，任务处理设备在内存中注册该任务信息，并启动对应的任务子进程重新执行任务信息。

可选地，上述退出码（returncode）用于表征该任务信息已经执行结束。

此处需要说明的是，上述第三确定模块 131，反馈模块 133 和第三注册模块 135 对应于实施例一中的步骤 S317 至步骤 S319，三个模块与对应的步骤所实现的实例和应用场景相同，但不限于上述实施例一所公开的内容。需要说明的是，上述模块作为装置的一部分可以运行在实施例一提供的计算机终端 20 中。

在本申请上述实施例中，如图 14 所示，上述装置还包括：生成模块 141 和第二保存模块 143。

其中，生成模块 141 用于从任务信息中提取路径参数，并根据路径参数生成任务执行路径。第二保存模块 143 用于代理进程根据任务执行路径将任务标识保存至目标位置。

此处需要说明的是，上述生成模块 141 和第二保存模块 143 对应于实施例一中的步骤 S321 至步骤 S323，两个模块与对应的步骤所实现的实例和应用场景相同，但不限于上述实施例一所公开的内容。需要说明的是，上述模块作为装置的一部分可以运行在实施例一提供的计算机终端 20 中。

在本申请上述实施例中，如图 15 所示，上述装置还包括：监听模块 151。

其中，监听模块 151 用于任务处理设备的主进程根据任务标识监听任务信息是否被启动执行，并根据退出码监听任务信息是否执行结束。

此处需要说明的是，上述监听模块 151 对应于实施例一中的步骤 S325，上述模块与对应的步骤所实现的实例和应用场景相同，但不限于上述实施例一所公开的内容。需要说明的是，上述模块作为装置的一部分可以运行在实施例一提供的计算机终端 20 中。

实施例 3

根据本申请实施例，还提供了一种用于实施上述基于分布式系统的任务处理方法的基于分布式系统的任务处理系统，如图 16 所示，该系统包括：主控设备 161 和任务处理设备 163。

其中，主控设备 161 用于存储任务信息。

任务处理设备 163 与主控设备 161 进行通讯，用于通过发起任务请求从主控设备中获取任务信息，其中，在主控设备中保存任务请求对应的任务信息。

可选地，上述任务处理设备 163 可以是部署在分布式系统中的集群的机器节点 node，上述任务请求可以是 http 请求，上述主控设备可以是 driver，主控设备的数据库中保存有任务处理设备需要的任务信息。

需要说明的是，分布式系统中任务都在资源管理层 Driver 层做分发，driver 管理 node、任务。每台 node 上正在执行的任务列表都存储在 driver 的 DB 中，存储周期为任务生命周期。

如图 3 所示，在一种可选的方案中，任务处理设备主动发送 http 请求给主控设备，主控设备接收到 http 请求之后，提取 http 请求对应的任务信息，通过接口将该任务信息发送给任务处理设备。

任务处理设备 163 还用于任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息。

可选地，上述代理进程可以是代理脚本，代理脚本可以作为主进程与任务子进程的通道。

如图 5 所示，任务处理设备 node 中的主进程（AlisaNode）与任务子进程隔离，代理进程（Python 进程）关联任务子进程，并且读写日志。结合图 5 和图 6 可知，在一种可选的方案中，node 从 driver 中获取到任务之后，启动代理进程，传入任务执行需要的参数，代理进程启动具体的任务子进程执行任务信息。

任务处理设备 163 还用于通过代理进程将任务执行信息保存至目标位置，其中，任务执行信息包括如下任意一个或多个信息：任务信息的任务标识、任务子进程的进程信息、任务子进程执行任务信息所生成的日志信息和退出码。

可选地，上述目标位置可以是代理进程的本地文件，路径可以是任务执行路径，上述任务信息的任务标识可以是任务 id，上述任务子进程的进程信息可以是任务子进程的进程号 pid。

需要说明的是，上述任务子进程执行任务信息所生成的日志信息为完整的日志信息，是指日志动作与任务处理设备重启不相关，并且只要代理进程存在，日志就可以正常读写。

如图 6 所示，在一种可选的方案中，任务子进程将所有的日志信息通过流方式返回给代理服务器，代理进程将接收到的日志信息、子进程的进程号、任务信息标号和退出

码保存到本地文件中。

本申请上述实施例三公开的方案中，任务处理设备通过发起任务请求从主控设备中获取主控设备中保存的任务信息，任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息，任务处理设备通过代理进程将任务执行信息保存至目标位置。

容易注意到，由于任务处理设备中主进程和任务子进程通过代理进程进行隔离，使得主进程停止或者被重启，代理进程及任务子进程都不感知，可以正常执行任务信息，从而实现正在执行的进程在主进程启动过程中不受影响。并且由于任务处理设备主动向主控设备获取任务信息，即任务处理设备单向主控设备访问，因此通过改变 http 方式，利用数据库同步存储数据，从而实现任务处理设备、主控设备不在同一个网段环境下分布式系统的部署问题。能够支持百万级的任务执行，并且通过 HTTP 的方式，可以随时部署 node，大大提升了系统的环境适应能力和可扩展能力。

由此，本申请提供的上述实施例三的方案解决了现有技术中分布式执行系统建立在特定的应用环境中，任务处理设备无法成功执行任务信息，导致任务丢失的技术问题。

在本申请提供的一种可选实施例中，在主进程接收到任务信息之后，任务处理设备 163 还用于如果主进程被重启，则通过代理进程离线启动任务子进程来执行任务信息，并离线保存任务执行信息。

在本申请提供的一种可选实施例中，在主进程被重启之后，任务处理设备 163 还用于通过访问主控设备，来获取当前正在执行的至少一个任务信息，任务信息包括：任务标识和任务执行路径；通过检测在目标位置中记录的任务执行信息，来确定在主进程被重启时至少一个任务信息中已经被执行或未被执行的任务。

在本申请提供的一种可选实施例中，任务处理设备 163 还用于在目标位置进行本地查询，确定是否已经存在与至少一个任务信息中任意一个任务标识相匹配的进程信息；如果存在，则确定在主进程被重启时，任务信息已经被对应的任务子进程启动执行，任务处理设备仅在内存中注册该任务信息；如果不存在，则确定在主进程被重启时，任务信息未被对应的任务子进程所执行，任务处理设备在内存中注册该任务信息，并启动对应的任务子进程执行任务信息。

在本申请提供的一种可选实施例中，任务处理设备 163 还用于在目标位置进行本地查询，确定是否已经存在与已经被对应的任务子进程启动执行的任务信息相匹配的退出码；如果存在，则确定在主进程被重启时，任务信息已经被对应的任务子进程执行结束，

任务处理设备终止在内存中注册该任务信息，直接将对应的任务日志反馈给主控设备；如果不存在，则确定在主进程被重启时，任务信息未被对应的任务子进程执行完毕，任务处理设备在内存中注册该任务信息，并启动对应的任务子进程重新执行任务信息。

在本申请提供的一种可选实施例中，任务处理设备 163 还用于从任务信息中提取路径参数，并根据路径参数生成任务执行路径；其中，代理进程根据任务执行路径将任务标识保存至目标位置。

在本申请提供的一种可选实施例中，任务处理设备 163 还用于主进程根据任务标识监听任务信息是否被启动执行，并根据退出码监听任务信息是否执行结束。

实施例 4

本申请的实施例可以提供一种计算机终端，该计算机终端可以是计算机终端群中的任意一个计算机终端设备。可选地，在本实施例中，上述计算机终端也可以替换为移动终端等终端设备。

可选地，在本实施例中，上述计算机终端可以位于计算机网络的多个网络设备中的至少一个网络设备。

在本实施例中，上述计算机终端可以执行基于分布式系统的任务处理方法中以下步骤的程序代码：任务处理设备通过发起任务请求从主控设备中获取任务信息，其中，在主控设备中保存任务请求对应的任务信息；任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息；任务处理设备通过代理进程将任务执行信息保存至目标位置，其中，任务执行信息包括如下任意一个或多个信息：任务信息的任务标识、任务子进程的进程信息、任务子进程执行任务信息所生成的日志信息和退出码。

可选地，图 17 是根据本申请实施例的一种计算机终端的结构框图。如图 17 所示，该计算机终端 A 可以包括：一个或多个（图中仅示出一个）处理器 171、存储器 173、以及传输装置 175。

其中，存储器 173 可用于存储软件程序以及模块，如本申请实施例中的基于分布式系统的任务处理方法和装置对应的程序指令/模块，处理器 171 通过运行存储在存储器内的软件程序以及模块，从而执行各种功能应用以及数据处理，即实现上述的基于分布式系统的任务处理方法。存储器 173 可包括高速随机存储器，还可以包括非易失性存储器，如一个或者多个磁性存储装置、闪存、或者其他非易失性固态存储器。在一些实例中，存储器 173 可进一步包括相对于处理器远程设置的存储器，这些远程存储器可以通过网

络连接至终端 A。上述网络的实例包括但不限于互联网、企业内部网、局域网、移动通信网及其组合。

处理器 171 可以通过传输装置调用存储器存储的信息及应用程序，以执行下述步骤：任务处理设备通过发起任务请求从主控设备中获取任务信息，其中，在主控设备中保存任务请求对应的任务信息；任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息；任务处理设备通过代理进程将任务执行信息保存至目标位置，其中，任务执行信息包括如下任意一个或多个信息：任务信息的任务标识、任务子进程的进程信息、任务子进程执行任务信息所生成的日志信息和退出码。

可选的，上述处理器 171 还可以执行如下步骤的程序代码：在主进程接收到任务信息之后，如果主进程被重启，则通过代理进程离线启动任务子进程来执行任务信息，并离线保存任务执行信息。

可选的，上述处理器 171 还可以执行如下步骤的程序代码：任务处理设备通过访问主控设备，来获取当前正在执行的至少一个任务信息，任务信息包括：任务标识和任务执行路径；任务处理设备通过检测在目标位置中记录的任务执行信息，来确定在主进程被重启时至少一个任务信息中已经被执行或未被执行的任务。

可选的，上述处理器 171 还可以执行如下步骤的程序代码：任务处理设备在目标位置进行本地查询，确定是否已经存在与至少一个任务信息中任意一个任务标识相匹配的进程信息；如果存在，则确定在主进程被重启时，任务信息已经被对应的任务子进程启动执行，任务处理设备仅在内存中注册该任务信息；如果不存在，则确定在主进程被重启时，任务信息未被对应的任务子进程所执行，任务处理设备在内存中注册该任务信息，并启动对应的任务子进程执行任务信息。

可选的，上述处理器 171 还可以执行如下步骤的程序代码：任务处理设备在目标位置进行本地查询，确定是否已经存在与已经被对应的任务子进程启动执行的任务信息相匹配的退出码；如果存在，则确定在主进程被重启时，任务信息已经被对应的任务子进程执行结束，任务处理设备终止在内存中注册该任务信息，直接将对应的任务日志反馈给主控设备；如果不存在，则确定在主进程被重启时，任务信息未被对应的任务子进程执行完毕，任务处理设备在内存中注册该任务信息，并启动对应的任务子进程重新执行任务信息。

可选的，上述处理器 171 还可以执行如下步骤的程序代码：从任务信息中提取路径参数，并根据路径参数生成任务执行路径；其中，代理进程根据任务执行路径将任务标

识保存至目标位置。

可选的，上述处理器 171 还可以执行如下步骤的程序代码：任务处理设备的主进程根据任务标识监听任务信息是否被启动执行，并根据退出码监听任务信息是否执行结束。

采用本申请实施例，提供了一种基于分布式系统的任务处理方法。本任务处理设备通过发起任务请求从主控设备中获取主控设备中保存的任务信息，任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息，任务处理设备通过代理进程将任务执行信息保存至目标位置。解决了现有技术中分布式执行系统建立在特定的应用环境中，任务处理设备无法成功执行任务信息，导致任务丢失的技术问题。

本领域普通技术人员可以理解，图 17 所示的结构仅为示意，计算机终端也可以是智能手机（如 Android 手机、iOS 手机等）、平板电脑、掌声电脑以及移动互联网设备（Mobile Internet Devices，MID）、PAD 等终端设备。图 17 其并不对上述电子装置的结构造成限定。例如，计算机终端 A 还可包括比图 17 中所示更多或者更少的组件（如网络接口、显示装置等），或者具有与图 17 所示不同的配置。

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终端设备相关的硬件来完成，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：闪存盘、只读存储器（Read-Only Memory，ROM）、随机存取器（Random Access Memory，RAM）、磁盘或光盘等。

实施例 5

本申请的实施例还提供了一种存储介质。可选地，在本实施例中，上述存储介质可以用于保存上述实施例一所提供的基于分布式系统的任务处理方法所执行的程序代码。

可选地，在本实施例中，上述存储介质可以位于计算机网络中计算机终端群中的任意一个计算机终端中，或者位于移动终端群中的任意一个移动终端中。

可选地，在本实施例中，存储介质被设置为存储用于执行以下步骤的程序代码：任务处理设备通过发起任务请求从主控设备中获取任务信息，其中，在主控设备中保存任务请求对应的任务信息；任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息；任务处理设备通过代理进程将任务执行信息保存至目标位置，其中，任务执行信息包括如下任意一个或多个信息：任务信息的任务标识、任务子进程的进程信息、任务子进程执行任务信息所生成的日志信息和退出码。

可选地，在本实施例中，存储介质被设置为存储用于执行以下步骤的程序代码：在

主进程接收到任务信息之后，如果主进程被重启，则通过代理进程离线启动任务子进程来执行任务信息，并离线保存任务执行信息。

可选地，在本实施例中，存储介质被设置为存储用于执行以下步骤的程序代码：任务处理设备通过访问主控设备，来获取当前正在执行的至少一个任务信息，任务信息包括：任务标识和任务执行路径；任务处理设备通过检测在目标位置中记录的任务执行信息，来确定在主进程被重启时至少一个任务信息中已经被执行或未被执行的任务。

可选地，在本实施例中，存储介质被设置为存储用于执行以下步骤的程序代码：任务处理设备在目标位置进行本地查询，确定是否已经存在与至少一个任务信息中任意一个任务标识相匹配的进程信息；如果存在，则确定在主进程被重启时，任务信息已经被对应的任务子进程启动执行，任务处理设备仅在内存中注册该任务信息；如果不存在，则确定在主进程被重启时，任务信息未被对应的任务子进程所执行，任务处理设备在内存中注册该任务信息，并启动对应的任务子进程执行任务信息。

可选地，在本实施例中，存储介质被设置为存储用于执行以下步骤的程序代码：任务处理设备在目标位置进行本地查询，确定是否已经存在与已经被对应的任务子进程启动执行的任务信息相匹配的退出码；如果存在，则确定在主进程被重启时，任务信息已经被对应的任务子进程执行结束，任务处理设备终止在内存中注册该任务信息，直接将对应的任务日志反馈给主控设备；如果不存在，则确定在主进程被重启时，任务信息未被对应的任务子进程执行完毕，任务处理设备在内存中注册该任务信息，并启动对应的任务子进程重新执行任务信息。

可选地，在本实施例中，存储介质被设置为存储用于执行以下步骤的程序代码：从任务信息中提取路径参数，并根据路径参数生成任务执行路径；其中，代理进程根据任务执行路径将任务标识保存至目标位置。

可选地，在本实施例中，存储介质被设置为存储用于执行以下步骤的程序代码：任务处理设备的主进程根据任务标识监听任务信息是否被启动执行，并根据退出码监听任务信息是否执行结束。

上述本申请实施例序号仅仅为了描述，不代表实施例的优劣。

在本申请的上述实施例中，对各个实施例的描述都各有侧重，某个实施例中没有详述的部分，可以参见其他实施例的相关描述。

在本申请所提供的几个实施例中，应该理解到，所揭露的技术内容，可通过其它的方式实现。其中，以上所描述的装置实施例仅仅是示意性的，例如所述单元的划分，仅

仅为一种逻辑功能划分，实际实现时可以有另外的划分方式，例如多个单元或组件可以结合或者可以集成到另一个系统，或一些特征可以忽略，或不执行。另一点，所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口，单元或模块的间接耦合或通信连接，可以是电性或其它的形式。

所述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单元来实现本实施例方案的目的。

另外，在本申请各个实施例中的各功能单元可以集成在一个处理单元中，也可以是各个单元单独物理存在，也可以两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现，也可以采用软件功能单元的形式实现。

所述集成的单元如果以软件功能单元的形式实现并作为独立的产品销售或使用时，可以存储在一个计算机可读取存储介质中。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或者该技术方案的全部或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中，包括若干指令用以使得一台计算机设备（可为个人计算机、服务器或者网络设备等）执行本申请各个实施例所述方法的全部或部分步骤。而前述的存储介质包括：U 盘、只读存储器（ROM，Read-Only Memory）、随机存取存储器（RAM，Random Access Memory）、移动硬盘、磁碟或者光盘等各种可以存储程序代码的介质。

以上所述仅是本申请的优选实施方式，应当指出，对于本技术领域的普通技术人员来说，在不脱离本申请原理的前提下，还可以做出若干改进和润饰，这些改进和润饰也应视为本申请的保护范围。

权 利 要 求 书

1. 一种基于分布式系统的任务处理方法，其特征在于，包括：

任务处理设备通过发起任务请求从主控设备中获取任务信息，其中，在所述主控设备中保存所述任务请求对应的所述任务信息；

所述任务处理设备的主进程接收到所述任务信息之后，通过触发代理进程启动对应的任务子进程来执行所述任务信息；

所述任务处理设备通过所述代理进程将任务执行信息保存至目标位置，其中，所述任务执行信息包括如下任意一个或多个信息：所述任务信息的任务标识、所述任务子进程的进程信息、所述任务子进程执行所述任务信息所生成的日志信息和退出码。

2. 根据权利要求 1 所述的方法，其特征在于，在所述主进程接收到所述任务信息之后，如果所述主进程被重启，则通过所述代理进程离线启动所述任务子进程来执行所述任务信息，并离线保存所述任务执行信息。

3. 根据权利要求 2 所述的方法，其特征在于，在所述主进程被重启之后，所述方法还包括：

所述任务处理设备通过访问所述主控设备，来获取当前正在执行的至少一个任务信息，所述任务信息包括：所述任务标识和任务执行路径；

所述任务处理设备通过检测在所述目标位置中记录的所述任务执行信息，来确定在所述主进程被重启时所述至少一个任务信息中已经被执行或未被执行的任务。

4. 根据权利要求 3 所述的方法，其特征在于，所述任务处理设备通过检测在所述目标位置中记录的所述任务执行信息，来确定所述至少一个任务信息中已经被执行或未被执行的任务，包括：

所述任务处理设备在所述目标位置进行本地查询，确定是否已经存在与所述至少一个任务信息中任意一个任务标识相匹配的进程信息；

如果存在，则确定在所述主进程被重启时，任务信息已经被对应的任务子进程启动执行，所述任务处理设备仅在内存中注册该任务信息；

如果不存在，则确定在所述主进程被重启时，任务信息未被所述对应的任务子进程所执行，所述任务处理设备在所述内存中注册该任务信息，并启动所述对应的任务子进程执行所述任务信息。

5. 根据权利要求 4 所述的方法，其特征在于，在所述任务处理设备仅在内存中注册该任务信息之前，所述方法还包括：

所述任务处理设备在所述目标位置进行本地查询，确定是否已经存在与已经被对应的任务子进程启动执行的任务信息相匹配的退出码；

如果存在，则确定在所述主进程被重启时，任务信息已经被对应的任务子进程执行结束，所述任务处理设备终止在所述内存中注册该任务信息，直接将对应的任务日志反馈给所述主控设备；

如果不存在，则确定在所述主进程被重启时，任务信息未被所述对应的任务子进程执行完毕，所述任务处理设备在所述内存中注册该任务信息，并启动所述对应的任务子进程重新执行所述任务信息。

6. 根据权利要求 3 所述的方法，其特征在于，在任务处理设备通过发起任务请求从主控设备中获取任务信息之后，所述方法还包括：

从所述任务信息中提取路径参数，并根据所述路径参数生成所述任务执行路径；

其中，所述代理进程根据所述任务执行路径将所述任务标识保存至所述目标位置。

7. 根据权利要求 1 至 6 中任意一项所述的方法，其特征在于，所述任务处理设备的所述主进程根据所述任务标识监听所述任务信息是否被启动执行，并根据所述退出码监听所述任务信息是否执行结束。

8. 一种基于分布式系统的任务处理装置，其特征在于，包括：

第一获取模块，用于任务处理设备通过发起任务请求从主控设备中获取任务信息，其中，在所述主控设备中保存所述任务请求对应的所述任务信息；

第一执行模块，用于所述任务处理设备的主进程接收到所述任务信息之后，通过触发代理进程启动对应的任务子进程来执行所述任务信息；

第一保存模块，用于所述任务处理设备通过所述代理进程将任务执行信息保存至目标位置，其中，所述任务执行信息包括如下任意一个或多个信息：所述任务信息的任务标识、所述任务子进程的进程信息、所述任务子进程执行所述任务信息所生成的日志信息和退出码。

9. 根据权利要求 8 所述的装置，其特征在于，所述装置还包括：

第二执行模块，用于在所述主进程接收到所述任务信息之后，如果所述主进程被重启，则通过所述代理进程离线启动所述任务子进程来执行所述任务信息，并离线保存所述任务执行信息。

10. 根据权利要求 9 所述的装置，其特征在于，所述装置还包括：

第二获取模块，用于所述任务处理设备通过访问所述主控设备，来获取当前正在执

行的至少一个任务信息，所述任务信息包括：所述任务标识和任务执行路径；

第一确定模块，用于所述任务处理设备通过检测在所述目标位置中记录的所述任务执行信息，来确定在所述主进程被重启时所述至少一个任务信息中已经被执行或未被执行的任务。

11. 根据权利要求 10 所述的装置，其特征在于，所述第一确定模块包括：

第二确定模块，用于所述任务处理设备在所述目标位置进行本地查询，确定是否已经存在与所述至少一个任务信息中任意一个任务标识相匹配的进程信息；

第一注册模块，用于如果存在，则确定在所述主进程被重启时，任务信息已经被对应的任务子进程启动执行，所述任务处理设备仅在内存中注册该任务信息；

第二注册模块，用于如果不存在，则确定在所述主进程被重启时，任务信息未被所述对应的任务子进程所执行，所述任务处理设备在所述内存中注册该任务信息，并启动所述对应的任务子进程执行所述任务信息。

12. 根据权利要求 11 所述的装置，其特征在于，所述装置还包括：

第三确定模块，用于所述任务处理设备在所述目标位置进行本地查询，确定是否已经存在与已经被对应的任务子进程启动执行的任务信息相匹配的退出码；

反馈模块，用于如果存在，则确定在所述主进程被重启时，任务信息已经被对应的任务子进程执行结束，所述任务处理设备终止在所述内存中注册该任务信息，直接将对应的任务日志反馈给所述主控设备；

第三注册模块，用于如果不存在，则确定在所述主进程被重启时，任务信息未被所述对应的任务子进程执行完毕，所述任务处理设备在所述内存中注册该任务信息，并启动所述对应的任务子进程重新执行所述任务信息。

13. 根据权利要求 10 所述的装置，其特征在于，所述装置还包括：

生成模块，用于从所述任务信息中提取路径参数，并根据所述路径参数生成所述任务执行路径；

第二保存模块，用于所述代理进程根据所述任务执行路径将所述任务标识保存至所述目标位置。

14. 根据权利要求 8 至 13 中任意一项所述的装置，其特征在于，所述装置还包括：

监听模块，用于所述任务处理设备的所述主进程根据所述任务标识监听所述任务信息是否被启动执行，并根据所述退出码监听所述任务信息是否执行结束。

15. 一种基于分布式系统的任务处理系统，其特征在于，包括：

主控设备，用于存储任务信息；

任务处理设备，与所述主控设备进行通讯，用于通过发起任务请求从所述主控设备中获取任务信息，其中，在所述主控设备中保存所述任务请求对应的所述任务信息；

所述任务处理设备还用于所述任务处理设备的主进程接收到所述任务信息之后，通过触发代理进程启动对应的任务子进程来执行所述任务信息；

所述任务处理设备还用于通过所述代理进程将任务执行信息保存至目标位置，其中，所述任务执行信息包括如下任意一个或多个信息：所述任务信息的任务标识、所述任务子进程的进程信息、所述任务子进程执行所述任务信息所生成的日志信息和退出码。

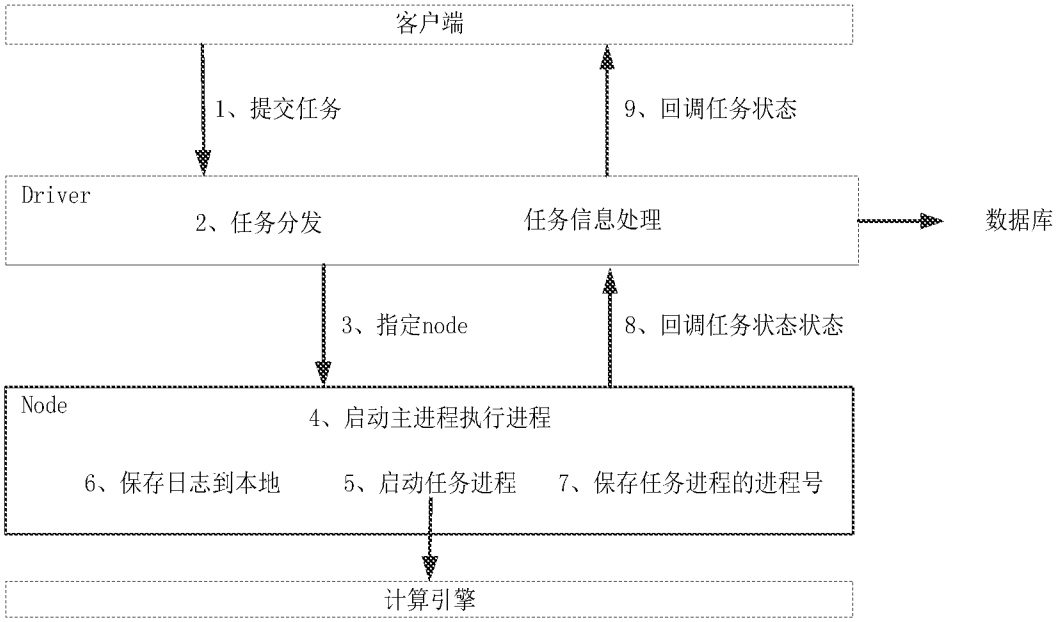

客户端
1、提交任务
9、回调任务状态
Driver
2、任务分发
任务信息处理
数据库
3、指定node
8、回调任务状态状态
Node
4、启动主进程执行进程
6、保存日志到本地
5、启动任务进程
7、保存任务进程的进程号
计算引擎

图 1

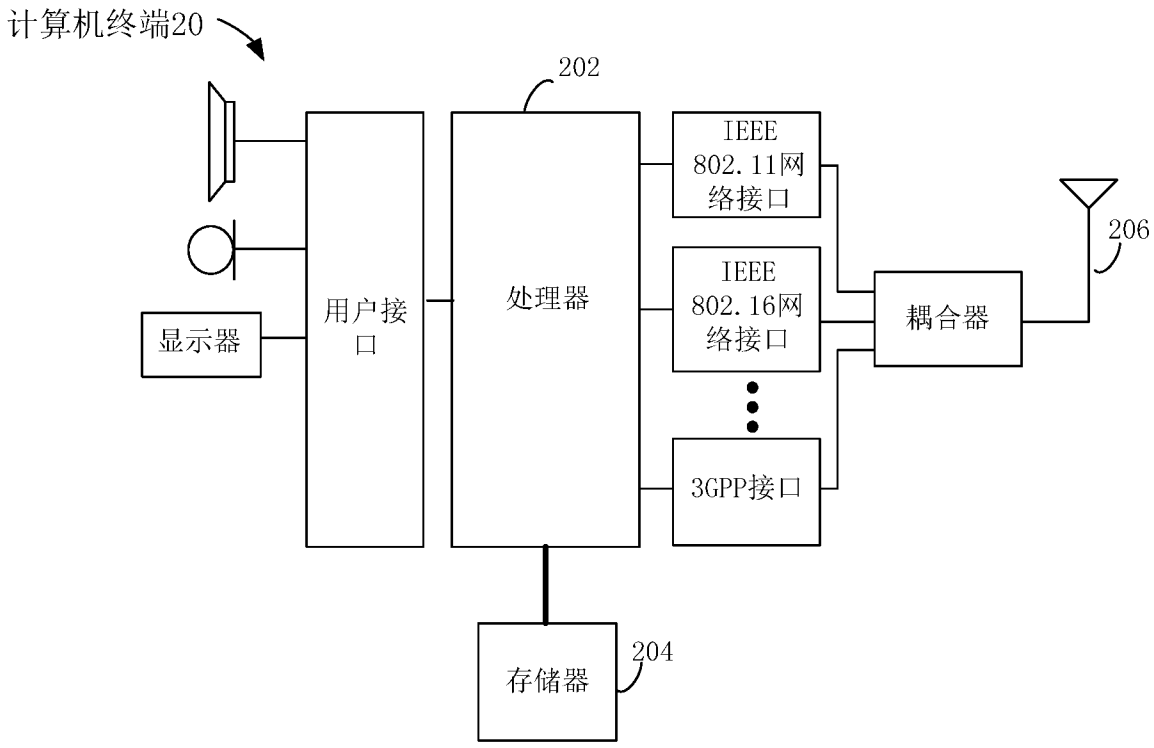

计算机终端20
202
IEEE 802.11网络接口
IEEE 802.16网络接口
3GPP接口
206
耦合器
显示器
用户接口
处理器
存储器
204

图 2

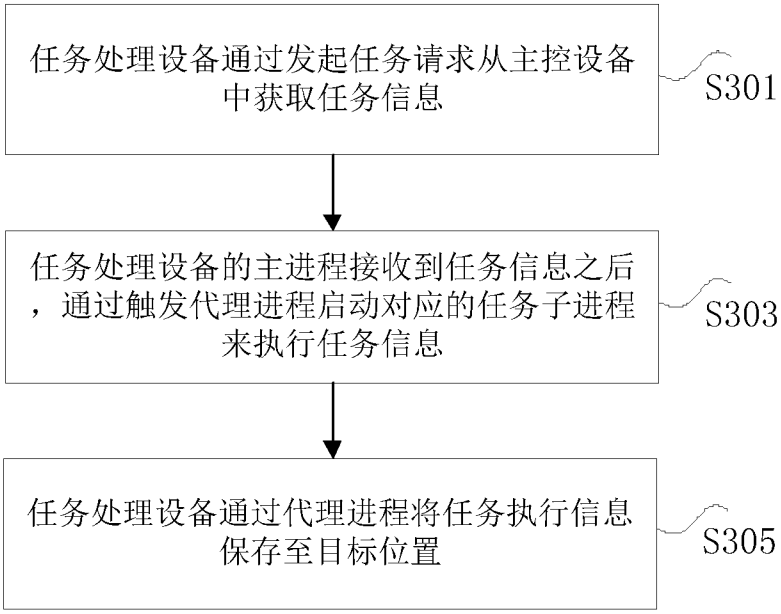

任务处理设备通过发起任务请求从主控设备中获取任务信息
S301
任务处理设备的主进程接收到任务信息之后，通过触发代理进程启动对应的任务子进程来执行任务信息
S303
任务处理设备通过代理进程将任务执行信息保存至目标位置
S305

图 3

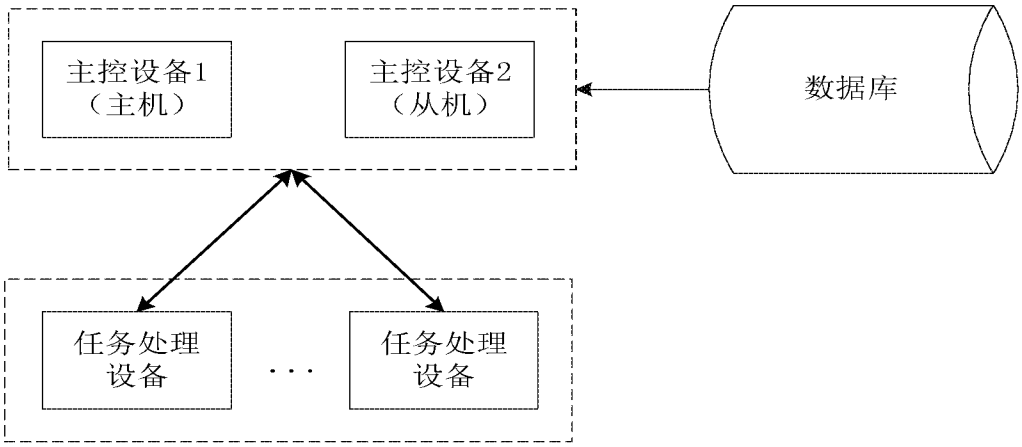


图 4

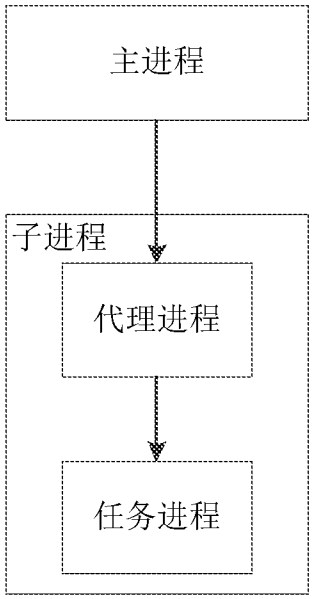


图 5

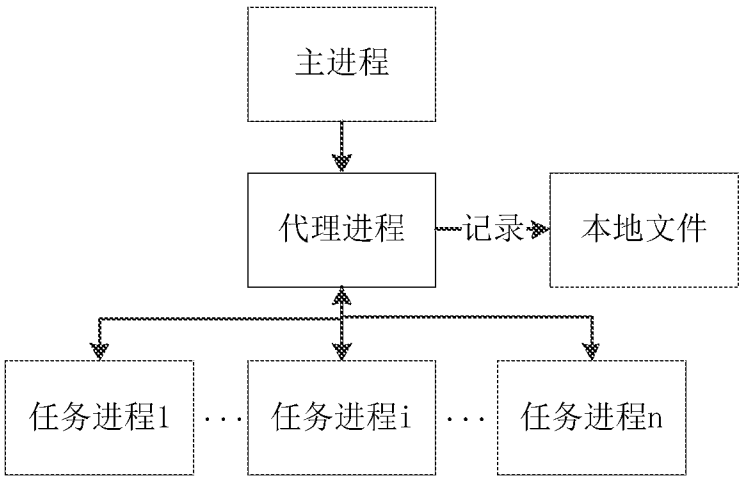

主进程
代理进程
记录
本地文件
任务进程1
任务进程i
任务进程n

图 6

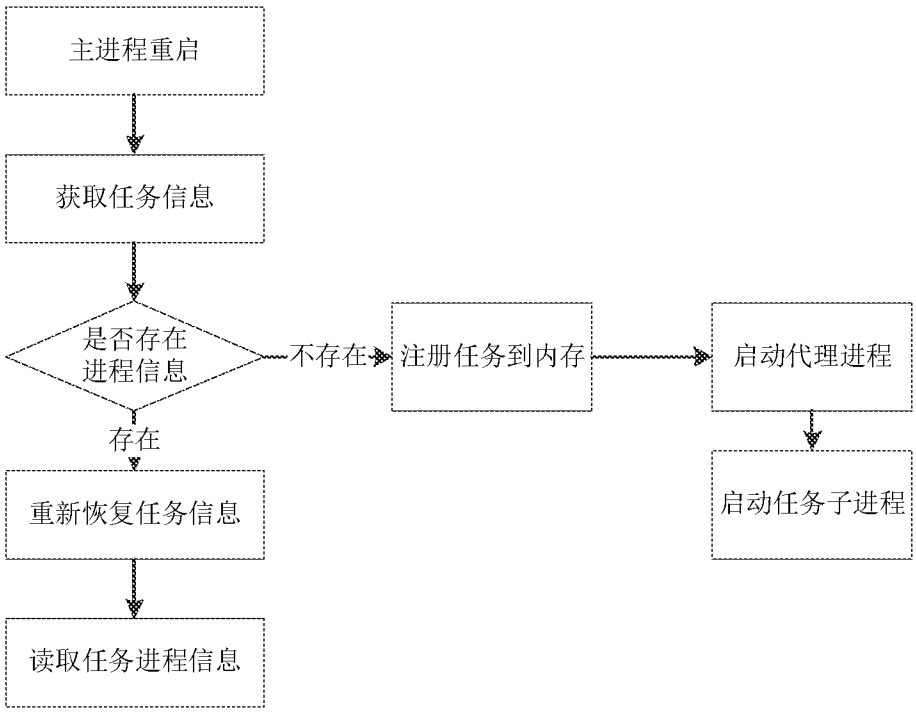

主进程重启
获取任务信息
是否存在
进程信息
不存在
注册任务到内存
启动代理进程
启动任务子进程
存在
重新恢复任务信息
读取任务进程信息

图 7

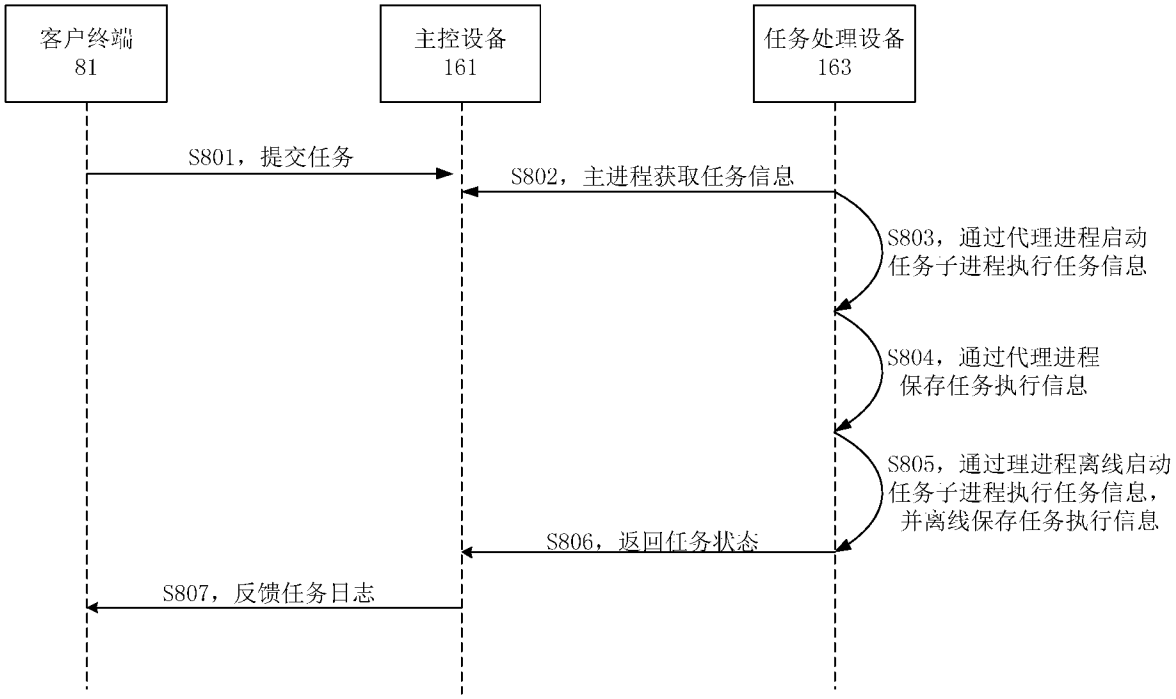

客户终端
81
主控设备
161
任务处理设备
163
S801，提交任务
S802，主进程获取任务信息
S803，通过代理进程启动
任务子进程执行任务信息
S804，通过代理进程
保存任务执行信息
S805，通过理进程离线启动
任务子进程执行任务信息，
并离线保存任务执行信息
S806，返回任务状态
S807，反馈任务日志

图 8

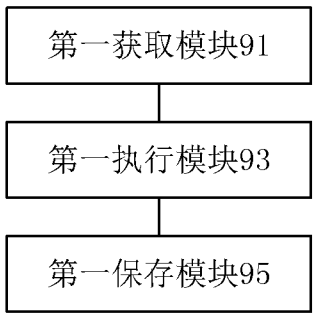

第一获取模块91
第一执行模块93
第一保存模块95

图 9

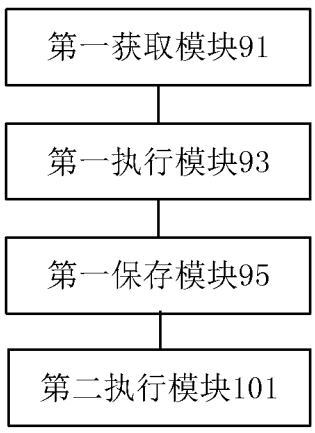

第一获取模块91
第一执行模块93
第一保存模块95
第二执行模块101

图 10

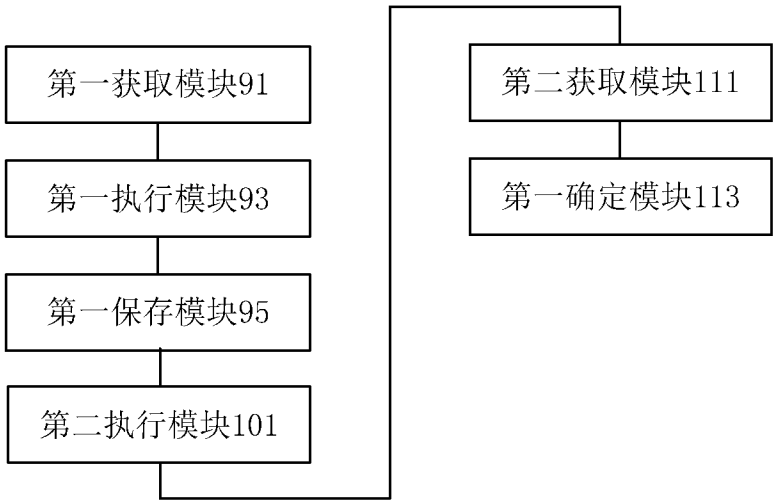

第一获取模块91
第一执行模块93
第一保存模块95
第二执行模块101
第二获取模块111
第一确定模块113

图 11

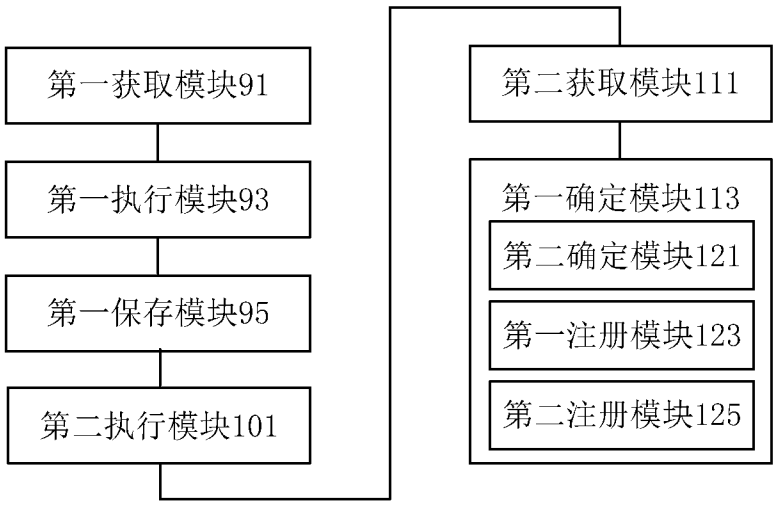

第一获取模块91
第一执行模块93
第一保存模块95
第二执行模块101
第二获取模块111
第一确定模块113
第二确定模块121
第一注册模块123
第二注册模块125

图 12

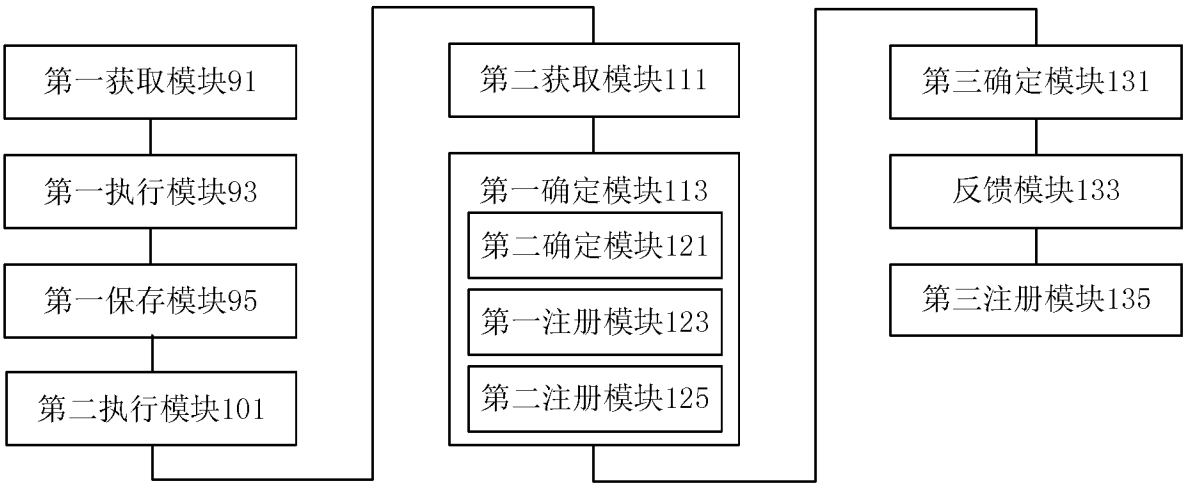

第一获取模块91
第一执行模块93
第一保存模块95
第二执行模块101
第二获取模块111
第一确定模块113
第二确定模块121
第一注册模块123
第二注册模块125
第三确定模块131
反馈模块133
第三注册模块135

图 13

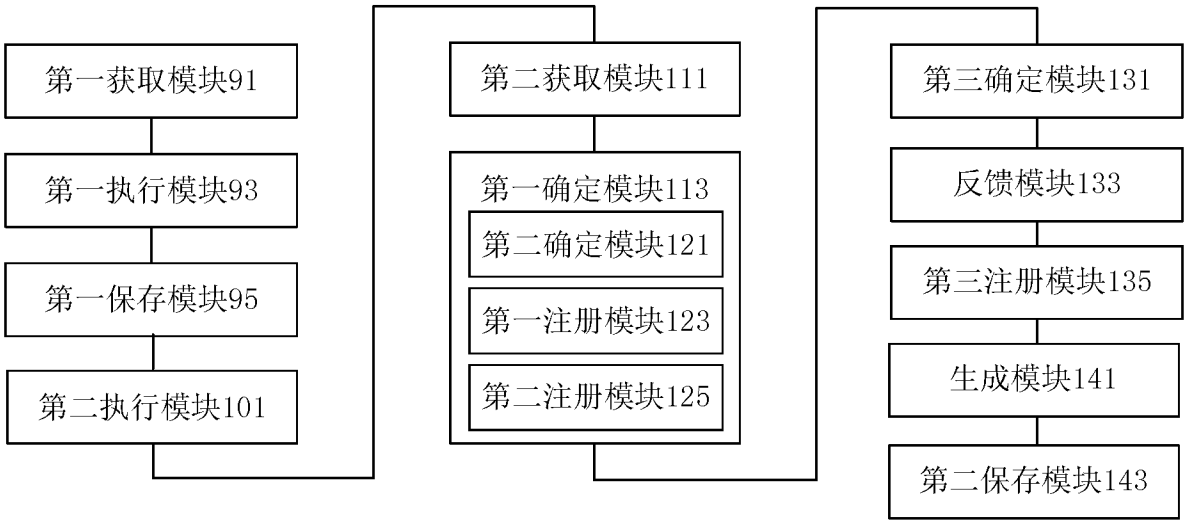

第一获取模块91
第一执行模块93
第一保存模块95
第二执行模块101
第二获取模块111
第一确定模块113
第二确定模块121
第一注册模块123
第二注册模块125
第三确定模块131
反馈模块133
第三注册模块135
生成模块141
第二保存模块143

图 14

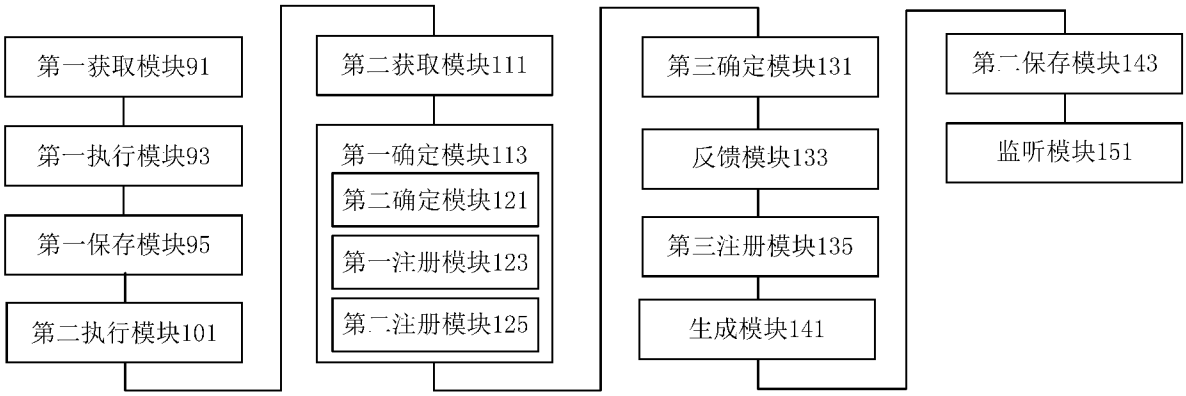

第一获取模块91
第一执行模块93
第一保存模块95
第二执行模块101
第二获取模块111
第一确定模块113
第二确定模块121
第一注册模块123
第二注册模块125
第三确定模块131
反馈模块133
第三注册模块135
生成模块141
第二保存模块143
监听模块151

图 15

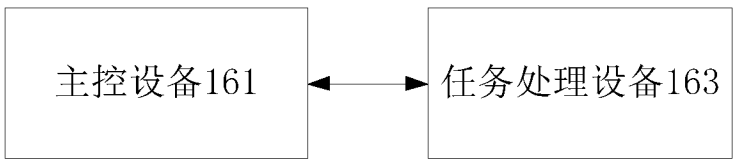

主控设备161
任务处理设备163

图 16

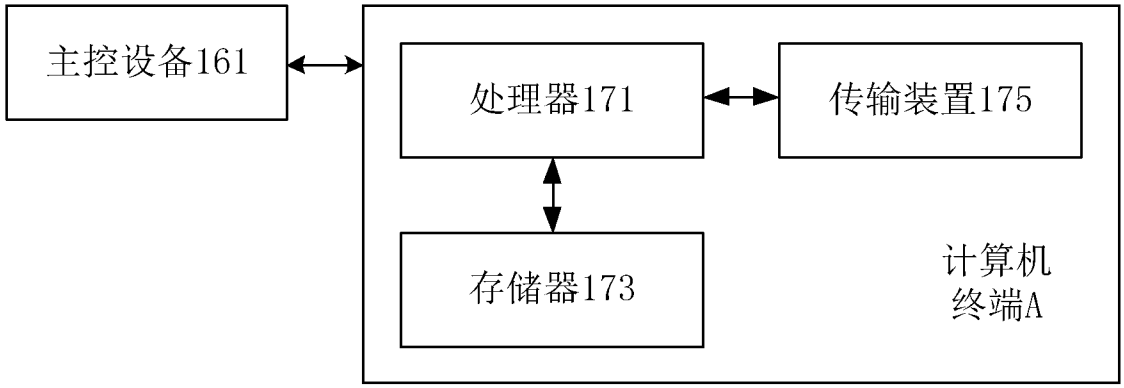


图 17

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2017/073721

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/48 (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

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)

CNKI, CNPAT, WPI, EPODOC, GOOGLE: parent node, main control equipment, child node, task processing equipment, main process, parent process, daemon process, subprocess, agent, distributed, server, node, parent, child, host, process, task, job, broker, restart

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	CN 103677973 A (XI'AN HEAVY EQUIPMENT MANUFACTURING GROUP WEINAN OPTOELECTRIONICS TECHNOLOGY CO., LTD.), 26 March 2014 (26.03.2014), description, paragraphs [0008]-[0009] and [0035]-[0037]	1, 7, 8, 14, 15
Y	CN 104331327 A (SHANDONG MASSCLOUDS CO., LTD.), 04 February 2015 (04.02.2015), description, paragraphs [0074]-[0091], and figures 1-4	1, 7, 8, 14, 15
A	CN 102929707 A (JIANGNAN INSTITUTE OF COMPUTING TECHNOLOGY), 13 February 2013 (13.02.2013), the whole document	1-15
A	US 2015378406 A1 (FUJITSU LIMITED), 31 December 2015 (31.12.2015), the whole document	1-15

☐ 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
03 May 2017 (03.05.2017)	**19 May 2017 (19.05.2017)**
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 **SU, Wen** Telephone No.: (86-10) **62413821**

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2017/073721

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN 103677973 A	26 March 2014	None	
CN 104331327 A	04 February 2015	None	
CN 102929707 A	13 February 2013	None	
US 2015378406 A1	31 December 2015	JP 2016012347 A	21 January 2016
		EP 2960791 A1	30 December 2015

国际检索报告

国际申请号
PCT/CN2017/073721

A. 主题的分类

G06F 9/48(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)

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））

CNKI，CNPAT，WPI，EPODOC，GOOGLE：分布式，服务器，父节点，主控设备，子节点，任务处理设备，主进程，父进程，守护进程，子进程，任务，代理，重启，distributed，server，node，parent，child，host，process，task，job，broker，restart

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
Y	CN 103677973 A (西安重装渭南光电科技有限公司) 2014年 3月 26日 (2014-03-26) 说明书第[0008]-[0009],[0035]-[0037]段	1,7,8,14,15
Y	CN 104331327 A (山东乾云启创信息科技有限公司) 2015年 2月 4日 (2015-02-04) 说明书第[0074]-[0091]段、图1-4	1,7,8,14,15
A	CN 102929707 A (无锡江南计算技术研究所) 2013年 2月 13日 (2013-02-13) 全文	1-15
A	US 2015378406 A1 (FUJITSU LIMITED) 2015年 12月 31日 (2015-12-31) 全文	1-15

☐ 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
2017年 5月 3日	2017年 5月 19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 苏文 电话号码 (86-10)62413821

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2017/073721

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 103677973 A	2014年 3月 26日	无	
CN 104331327 A	2015年 2月 4日	无	
CN 102929707 A	2013年 2月 13日	无	
US 2015378406 A1	2015年 12月 31日	JP 2016012347 A	2016年 1月 21日
		EP 2960791 A1	2015年 12月 30日