

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局

(43) 国际公布日
2020 年 8 月 20 日 (20.08.2020)



(10) 国际公布号
WO 2020/164265 A1

(51) 国际专利分类号:
G06F 9/455 (2006.01)

(21) 国际申请号: PCT/CN2019/116704

(22) 国际申请日: 2019 年 11 月 8 日 (08.11.2019)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201910110281.9 2019 年 2 月 11 日 (11.02.2019) CN

(71) 申请人: 平安科技(深圳)有限公司(PING AN TECHNOLOGY (SHENZHEN) CO., LTD.) [CN/CN];
中国广东省深圳市福田区福田街道福安社区益田路 5033 号平安金融中心 23 楼, Guangdong 518000 (CN)。

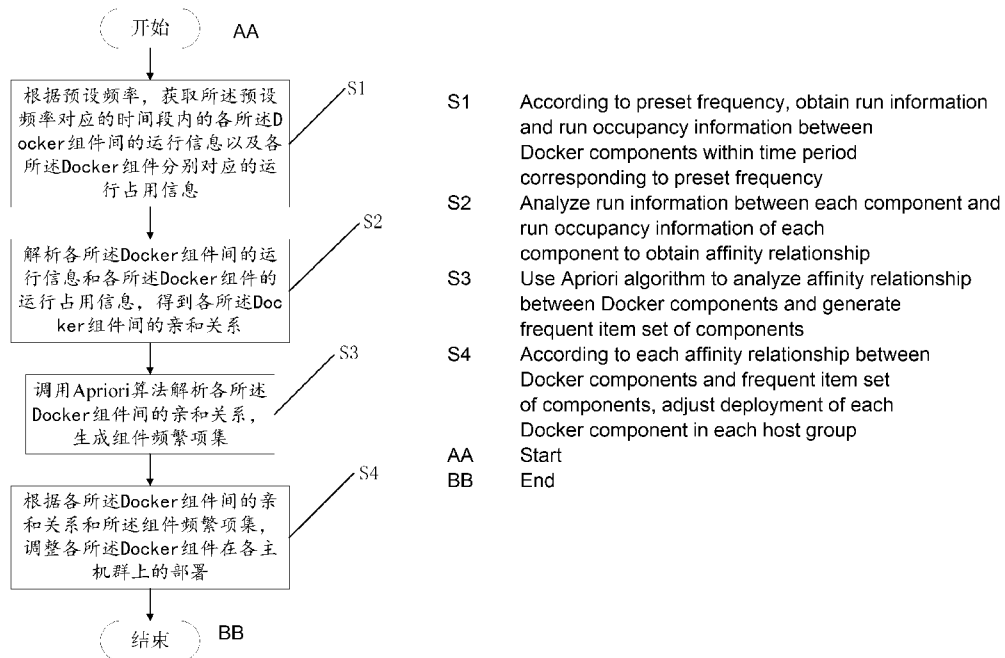
(72) 发明人: 赵骏(ZHAO, Jun); 中国广东省深圳市福田区福田街道福安社区益田路 5033 号平安金融中心 23 楼, Guangdong 518000 (CN)。

(74) 代理人: 深圳市明日今典知识产权代理有限公司(普通合伙)(SHENZHEN MINGRIJINDIAN INTELLECTUAL PROPERTY AGENCY FIRM (GENERAL)); 中国广东省深圳市南山区南头街道安乐社区关口二路 15 号智恒产业园 30 栋 4 层 406 室, Guangdong 518000 (CN)。

(81) 指定国(除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK,

(54) Title: METHOD AND RELATED DEVICE FOR ADJUSTING DEPLOYMENT OF DOCKER COMPONENT ON BASIS OF AFFINITY

(54) 发明名称: 基于亲和性调整Docker组件部署的方法和相关设备



(57) Abstract: The present application provides a method, apparatus, computer device, and readable storage medium for adjusting the deployment of a Docker component on the basis of affinity, relating to the technical field of distributed deployment, said method comprising: according to a preset frequency, obtaining run information and run occupancy information between Docker components within a time period corresponding to the preset frequency; analyzing the run information between each component and the run occupancy information of each component to obtain an affinity relationship; generating a frequent item set of components according to each

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, 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))。

affinity relationship; according to each affinity relationship and the frequent item set of components, adjusting the deployment of each component in each host group. In the present application, the run information and run occupancy information of each Docker component in a system is monitored according to a preset frequency, and the affinity relationship between each component is automatically parsed; according to the priority and the affinity between the components, the deployment of various components on each host group is adjusted such that components having higher affinity may be deployed on the same geographical host group, thus facilitating data communication transmission and execution efficiency.

(57) 摘要: 本申请提供一种基于亲和性调整Docker组件部署的方法、装置、计算机设备和可读存储介质, 涉及分布式部署技术领域, 方法包括: 根据预设频率, 获取预设频率对应时间段内各Docker组件间的运行信息和运行占用信息; 解析各组件间的运行信息和各组件的运行占用信息, 得到亲和关系; 根据各亲和关系生成组件频繁项集; 根据各亲和关系和组件频繁项集, 调整各组件在各主机群的部署。本申请通过根据预设频率监控系统中各Docker组件的运行信息、运行占用信息, 自动解析得到各组件间的亲和关系, 再根据优先级和组件间的亲和关系, 调整各组件在各主机群上的部署, 使得具有较高亲和性的组件可以部署在相同的地域的主机群上, 有利于数据通信传输和执行效率。

基于亲和性调整Docker组件部署的方法和相关设备

本申请要求于2019年2月11日提交中国专利局、申请号为201910110281.9，发明名称为“基于亲和性调整Docker组件部署的方法和相关设备”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

本申请涉及分布式部署技术领域，特别涉及一种基于亲和性调整 Docker 组件部署的方法和相关设备。

背景技术

在大型的金融系统中，常常使用 Docker 容器来部署微服务的分布式系统。分布式系统中常常涉及到十几个，甚至是几十个组件，各组件设置在 Docker 容器中，基于 Docker 容器来运行分布式系统。各 Docker 组件的运行需要高效的交互以及合理的资源利用。因此，需要解析出各组件间的亲和关系，以便用于管理哪些 Docker 组件部署在一起，哪些 Docker 组件在部署时需要相互隔离。其中，亲和关系是指组件间的亲和性的关系，亲和性用以表现组件间的内在联系是否密切，比如根据组件与组件间运行时的相互调用和资源共享等情况是否频繁，从而判断组件间的亲和性。现有用户通过 k8s (Kubernetes) 这一内部容器管理系统来实现根据亲和关系对 Docker 组件的部署。但是，通过 k8s 进行 Docker 组件部署时，各组件的亲和关系需要用户通过手动设置非常详细的标签和配置，然后 k8s 才能根据各组件的标签和配置进行组件间亲和关系的确定。并且，配置和标签一旦确定后，组件间的亲和关系将不能被调整，除非对系统进行重启，才能再次完成组件的部署。

技术问题

本申请的主要目的为提供一种基于亲和性调整 Docker 组件部署的方法和相关设备，旨在解决现有 Docker 组件间的亲和关系需要手动设置以及 Docker 组件的部署不具有即时性的弊端。

技术解决方案

为实现上述目的，本申请提供了一种基于亲和性调整Docker组件部署的方法，所述Docker组件部署于主机群中，所述主机群为设置在同一地区的所有所述主机形成的集群，所述方法包括：

根据预设频率，获取所述预设频率对应的时间段内的各所述Docker组件间的运行信息以及各所述Docker组件分别对应的运行占用信息，其中，所述运行

信息为各所述Docker组件间的调用关系信息和资源共享信息，所述调用关系信息为组件与组件间相互对应的调用关系和调用次数，所述资源共享信息为组件与组件间在运行时共同调用的资源的对应信息；

解析各所述Docker组件间的运行信息和各所述Docker组件的运行占用信息，得到各所述Docker组件间的亲和关系；

调用Apriori算法解析各所述Docker组件间的亲和关系，生成组件频繁项集，其中，所述组件频繁项集为包含各所述Docker组件间的亲和关系的集合；

根据各所述Docker组件间的亲和关系和所述组件频繁项集，调整各所述Docker组件在各主机群上的部署。

本申请还提供了一种基于亲和性调整Docker组件部署的装置，所述Docker组件部署于主机群中，所述主机群为设置在同一地区的所有所述主机形成的集群，所述装置包括：

获取模块，用于根据预设频率，获取所述预设频率对应的时间段内的各所述Docker组件间的运行信息以及各所述Docker组件分别对应的运行占用信息，其中，所述运行信息为各所述Docker组件间的调用关系信息和资源共享信息，所述调用关系信息为组件与组件间相互对应的调用关系和调用次数，所述资源共享信息为组件与组件间在运行时共同调用的资源的对应信息；

解析模块，用于解析各所述Docker组件间的运行信息和各所述Docker组件的运行占用信息，得到各所述Docker组件间的亲和关系；

第一生成模块，用于调用Apriori算法解析各所述Docker组件间的亲和关系，生成组件频繁项集，其中，所述组件频繁项集为包含各所述Docker组件间的亲和关系的集合；

调整模块，用于根据各所述Docker组件间的亲和关系和所述组件频繁项集，调整各所述Docker组件在各主机群上的部署。

本申请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任一项所述方法的步骤。

本申请还提供一种计算机可读存储介质，其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任一项所述的方法的步骤。

有益效果

本申请中提供的一种基于亲和性调整Docker组件部署的方法、装置和计算

机设备，通过根据预设频率监控系统中各Docker组件的运行信息和运行占用信息，从而自动解析得到各组件间的亲和关系。再根据预设的优先级和组件间的亲和关系，动态调整各组件在各不同地区的主机群上的部署，从而使得具有高亲和性的组件可以部署在相同的地域的主机群上，有利于数据通信传输和执行效率。

附图说明

图1是本申请一实施例中亲和性调整Docker组件部署的方法步骤示意图；

图2是本申请一实施例中亲和性调整Docker组件部署的装置整体结构框图；

图3是本申请一实施例的计算机设备的结构示意图。

本申请目的的实现、功能特点及优点将结合实施例，参照附图做进一步说明。

本发明的最佳实施方式

为了使本申请的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本申请进行进一步详细说明。应当理解，此处描述的具体实施例仅仅用以解释本申请，并不用于限定本申请。

参照图1，本申请一实施例中提供了一种基于亲和性调整Docker组件部署的方法，所述Docker组件部署于主机群中，所述主机群为设置在同一地区的所有所述主机形成的集群，所述方法包括：

S1：根据预设频率，获取所述预设频率对应的时间段内的各所述Docker组件间的运行信息以及各所述Docker组件分别对应的运行占用信息，其中，所述运行信息为各所述Docker组件间的调用关系信息和资源共享信息，所述调用关系信息为组件与组件间相互对应的调用关系和调用次数，所述资源共享信息为组件与组件间在运行时共同调用的资源的对应信息；

S2：解析各所述Docker组件间的运行信息和各所述Docker组件的运行占用信息，得到各所述Docker组件间的亲和关系；

S3：调用Apriori算法解析各所述Docker组件间的亲和关系，生成组件频繁项集，其中，所述组件频繁项集为包含各所述Docker组件间的亲和关系的集合；

S4：根据各所述Docker组件间的亲和关系和所述组件频繁项集，调整各所述Docker组件在各主机群上的部署。

本实施例中，各主机群中的监控服务实时监控docker容器中各docker组件间的运行信息和各docker组件分别对应的运行占用信息。其中，运行信息包括各d

ocker组件间的调用关系信息和资源共享信息；调用关系信息包括各docker组件间相互对应的调用关系和调用次数，由于docker容器组件，即docker组件中的服务在相互调用时，需要将调用请求先发送到API Gateway组件，由API Gateway组件负责将调用请求转发到目的端组件的服务，因此调整系统可以通过监控API Gateway组件中调用请求的转发情况，得到各调用请求对应的发起端组件和目的端组件，从而得到各组件间的相互对应的调用关系，并且通过统计各具有调用关系的调用请求的转发数量，得到具有调用关系的两个组件对应的调用次数；资源共享信息为组件与组件间在运行时共同调用的资源的信息，调整系统通过网络监控可以获取docker容器组件中服务访问资源，并根据访问的资源请求端得到具有资源共享的情况的两个组件，即组件与组件间在运行时公共调用的资源的对应信息；运行占用信息包括组件服务在运行时的CPU占用率和内存占用量，调整系统通过docker容器中的监控服务可以实时监控得到。调整系统根据预设频率调整各docker组件在各主机群上的部署。因此，调整系统根据预设频率，从各主机群实时监控的所有信息中截取预设频率对应的时间段的各docker组件间的运行信息和各docker组件分别对应的运行占用信息进行下一步的解析。其中，各docker组件分别对应的运行占用信息为预设频率对应的时间段内的平均值。调整系统首先根据各docker组件间的调用次数和预设频率对应的时间段，计算得到各docker组件的调用频率。然后，将各docker组件间的调用频率、各docker组件间的资源共享信息和各docker组件的运行占用信息输入预先构建的亲关系库中。其中，亲关系库中存储有组件间的调用频率、组件间的资源共享信息、组件的运行占用信息与亲关系映射关系表，调用系统根据映射关系表匹配得到各docker组件间的亲关系。其中，亲关系包括亲和信息以及亲和信息对应的两个组件，亲和信息包括亲和、较亲和、违亲和和反亲和。在进行匹配的过程中，部分组件与其它组件不具有任何的亲关系，具体表示为“无亲和关系”。调整系统根据匹配后得到的各docker组件间的亲关系，从各组件中筛选得到与其它组件具有确定的亲关系的各个第一组件，即从所有组件中除去“无亲和关系”的部分组件，得到各个第一组件。调整系统根据筛选得到的各个第一组件以及各组件间的亲关系，生成各第一组件的亲关系对应过的亲和性集。其中，亲和性集为具有确定的亲关系的两个第一组件和对应的亲和信息组成的集合，比如，[组件1，组件2，亲和]。调整系统通过Apriori算法解析所有的亲和性集，统计得到各个第一组件的出现次数以及第一组

件的各个亲和信息对应的出现次数，并根据各个第一组件、各个第一组件的出现次数以及第一组件的各亲和信息对应过的出现次数，生成组件频繁项集。比如，各第一组件的亲和性集为：{[组件1，组件2，亲和]，[组件2，组件3，反亲和]，[组件2，组件4，较亲和]}，经过解析后生成的组件频繁项集为：{[组件1，出现次数：1次，亲和：1次，较亲和：0次，违亲和：0次，反亲和：0次]，[组件2，出现次数：3次，亲和：1次，较亲和：1次，违亲和：0次，反亲和：1次]，[组件3，出现次数：1次，亲和：0次，较亲和：0次，违亲和：0次，反亲和：1次]，[组件4，出现次数：1次，亲和：0次，较亲和：1次，违亲和：0次，反亲和：0次]}。调整系统首先根据预先设定的优先级，从组件频繁项集中筛选出优先级第一的第一组件，并将优先级第一的第一组件部署在第一主机群。其中，预先设定的优先级为：出现次数>亲和>较亲和>违亲和>反亲和，各项中又以对应的数值的大小降序排列；第一主机群为各主机群中的任意一台。然后，调整系统根据优先级，从组件频繁项集中筛选出优先级第二的第一组件，并根据亲和关系判断优先级第二的第一组件和优先级第一的第一组件之间是否为亲和或较亲和，即是否符合预设条件，以完成预设条件的判断动作。若符合，则将优先级第二的第一组件部署在第一主机群；若不符合，则将优先级第二的第一组件部署在第二主机群，已完成部署组件动作。其中，第二主机群为各主机群中除去第一主机群之外的任一主机群，同一主机群为设置在同一地区的所有主机群形成的集群。调整系统依次按照优先级降序从组件频繁项集中筛选第一组件，并逐一与已部署在主机群中的第一组件执行预设条件的判断动作，直至组件频繁项集中的所有第一组件均部署完成。

进一步的，所述运行占用信息包括CPU占用率和内存占用量，所述解析各所述Docker组件间的运行信息和各所述Docker组件的运行占用信息，得到各所述Docker组件间的亲和关系的步骤，包括：

S201：根据各所述Docker组件间的调用次数和所述预设频率对应的时间段，计算得到各Docker组件间的调用频率；

S202：将各所述Docker组件间的调用频率、各所述Docker组件间的资源共享信息和各所述Docker组件的运行占用信息输入预先构建的亲和关系库中，匹配得到各所述Docker组件间的亲和关系，其中，所述亲和关系库中存储有组件间的调用频率、组件间的资源共享信息、组件的运行占用信息与亲和关系映射关系表。

本实施例中，调整系统将各Docker组件间的调用次数和预设频率对应的时间段输入预设的计算公式中，计算得到各Docker组件间的调用频率。其中，计算公式为 $F=N/S$ ， F 为调用频率， N 为调用次数， S 为预设频率对应的时间段，比如，预设频率为30分钟/次，则 S 为30分钟。在得到各Docker组件间的调用频率后，调整系统将各Docker组件间的调用频率、各Docker组件间的资源共享信息和各Docker组件的运行占用信息输入预先构建的亲和相关性库中，根据亲和相关性库中存储的组件间的调用频率、组件间的资源共享信息、组件的运行占用信息与亲和相关性映射关系表，匹配得到各组件间的亲和相关性。其中，映射关系表为：

调用 频率 ≥ 2	资源 共享	CPU占用率不超过 30%或者内存占用不超过 512Mb	亲和
调用 频率 ≥ 2	无资 源共享	CPU占用率不超过 30%或者内存占用不超过 512Mb	较亲和
调用 频率 ≥ 2	资源 共享	CPU占用率超过30% 或者内存占用超过512Mb	较亲和
调用 频率 ≥ 2	无资 源共享	CPU占用率超过30% 或者内存占用超过512Mb	违亲和
调用 频率 < 2	资源 共享	CPU占用率不超过 30%或者内存占用不超过 512Mb	较亲和
调用 频率 < 2	无资 源共享	CPU占用率不超过 30%或者内存占用不超过 512Mb	违亲和
调用 频率 < 2	资源 共享	CPU占用率超过30% 或者内存占用超过512Mb	违亲和
调用 频率 < 2	无资 源共享	CPU占用率超过30% 或者内存占用超过512Mb	反亲和
调用 频率=0	/	/	无亲和关 系

进一步的，所述Docker组件间的亲和关系包括两个Docker组件对应的亲和信息，所述调用Apriori算法解析各所述Docker组件间的亲和关系，生成组件频繁项集的步骤，包括：

S301：根据各所述Docker组件间的亲和关系，从各所述Docker组件中筛选得到与其他Docker组件具有确定的亲和关系的各个第一组件；

S302：根据各所述第一组件以及各所述Docker组件间的亲和关系，生成各所述第一组件间的亲和关系对应的亲和性集；

S303：根据各所述亲和性集，统计得到各所述第一组件的出现次数以及各所述第一组件对应的各亲和信息对应的出现次数；

S304：根据各所述第一组件、各所述第一组件的出现次数以及各所述第一组件的各亲和信息对应的出现次数，生成所述组件频繁项集。

本实施例中，调整系统根据匹配得到的各Docker组件间的亲和关系中，包括两个Docker组件对应的亲和信息，比如，亲和关系为：组件1，组件2，亲和，其中“亲和”为亲和信息。调整系统从各Docker组件中筛选得到与其它Docker组件具有确定的亲和关系的各个第一组件，即从所有Docker组件中除去“无亲和关系”的部分Docker组件，得到各个第一组件。调整系统根据筛选得到的各个第一组件以及各Docker组件间的亲和关系，生成各第一组件的亲和关系对应的亲和性集。其中，亲和性集为具有确定的亲和关系的两个第一组件和对应的亲和信息组成的集合，比如，亲和性集为：[组件1，组件2，亲和]。调整系统通过Apriori算法解析所有的亲和性集，统计得到各个第一组件的出现次数以及第一组件的各个亲和信息对应的出现次数，并根据各个第一组件、各个第一组件的出现次数以及第一组件的各亲和信息对应过的出现次数，生成组件频繁项集。比如，各第一组件的亲和性集为：{[组件1，组件2，亲和]，[组件2，组件3，反亲和]，[组件2，组件4，较亲和]}，经过解析后生成的组件频繁项集为：{[组件1，出现次数：1次，亲和：1次，较亲和：0次，违亲和：0次，反亲和：0次]，[组件2，出现次数：3次，亲和：1次，较亲和：1次，违亲和：0次，反亲和：1次]，[组件3，出现次数：1次，亲和：0次，较亲和：0次，违亲和：0次，反亲和：1次]，[组件4，出现次数：1次，亲和：0次，较亲和：1次，违亲和：0次，反亲和：0次]}。

进一步的，所述亲和信息包括亲和、较亲和、违亲和以及反亲和，所述根据各所述Docker组件间的亲和关系和所述组件频繁项集，调整各所述Docker组

件在各主机群上的部署的步骤，包括：

S401：根据预先设定的优先级，从所述组件频繁项集中筛选出优先级第一的第一组件，并将所述优先级第一的第一组件部署在第一主机群，所述第一主机群为各所述主机群中的任一主机群；

S402：根据预先设定的优先级，从所述组件频繁项集中筛选出优先级第二的第一组件，并判断所述优先级第二的第一组件和所述优先级第一的第一组件之间的亲和关系是否符合预设条件，以完成预设条件的判断动作，其中，所述预设条件为组件之间的亲和关系为亲和或较亲和；

S403：若符合，则将所述优先级第二的第一组件部署在所述第一主机群；若不符合，则将优先级第二的第一组件部署在第二主机群，以完成部署组件动作，其中，所述第二主机群为各所述主机群中除所述第一主机群之外的任一主机群；

S404：依次按照优先级降序从所述组件频繁项集中筛选第一组件，并与当前已部署在主机群中的各第一组件逐一执行所述预设条件的判断动作和所述部署组件动作，直至所述组件频繁项集中的所有第一组件均部署完成。

本实施例中，调整系统首先根据预先设定的优先级，从组件频繁项集中筛选出优先级第一的第一组件，并将优先级第一的第一组件部署在第一主机群。其中，预先设定的优先级为：出现次数>亲和>较亲和>违亲和>反亲和，各项中又以对应的数值的大小降序排列；第一主机群为各主机群中的任意一台。然后，调整系统根据优先级，从组件频繁项集中筛选出优先级第二的第一组件，并根据亲和关系判断优先级第二的第一组件和优先级第一的第一组件之间是否为亲和或较亲和，即是否符合预设条件，以完成预设条件的判断动作。若符合，则将优先级第二的第一组件部署在第一主机群；若不符合，则将优先级第二的第一组件部署在第二主机群，以完成部署组件动作。其中，第二主机群为各主机群中除去第一主机群之外的任一主机群。调整系统依次按照优先级降序从组件频繁项集中筛选第一组件，并与当前已部署在主机群中的各个第一组件逐一执行预设条件的判断动作和部署组件动作，直至组件频繁项集中的所有第一组件均部署完成。比如，组件频繁项集为：{[组件1，出现次数：1次，亲和：1次，较亲和：0次，违亲和：0次，反亲和：0次]，[组件2，出现次数：3次，亲和：1次，较亲和：1次，违亲和：0次，反亲和：1次]，[组件3，出现次数：1次，亲和：0次，较亲和：0次，违亲和：0次，反亲和：1次]，[组件4，出现次数：

1次, 亲和: 0次, 较亲和: 1次, 违亲和: 0次, 反亲和: 0次}}。组件2出现了3次还有1次亲和, 优先级第一, 首先部署组件2到主机群1上。组件1的出现次数第2, 亲和次数相比较多, 优先级第二, 所以接下来部署组件1。由于组件1和组件2亲和, 则将组件1也部署到主机群1中。组件4和组件3, 虽然出现次数都是1次, 亲和次数都为0, 但是组件4是较亲和, 组件3是反亲和, 所以优先部署组件4。组件4和主机群1上已经部署的组件1没有关联, 和组件2是较亲和, 所以可以部署在主机群1上。而如果出现违亲和和反亲和的关联, 则不能部署在一起, 要在新的主机群上独立部署。最后部署组件3, 组件3和主机群1中部署的组件2是反亲和的, 不能部署到主机群1中, 要部署到新的主机群2中。

进一步的, 所述将所述优先级第二的第一组件部署在所述第一主机群的步骤, 包括:

S4011: 判断所述优先级第二的第一组件与所述优先级第一的第一组件之间的亲和关系是否为亲和;

S4012: 若是亲和, 则将所述优先级第二的第一组件与所述优先级第一的第一组件部署在同一主机; 若是较亲和, 则将所述优先级第二的第一组件部署在所述第一主机群中的任一主机。

本实施例中, 主机群包括多个主机。调整系统在判定优先级第二的第一组件需要与优先级第一的第一组件部署在同一主机群时, 还需要判断两个第一组件之间的关系是否为亲和, 即为最高亲和度。如果是, 则将优先级第二的第一组件与优先级第一的第一组件部署在同一主机上, 以便两者之间的数据通信传输和执行效率更高。若不是两个第一组件间的关系为较亲和, 则将优先级第二的第一组件随机部署在第一主机群中的任一主机上。

进一步的, 所述根据预先设定的优先级, 从所述组件频繁项集中筛选出优先级第二的第一组件, 并判断所述优先级第二的第一组件和所述优先级第一的第一组件之间的亲和关系是否符合预设条件, 以完成预设条件的判断动作的步骤中, 包括:

S40201: 判断当前已部署有第一组件的主机群数量是否不小于2;

S40202: 若不小于2, 则逐一判断当前次进行部署的第一组件与各所述当前已部署有第一组件的主机群中的各Docker组件之间的亲和关系是否均符合所述预设条件, 以完成预设条件的判断动作;

所述若符合, 则将所述优先级第二的第一组件部署在所述第一主机群; 若

不符合，则将优先级第二的第一组件部署在第二主机群，以完成部署组件动作的步骤中，包括：

S40301：将所述当前次进行部署的第一组件随机部署在各所述当前已部署有第一组件的主机群中的任意一个。

本实施例中，在当前已部署有第一组件的主机群的数量不小于2时，如果当前次进行部署的第一组件与各个当前已部署有第一组件的主机群中的各组件之间的亲和关系均符合预设条件，则将当前次进行部署的第一组件随机部署在各个当前已部署有第一组件的主机群中的任意一个。比如，当前已部署有第一组件的主机群为：主机群A，已部署的第一组件为a；主机群B，已部署的第一组件为b；主机群C，已部署的第一组件为c。当前次进行部署的第一组件为d，与第一组件a、b、c的亲和关系均为亲和或较亲和，则调整系统将第一组件d随机部署在主机群A、主机群B和主机群C中任意一个。

进一步的，所述根据预设频率，获取所述预设频率对应的时间段内的各所述Docker组件间的运行信息以及各所述Docker组件分别对应的运行占用信息的步骤之前，包括：

S5：实时监控API Gateway中调用请求的调用数据，其中，所述调用请求为各所述Docker组件间相互调用时发起的请求，所述调用数据包括请求端组件、目的端组件和调用请求的数量；

S6：根据所述请求端组件和目的端组件生成所述组件与组件间相互对应的调用关系，并根据所述调用请求的数量，更新所述请求端组件和所述目的端组件对应的调用次数；

S7：根据所述组件与组件间相互对应的调用关系和更新后的所述调用次数，生成所述调用关系信息。

本实施例中，docker容器中的组件的服务在相互调用时，需要将调用请求先发送到API Gateway组件，然后由API Gateway组件负责将调用请求转发到目的端组件的服务。调整系统通过各主机中安装的监控服务Agent，监控并从API Gateway组件采集Docker容器服务之间的交互调用数据，其中，调用数据包括请求端组件、目的端组件和调用请求的数量。调整系统根据请求端组件和目的端组件，生成组件与组件间相互对应的调用关系，各组具有调用关系的两个组件记录有对应的调用次数。调整系统根据调用请求的数量，在该组具有调用关系的两个组件原来的调用次数的基础上，再次加上本次调用请求的数量，从而完

成对该组具有调用关系的两个组件对应的调用次数的更新。调整系统根据组件与组件间相互对应的调用关系，以及更新后的调用次数，得到该组具有调用关系的两个组件的调用关系信息。

本实施例提供一种基于亲和性调整 Docker 组件部署的方法，通过根据预设频率监控系统中各 Docker 组件的运行信息和运行占用信息，从而自动解析得到各组件间的亲和关系。再根据预设的优先级和组件间的亲和关系，动态调整各组件在各不同地区的主机群上的部署，从而使得具有较高亲和性的组件可以部署在相同的地域的主机群上，有利于数据通信传输和执行效率。

参照图2，本申请一实施例中还提供了一种基于亲和性调整 Docker 组件部署的装置，所述 Docker 组件部署于主机群中，所述主机群为设置在同一地区的所有所述主机形成的集群，所述装置包括：

获取模块1，用于根据预设频率，获取所述预设频率对应的时间段内的各所述 Docker 组件间的运行信息以及各所述 Docker 组件分别对应的运行占用信息，其中，所述运行信息为各所述 Docker 组件间的调用关系信息和资源共享信息，所述调用关系信息为组件与组件间相互对应的调用关系和调用次数，所述资源共享信息为组件与组件间在运行时共同调用的资源的对应信息；

解析模块2，用于解析各所述 Docker 组件间的运行信息和各所述 Docker 组件的运行占用信息，得到各所述 Docker 组件间的亲和关系；

第一生成模块3，用于调用 Apriori 算法解析各所述 Docker 组件间的亲和关系，生成组件频繁项集，其中，所述组件频繁项集为包含各所述 Docker 组件间的亲和关系的集合；

调整模块4，用于根据各所述 Docker 组件间的亲和关系和所述组件频繁项集，调整各所述 Docker 组件在各主机群上的部署。

本实施例中，装置各模块、单元的实施例与上述对应的方法步骤一致，在此不作详述。

本实施例提供一种基于亲和性调整 Docker 组件部署的装置，通过根据预设频率监控系统中各 Docker 组件的运行信息和运行占用信息，从而自动解析得到各组件间的亲和关系。再根据预设的优先级和组件间的亲和关系，动态调整各组件在各不同地区的主机群上的部署，从而使得具有较高亲和性的组件可以部署在相同的地域的主机群上，有利于数据通信传输和执行效率。

参照图 3，本申请实施例中还提供一种计算机设备，该计算机设备可以是服务器，其内部结构可以如图 3 所示。该计算机设备包括通过系统总线连接的处理器、存储器、网络接口和数据库。其中，该计算机设计的处理器用于提供计算和控制能力。该计算机设备的存储器包括非易失性存储介质、内存储器。该非易失性存储介质存储有操作系统、计算机可读指令和数据库。该内存储器为非易失性存储介质中的操作系统和计算机可读指令的运行提供环境。该计算机设备的数据库用于存储亲和关系库等数据。该计算机设备的网络接口用于与外部的终端通过网络连接通信。该计算机可读指令被处理器执行时执行如上述各方法的实施例的流程。本领域技术人员可以理解，图 3 中示出的结构，仅仅是与本申请方案相关的部分结构的框图，并不构成对本申请方案所应用于其上的计算机设备的限定。

本申请一实施例还提供一种计算机可读存储介质，该可读存储介质可以是易失性可读存储介质，也可以是非易失性可读存储介质，其上存储有计算机可读指令，计算机可读指令被处理器执行时，执行如上述各方法的实施例的流程。以上所述仅为本申请的优选实施例，并非因此限制本申请的专利范围，凡是利用本申请说明书及附图内容所作的等效结构或等效流程变换，或直接或间接运用在其他相关的技术领域，均同理包括在本申请的专利保护范围内。

权利要求书

1、一种基于亲和性调整Docker组件部署的方法，所述Docker组件部署于主机群中，所述主机群为设置在同一地区的所有所述主机形成的集群，其特征在于，所述方法包括：

5 根据预设频率，获取所述预设频率对应的时间段内的各所述Docker组件间的运行信息以及各所述Docker组件分别对应的运行占用信息，其中，所述运行信息为各所述Docker组件间的调用关系信息和资源共享信息，所述调用关系信息为组件与组件间相互对应的调用关系和调用次数，所述资源共享信息为组件与组件间在运行时共同调用的资源的对应信息；

10 解析各所述Docker组件间的运行信息和各所述Docker组件的运行占用信息，得到各所述Docker组件间的亲和关系；

调用Apriori算法解析各所述Docker组件间的亲和关系，生成组件频繁项集，其中，所述组件频繁项集为包含各所述Docker组件间的亲和关系的集合；

15 根据各所述Docker组件间的亲和关系和所述组件频繁项集，调整各所述Docker组件在各主机群上的部署。

2、根据权利要求1所述的基于亲和性调整Docker组件部署的方法，其特征在于，所述运行占用信息包括CPU占用率和内存占用量，所述解析各所述Docker组件间的运行信息和各所述Docker组件的运行占用信息，得到各所述Docker组件间的亲和关系的步骤，包括：

20 根据各所述Docker组件间的调用次数和所述预设频率对应的时间段，计算得到各Docker组件间的调用频率；

将各所述Docker组件间的调用频率、各所述Docker组件间的资源共享信息和各所述Docker组件的运行占用信息输入预先构建的亲和关系库中，匹配得到各所述Docker组件间的亲和关系，其中，所述亲和关系库中存储有组件间的调用频率、组件间的资源共享信息、组件的运行占用信息与亲和关系映射关系表。

25 3、根据权利要求1所述的基于亲和性调整Docker组件部署的方法，其特征在于，所述Docker组件间的亲和关系包括两个Docker组件对应的亲和信息，所述调用Apriori算法解析各所述Docker组件间的亲和关系，生成组件频繁项集的步骤，包括：

30 根据各所述Docker组件间的亲和关系，从各所述Docker组件中筛选得到与其他Docker组件具有确定的亲和关系的各个第一组件；

根据各所述第一组件以及各所述Docker组件间的亲和关系,生成各所述第一组件间的亲和关系对应的亲和性集;

根据各所述亲和性集,统计得到各所述第一组件的出现次数以及各所述第一组件对应的各亲和信息对应的出现次数;

5 根据各所述第一组件、各所述第一组件的出现次数以及各所述第一组件的各亲和信息对应的出现次数,生成所述组件频繁项集。

4、根据权利要求3所述的基于亲和性调整Docker组件部署的方法,其特征在于,所述亲和信息包括亲和、较亲和、违亲和以及反亲和,所述根据各所述Docker组件间的亲和关系和所述组件频繁项集,调整各所述Docker组件在各主机群上的部署的步骤,包括:

根据预先设定的优先级,从所述组件频繁项集中筛选出优先级第一的第一组件,并将所述优先级第一的第一组件部署在第一主机群,所述第一主机群为各所述主机群中的任一主机群;

15 根据预先设定的优先级,从所述组件频繁项集中筛选出优先级第二的第一组件,并判断所述优先级第二的第一组件和所述优先级第一的第一组件之间的亲和关系是否符合预设条件,以完成预设条件的判断动作,其中,所述预设条件为组件之间的亲和关系为亲和或较亲和;

20 若符合,则将所述优先级第二的第一组件部署在所述第一主机群;若不符合,则将优先级第二的第一组件部署在第二主机群,以完成部署组件动作,其中,所述第二主机群为各所述主机群中除所述第一主机群之外的任一主机群;

依次按照优先级降序从所述组件频繁项集中筛选第一组件,并与当前已部署在主机群的第一组件逐一执行所述预设条件的判断动作和所述部署组件动作,直至所述组件频繁项集中的所有第一组件均部署完成。

25 5、根据权利要求4所述的基于亲和性调整Docker组件部署的方法,其特征在于,所述将所述优先级第二的第一组件部署在所述第一主机群的步骤,包括:

判断所述优先级第二的第一组件与所述优先级第一的第一组件之间的亲和关系是否为亲和;

30 若是亲和,则将所述优先级第二的第一组件与所述优先级第一的第一组件部署在同一主机;若是较亲和,则将所述优先级第二的第一组件部署在所述第一主机群中的任一主机。

6、根据权利要求4所述的基于亲和性调整Docker组件部署的方法,其特征

在于,所述根据预先设定的优先级,从所述组件频繁项集中筛选出优先级第二的第一组件,并判断所述优先级第二的第一组件和所述优先级第一的第一组件之间的亲和关系是否符合预设条件,以完成预设条件的判断动作的步骤中,包括:

5 判断当前已部署有第一组件的主机群数量是否不小于2;

若不小于2,则逐一判断当前次进行部署的第一组件与各所述当前已部署有第一组件的主机群中的各Docker组件之间的亲和关系是否均符合所述预设条件,以完成预设条件的判断动作;

10 所述若符合,则将所述优先级第二的第一组件部署在所述第一主机群;若不符合,则将优先级第二的第一组件部署在第二主机群,以完成部署组件动作的步骤中,包括

将所述当前次进行部署的第一组件随机部署在各所述当前已部署有第一组件的主机群中的任意一个。

15 7、根据权利要求1所述的基于亲和性调整Docker组件部署的方法,其特征在于,所述根据预设频率,获取所述预设频率对应的时间段内的各所述Docker组件间的运行信息以及各所述Docker组件分别对应的运行占用信息的步骤之前,包括:

20 实时监控API Gateway中调用请求的调用数据,其中,所述调用请求为各所述Docker组件间相互调用时发起的请求,所述调用数据包括请求端组件、目的端组件和调用请求的数量;

根据所述请求端组件和目的端组件生成所述组件与组件间相互对应的调用关系,并根据所述调用请求的数量,更新所述请求端组件和所述目的端组件对应的调用次数;

25 根据所述组件与组件间相互对应的调用关系和更新后的所述调用次数,生成所述调用关系信息。

8、一种基于亲和性调整Docker组件部署的装置,所述Docker组件部署于主机群中,所述主机群为设置在同一地区的所有所述主机形成的集群,其特征在于,所述装置包括:

30 获取模块,用于根据预设频率,获取所述预设频率对应的时间段内的各所述Docker组件间的运行信息以及各所述Docker组件分别对应的运行占用信息,其中,所述运行信息为各所述Docker组件间的调用关系信息和资源共享信息,

所述调用关系信息为组件与组件间相互对应的调用关系和调用次数,所述资源共享信息为组件与组件间在运行时共同调用的资源的对应信息;

解析模块,用于解析各所述Docker组件间的运行信息和各所述Docker组件的运行占用信息,得到各所述Docker组件间的亲和关系;

- 5 第一生成模块,用于调用Apriori算法解析各所述Docker组件间的亲和关系,生成组件频繁项集,其中,所述组件频繁项集为包含各所述Docker组件间的亲和关系的集合;

调整模块,用于根据各所述 Docker 组件间的亲和关系和所述组件频繁项集,调整各所述 Docker 组件在各主机群上的部署。

- 10 9、根据权利要求 8 所述的基于亲和性调整 Docker 组件部署的装置,其特征在于,所述运行占用信息包括 CPU 占用率和内存占用量,所述解析模块,包括:

计算单元,用于根据各所述Docker组件间的调用次数和所述预设频率对应的时间段,计算得到各Docker组件间的调用频率;

- 15 匹配单元,用于将各所述 Docker 组件间的调用频率、各所述 Docker 组件间的资源共享信息和各所述 Docker 组件的运行占用信息输入预先构建的亲和关系库中,匹配得到各所述 Docker 组件间的亲和关系,其中,所述亲和关系库中存储有组件间的调用频率、组件间的资源共享信息、组件的运行占用信息与亲和关系映射关系表。

- 20 10、根据权利要求8所述的基于亲和性调整Docker组件部署的装置,其特征在于,所述Docker组件间的亲和关系包括两个Docker组件对应的亲和信息,所述第一生成模块,包括:

筛选单元,用于根据各所述Docker组件间的亲和关系,从各所述Docker组件中筛选得到与其他Docker组件具有确定的亲和关系的各个第一组件;

- 25 第一生成单元,用于根据各所述第一组件以及各所述Docker组件间的亲和关系,生成各所述第一组件间的亲和关系对应的亲和性集;

统计单元,用于根据各所述亲和性集,统计得到各所述第一组件的出现次数以及各所述第一组件对应的各亲和信息对应的出现次数;

- 30 第二生成单元,用于根据各所述第一组件、各所述第一组件的出现次数以及各所述第一组件的各亲和信息对应的出现次数,生成所述组件频繁项集。

11、根据权利要求10所述的基于亲和性调整Docker组件部署的装置,其特

征在于,所述亲和信息包括亲和、较亲和、违亲和以及反亲和,所述调整模块,包括:

第一部署单元,用于根据预先设定的优先级,从所述组件频繁项集中筛选出优先级第一的第一组件,并将所述优先级第一的第一组件部署在第一主机群,所述第一主机群为各所述主机群中的任一主机群;

判断单元,用于根据预先设定的优先级,从所述组件频繁项集中筛选出优先级第二的第一组件,并判断所述优先级第二的第一组件和所述优先级第一的第一组件之间的亲和关系是否符合预设条件,以完成预设条件的判断动作,其中,所述预设条件为组件之间的亲和关系为亲和或较亲和;

第二部署单元,用于将所述优先级第二的第一组件部署在所述第一主机群;或者,用于将优先级第二的第一组件部署在第二主机群,以完成部署组件动作,其中,所述第二主机群为各所述主机群中除所述第一主机群之外的任一主机群;

循环单元,用于依次按照优先级降序从所述组件频繁项集中筛选第一组件,并与当前已部署在主机群的各第一组件逐一执行所述预设条件的判断动作和所述部署组件动作,直至所述组件频繁项集中的所有第一组件均部署完成。

12、根据权利要求11所述的基于亲和性调整Docker组件部署的装置,其特征在于,所述第二部署单元,包括:

第一判断子单元,用于判断所述优先级第二的第一组件与所述优先级第一的第一组件之间的亲和关系是否为亲和;

第一部署子单元,用于将所述优先级第二的第一组件与所述优先级第一的第一组件部署在同一主机;

第二部署子单元,用于将所述优先级第二的第一组件部署在所述第一主机群中的任一主机。

13、根据权利要求11所述的基于亲和性调整Docker组件部署的装置,其特征在于,所述判断单元包括:

第二判断子单元,用于判断当前已部署有第一组件的主机群数量是否不小于2;

第三判断子单元,用于逐一判断当前次进行部署的第一组件与各所述当前已部署有第一组件的主机群中的各Docker组件之间的亲和关系是否均符合所述预设条件,以完成预设条件的判断动作;

所述第二部署单元包括:

第三部署子单元,用于将所述当前次进行部署的第一组件随机部署在各所述当前已部署有第一组件的主机群中的任意一个。

14、根据权利要求8所述的基于亲和性调整Docker组件部署的装置,其特征在于,所述装置还包括:

监控模块,用于实时监控API Gateway中调用请求的调用数据,其中,所述调用请求为各所述Docker组件间相互调用时发起的请求,所述调用数据包括请求端组件、目的端组件和调用请求的数量;

第二生成模块,用于根据所述请求端组件和目的端组件生成所述组件与组件间相互对应的调用关系,并根据所述调用请求的数量,更新所述请求端组件和所述目的端组件对应的调用次数;

第三生成模块,用于根据所述组件与组件间相互对应的调用关系和更新后的所述调用次数,生成所述调用关系信息。

15、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,其特征在于,所述处理器执行时实现基于亲和性调整 Docker 组件部署的方法,所述 Docker 组件部署于主机群中,所述主机群为设置在同一地区的所有所述主机形成的集群,该方法包括:

根据预设频率,获取所述预设频率对应的时间段内的各所述Docker组件间的运行信息以及各所述Docker组件分别对应的运行占用信息,其中,所述运行信息为各所述Docker组件间的调用关系信息和资源共享信息,所述调用关系信息为组件与组件间相互对应的调用关系和调用次数,所述资源共享信息为组件与组件间在运行时共同调用的资源的对应信息;

解析各所述Docker组件间的运行信息和各所述Docker组件的运行占用信息,得到各所述Docker组件间的亲和关系;

调用Apriori算法解析各所述Docker组件间的亲和关系,生成组件频繁项集,其中,所述组件频繁项集为包含各所述Docker组件间的亲和关系的集合;

根据各所述 Docker 组件间的亲和关系和所述组件频繁项集,调整各所述 Docker 组件在各主机群上的部署。

16、根据权利要求 15 所述的计算机设备,其特征在于,所述运行占用信息包括 CPU 占用率和内存占用量,所述解析各所述 Docker 组件间的运行信息

和各所述 Docker 组件的运行占用信息；所述处理器执行所述解析各所述 Docker 组件间的运行信息和各所述 Docker 组件的运行占用信息，得到各所述 Docker 组件间的亲和关系的步骤，包括：

5 根据各所述 Docker 组件间的调用次数和所述预设频率对应的时间段，计算得到各 Docker 组件间的调用频率；

将各所述 Docker 组件间的调用频率、各所述 Docker 组件间的资源共享信息和各所述 Docker 组件的运行占用信息输入预先构建的亲和关系库中，匹配得到各所述 Docker 组件间的亲和关系，其中，所述亲和关系库中存储有组件间的调用频率、组件间的资源共享信息、组件的运行占用信息与亲和关系映射关系表。

17、根据权利要求 15 所述的计算机设备，其特征在于，所述 Docker 组件间的亲和关系包括两个 Docker 组件对应的亲和信息；所述处理器执行所述调用 Apriori 算法解析各所述 Docker 组件间的亲和关系，生成组件频繁项集的步骤，包括：

15 根据各所述 Docker 组件间的亲和关系，从各所述 Docker 组件中筛选得到与其他 Docker 组件具有确定的亲和关系的各个第一组件；

根据各所述第一组件以及各所述 Docker 组件间的亲和关系，生成各所述第一组件间的亲和关系对应的亲和性集；

20 根据各所述亲和性集，统计得到各所述第一组件的出现次数以及各所述第一组件对应的各亲和信息对应的出现次数；

根据各所述第一组件、各所述第一组件的出现次数以及各所述第一组件的各亲和信息对应的出现次数，生成所述组件频繁项集。

25 18、一种计算机可读存储介质，其上存储有计算机可读指令，其特征在于，所述计算机可读指令被处理器执行时实现基于亲和性调整 Docker 组件部署的方法，所述 Docker 组件部署于主机群中，所述主机群为设置在同一地区的所有所述主机形成的集群，其特征在于，所述方法包括：

30 根据预设频率，获取所述预设频率对应的时间段内的各所述 Docker 组件间的运行信息以及各所述 Docker 组件分别对应的运行占用信息，其中，所述运行信息为各所述 Docker 组件间的调用关系信息和资源共享信息，所述调用关系信息为组件与组件间相互对应的调用关系和调用次数，所述资源共享信息为组件与组件间在运行时共同调用的资源的对应信息；

解析各所述Docker组件间的运行信息和各所述Docker组件的运行占用信息，得到各所述Docker组件间的亲和关系；

调用Apriori算法解析各所述Docker组件间的亲和关系，生成组件频繁项集，其中，所述组件频繁项集为包含各所述Docker组件间的亲和关系的集合；

5 根据各所述Docker组件间的亲和关系和所述组件频繁项集，调整各所述Docker组件在各主机群上的部署。

19、根据权利要求18所述的计算机可读存储介质，其特征在于，所述运行占用信息包括CPU占用率和内存占用量；所述处理器执行所述解析各所述Docker组件间的运行信息和各所述Docker组件的运行占用信息，得到各所述
10 Docker组件间的亲和关系的步骤，包括：

根据各所述Docker组件间的调用次数和所述预设频率对应的时间段，计算得到各Docker组件间的调用频率；

将各所述Docker组件间的调用频率、各所述Docker组件间的资源共享信息和各所述Docker组件的运行占用信息输入预先构建的亲和关系库中，匹配得到
15 各所述Docker组件间的亲和关系，其中，所述亲和关系库中存储有组件间的调用频率、组件间的资源共享信息、组件的运行占用信息与亲和关系映射关系表。

20、根据权利要求18所述的计算机可读存储介质，其特征在于，所述Docker组件间的亲和关系包括两个Docker组件对应的亲和信息，所述处理器执行所述调用Apriori算法解析各所述Docker组件间的亲和关系，生成组件频
20 繁项集的步骤，包括：

根据各所述Docker组件间的亲和关系，从各所述Docker组件中筛选得到与其他Docker组件具有确定的亲和关系的各个第一组件；

根据各所述第一组件以及各所述Docker组件间的亲和关系，生成各所述第一组件间的亲和关系对应的亲和性集；

25 根据各所述亲和性集，统计得到各所述第一组件的出现次数以及各所述第一组件对应的各亲和信息对应的出现次数；

根据各所述第一组件、各所述第一组件的出现次数以及各所述第一组件的各亲和信息对应的出现次数，生成所述组件频繁项集。

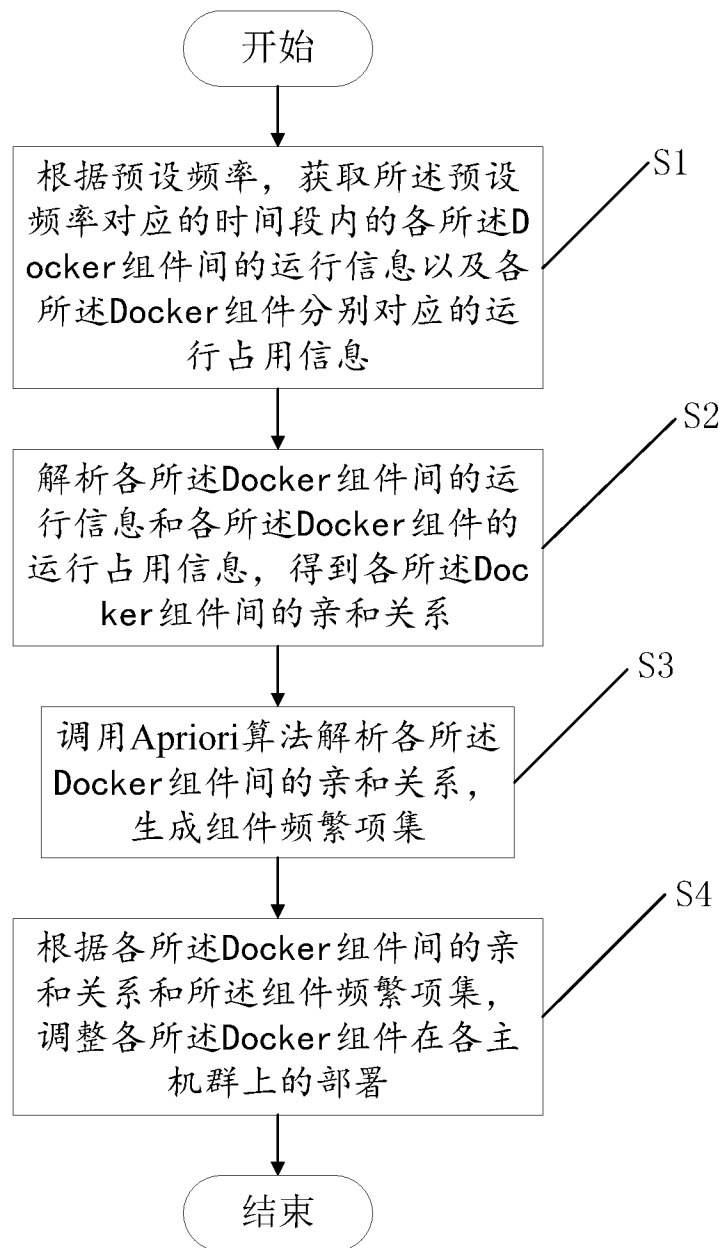


图 1

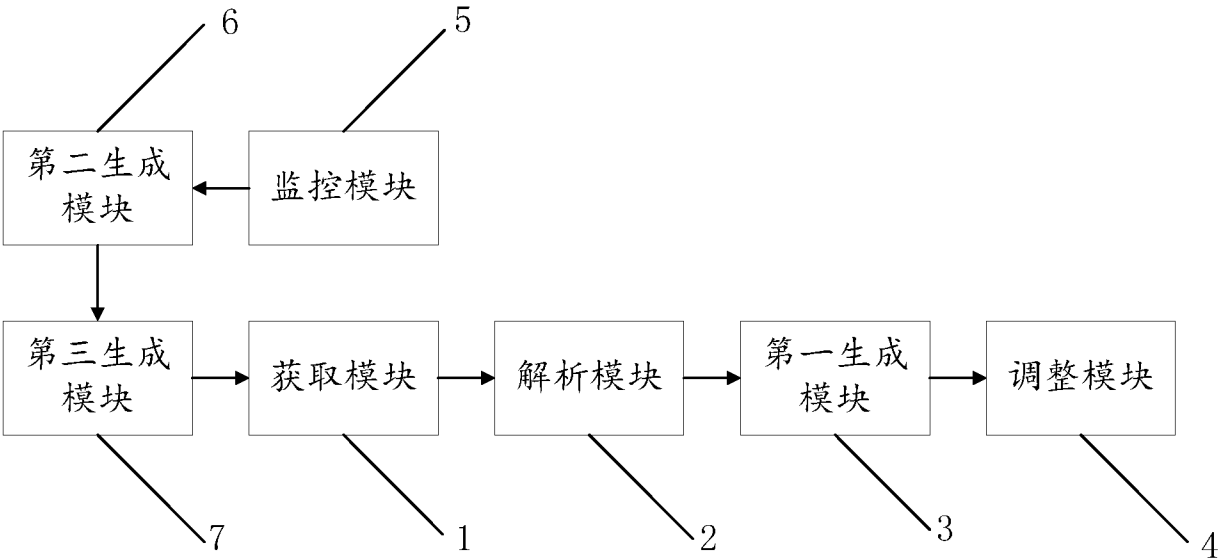


图 2

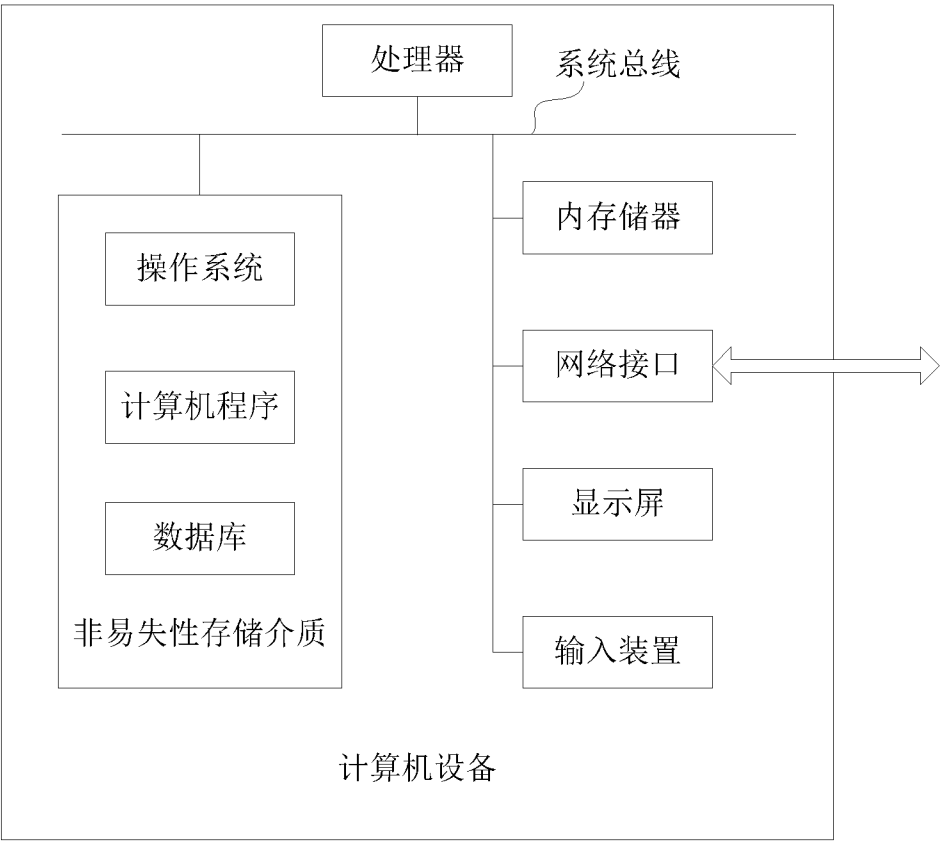


图 3

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2019/116704

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/455(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)

CNABS; CNTXT; VEN; WOTXT; EPTXT; USTXT; CNKI: 组件, 部署, 调整, 调用, 关系, 次数, 共同, 频率, 亲和, 频繁项集, component, deploy, adjust, call, relationship, times, common, frequency, affinity, frequent item set

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN 109933406 A (PING AN TECHNOLOGY (SHENZHEN) CO., LTD.) 25 June 2019 (2019-06-25) entire document	1-20
A	CN 108234437 A (AISINO CORPORATION INC.) 29 June 2018 (2018-06-29) description, paragraphs 11-54	1-20
A	CN 109271171 A (DBAPPSECURITY LTD.) 25 January 2019 (2019-01-25) description, paragraphs 29-66	1-20
A	US 2017308822 A1 (FAIR ISAAC CORPORATION) 26 October 2017 (2017-10-26) entire document	1-20



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

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

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

“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

“O” document referring to an oral disclosure, use, exhibition or other means

“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&” document member of the same patent family

Date of the actual completion of the international search

03 February 2020

Date of mailing of the international search report

12 February 2020

Name and mailing address of the ISA/CN

China National Intellectual Property Administration (ISA/
CN)
No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing
100088
China

Facsimile No. (86-10)62019451

Authorized officer

Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2019/116704

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	109933406	A	25 June 2019	None			
CN	108234437	A	29 June 2018	None			
CN	109271171	A	25 January 2019	None			
US	2017308822	A1	26 October 2017	US	10380508	B2	13 August 2019

国际检索报告

国际申请号

PCT/CN2019/116704

A. 主题的分类

G06F 9/455 (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))

CNABS; CNTXT; VEN; WOTXT; EPTXT; USTXT; CNKI: 组件, 部署, 调整, 调用, 关系, 次数, 共同, 频率, 亲和, 频繁项集, component, deploy, adjust, call, relationship, times, common, frequency, affinity, frequent item set

C. 相关文件

类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求
PX	CN 109933406 A (平安科技深圳有限公司) 2019年 6月 25日 (2019 - 06 - 25) 全文	1-20
A	CN 108234437 A (航天信息股份有限公司) 2018年 6月 29日 (2018 - 06 - 29) 说明书第11-54段	1-20
A	CN 109271171 A (杭州安恒信息技术股份有限公司) 2019年 1月 25日 (2019 - 01 - 25) 说明书第29-66段	1-20
A	US 2017308822 A1 (FAIR ISAAC CORPORATION) 2017年 10月 26日 (2017 - 10 - 26) 全文	1-20

☐ 其余文件在C栏的续页中列出。☒ 见同族专利附件。

* 引用文件的具体类型:

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件 (如具体说明的)

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期

2020年 2月 3日

国际检索报告邮寄日期

2020年 2月 12日

ISA/CN的名称和邮寄地址

中国国家知识产权局 (ISA/CN)

中国北京市海淀区蓟门桥西土城路6号 100088

传真号 (86-10)62019451

授权官员

郑舒玲

电话号码 86-(20)-28950720

国际检索报告
关于同族专利的信息

国际申请号

PCT/CN2019/116704

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	109933406	A	2019年 6月 25日	无			
CN	108234437	A	2018年 6月 29日	无			
CN	109271171	A	2019年 1月 25日	无			
US	2017308822	A1	2017年 10月 26日	US	10380508	B2	2019年 8月 13日