



(12)发明专利申请

(10)申请公布号 CN 111538616 A

(43)申请公布日 2020.08.14

(21)申请号 202010365756.1

(22)申请日 2020.04.30

(71)申请人 深圳前海微众银行股份有限公司  
地址 518000 广东省深圳市前海深港合作区前湾一路1号A栋201室(入驻深圳市前海商务秘书有限公司)

(72)发明人 卢道和 杨军 陈广胜 陈翼

(74)专利代理机构 深圳市世纪恒程知识产权代理事务所 44287

代理人 薛福玲

(51)Int.Cl.  
G06F 11/07(2006.01)

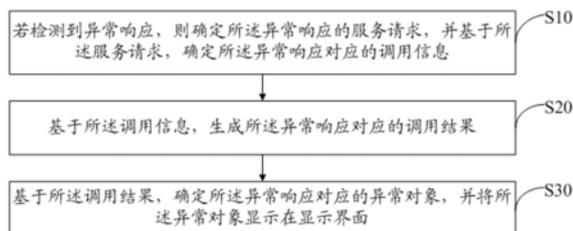
权利要求书2页 说明书13页 附图1页

(54)发明名称

异常定位方法、装置、系统与计算机可读存储介质

(57)摘要

本发明公开了一种异常定位方法,包括:若检测到异常响应,则确定所述异常响应的服务请求,并基于所述服务请求,确定所述异常响应对应的调用信息;基于所述调用信息,生成所述异常响应对应的调用结果;基于所述调用结果,确定所述异常响应对应的异常对象,并将所述异常对象显示在显示界面。本发明还公开了一种异常定位装置、系统与计算机可读存储介质。本发明在检测到异常响应时,不需要登录多个系统一一查找异常的根源,而是通过将异常响应对应的调用信息,生成调用结果,再从调用结果定位出异常对象,使得异常对象可直接显示在显示界面上,在提高异常定位的便捷性的同时,实现异常的快速定位。



1. 一种异常定位方法,其特征在于,所述异常定位方法包括如下步骤:

若检测到异常响应,则确定所述异常响应的服务请求,并基于所述服务请求,确定所述异常响应对应的调用信息;

基于所述调用信息,生成所述异常响应对应的调用结果;

基于所述调用结果,确定所述异常响应对应的异常对象,并将所述异常对象显示在显示界面。

2. 如权利要求1所述的异常定位方法,其特征在于,所述调用信息包括调用链ID和跨度ID,所述基于所述调用信息,生成所述异常响应对应的调用结果的步骤包括:

基于所述调用链ID,确定所述异常响应对应的服务节点;

基于所述跨度ID,确定所述服务节点的调用顺序以及所述服务节点的层级关系;

基于所述服务节点、所述调用顺序和所述层级关系,生成所述异常响应对应的调用结果。

3. 如权利要求2所述的异常定位方法,其特征在于,所述调用信息包括响应ID,所述基于所述跨度ID,确定所述服务节点的调用顺序以及所述服务节点的层级关系的步骤包括:

基于所述跨度ID,确定所述服务节点的调用顺序,

基于所述响应ID,确定所述服务节点的响应方;

基于所述调用顺序和所述响应方,确定所述服务节点的层级关系。

4. 如权利要求2所述的异常定位方法,其特征在于,所述基于所述服务节点、所述调用顺序和所述层级关系,生成所述异常响应对应的调用结果的步骤包括:

基于所述层级关系,确定所述服务节点的调用深度;

基于所述调用深度、所述服务节点和所述调用顺序,生成所述异常响应对应的调用结果。

5. 如权利要求2所述的异常定位方法,其特征在于,所述基于所述服务节点、所述调用顺序和所述层级关系,生成所述异常响应对应的调用结果的步骤包括:

基于所述调用顺序,确定所述服务节点中的根服务节点;

基于所述层级关系和所述根服务节点,将所述服务节点生成调用树,并将所述调用树确定为所述异常响应对应的调用结果。

6. 如权利要求2所述的异常定位方法,其特征在于,所述若检测到异常响应,则确定所述异常响应的服务请求,并基于所述服务请求,确定所述异常响应对应的调用信息的步骤之前,所述异常定位方法还包括:

若检测到所述服务请求,则基于预设代理组件,生成所述服务请求的所述调用链ID,以及所述服务请求的第一跨度ID,并将所述调用链ID封装进所述服务请求;

基于所述预设代理组件,确定所述服务请求调用的所述服务节点,并基于所述服务节点,确定所述服务请求的第二跨度ID,所述跨度ID包括所述第一跨度ID和所述第二跨度ID;

将所述跨度ID和所述调用链ID关联保存。

7. 如权利要求1-6任一项所述的异常定位方法,其特征在于,所述基于所述调用结果,确定所述异常响应对应的异常对象,并将所述异常对象显示在显示界面的步骤包括:

识别所述调用结果中的异常节点,并基于所述异常节点的日志信息,确定所述异常节点的异常对象;

确定所述异常节点在所述调用结果的异常位置；

基于所述异常位置，将所述异常对象显示在显示界面。

8. 一种异常定位装置，其特征在于，所述异常定位装置包括：

第一确定模块，用于若检测到异常响应，则确定所述异常响应的服务请求，并基于所述服务请求，确定所述异常响应对应的调用信息；

生成模块，用于基于所述调用信息，生成所述异常响应对应的调用结果；

第二确定模块，用于基于所述调用结果，确定所述异常响应对应的异常对象，并将所述异常对象显示在显示界面。

9. 一种异常定位系统，其特征在于，所述异常定位系统包括：存储器、处理器及存储在所述存储器上并可在所述处理器上运行的异常定位程序，所述异常定位程序被所述处理器执行时实现如权利要求1至7中任一项所述的异常定位方法的步骤。

10. 一种计算机可读存储介质，其特征在于，所述计算机可读存储介质上存储有异常定位程序，所述异常定位程序被处理器执行时实现如权利要求1至7中任一项所述的异常定位方法的步骤。

## 异常定位方法、装置、系统与计算机可读存储介质

### 技术领域

[0001] 本发明涉及金融科技 (Fintech) 技术领域, 尤其涉及异常定位方法、装置、系统与计算机可读存储介质。

### 背景技术

[0002] 近年来, 随着金融科技 (Fintech), 尤其是互联网金融的不断发展, 异常定位技术被引入银行等金融机构的日常服务中。在银行等金融机构的日常运维服务过程中, 为满足日趋复杂的业务需求, 往往需要调用多个系统才能响应一次请求, 若在此过程中, 出现异常响应, 则需要登录每一个系统查看相关日志信息, 才能知道具体是哪一系统出现故障, 显然, 此种查找方式效率极为低下。

[0003] 在现有技术中, 为提高异常的查找效率, 一般采用链路跟踪的方式, 也即, 收集当前请求所调用的系统的链路信息, 如调用的系统, 调用的时间顺序等, 从而确定当前请求所调用的系统链路, 再从确定的系统链路查找异常根源。然而, 通过对收集到的链路信息进行查询, 虽然缩小了查找范围, 不再需要查找所有的系统, 但得到的结果依然无法准确地定位出请求异常的错误根源, 可见, 多系统调用的异常定位效率还有待提高。

### 发明内容

[0004] 本发明的主要目的在于提出一种异常定位方法、装置、系统与计算机可读存储介质, 旨在实现异常的快速定位。

[0005] 为实现上述目的, 本发明提供一种异常定位方法, 所述异常定位方法包括如下步骤:

[0006] 若检测到异常响应, 则确定所述异常响应的服务请求, 并基于所述服务请求, 确定所述异常响应对应的调用信息;

[0007] 基于所述调用信息, 生成所述异常响应对应的调用结果;

[0008] 基于所述调用结果, 确定所述异常响应对应的异常对象, 并将所述异常对象显示在显示界面。

[0009] 优选地, 所述调用信息包括调用链ID和跨度ID, 所述基于所述调用信息, 生成所述异常响应对应的调用结果的步骤包括:

[0010] 基于所述调用链ID, 确定所述异常响应对应的服务节点;

[0011] 基于所述跨度ID, 确定所述服务节点的调用顺序以及所述服务节点的层级关系;

[0012] 基于所述服务节点、所述调用顺序和所述层级关系, 生成所述异常响应对应的调用结果。

[0013] 优选地, 所述调用信息包括响应ID, 所述基于所述跨度ID, 确定所述服务节点的调用顺序以及所述服务节点的层级关系的步骤包括:

[0014] 基于所述跨度ID, 确定所述服务节点的调用顺序,

[0015] 基于所述响应ID, 确定所述服务节点的响应方;

- [0016] 基于所述调用顺序和所述响应方,确定所述服务节点的层级关系。
- [0017] 优选地,所述基于所述服务节点、所述调用顺序和所述层级关系,生成所述异常响应对应的调用结果的步骤包括:
- [0018] 基于所述层级关系,确定所述服务节点的调用深度;
- [0019] 基于所述调用深度、所述服务节点和所述调用顺序,生成所述异常响应对应的调用结果。
- [0020] 优选地,所述基于所述服务节点、所述调用顺序和所述层级关系,生成所述异常响应对应的调用结果的步骤包括:
- [0021] 基于所述调用顺序,确定所述服务节点中的根服务节点;
- [0022] 基于所述层级关系和所述根服务节点,将所述服务节点生成调用树,并将所述调用树确定为所述异常响应对应的调用结果。
- [0023] 优选地,所述若检测到异常响应,则确定所述异常响应的业务信息,并基于所述业务信息,确定所述异常响应对应的调用信息的步骤之前,所述异常定位方法还包括:
- [0024] 若检测到所述服务请求,则基于预设代理组件,生成所述服务请求的所述调用链ID,以及所述服务请求的第一跨度ID;
- [0025] 基于所述预设代理组件,确定所述服务请求调用的所述服务节点,并基于所述服务节点,生成所述服务请求的第二跨度ID,所述跨度ID包括所述第一跨度ID和所述第二跨度ID;
- [0026] 将所述跨度ID和所述调用链ID关联保存。
- [0027] 优选地,所述基于所述调用结果,确定所述异常响应对应的异常对象,并将所述异常对象显示在显示界面的步骤包括:
- [0028] 识别所述调用结果中的异常节点,并基于所述异常节点的日志信息,确定所述异常节点的异常对象;
- [0029] 确定所述异常节点在所述调用结果的异常位置;
- [0030] 基于所述异常位置,将所述异常对象显示在显示界面。
- [0031] 此外,为实现上述目的,本发明还提供一种异常定位装置,所述异常定位装置包括:
- [0032] 第一确定模块,用于若检测到异常响应,则确定所述异常响应的服务请求,并基于所述服务请求,确定所述异常响应对应的调用信息;
- [0033] 生成模块,用于基于所述调用信息,生成所述异常响应对应的调用结果;
- [0034] 第二确定模块,用于基于所述调用结果,确定所述异常响应对应的异常对象,并将所述异常对象显示在显示界面。
- [0035] 优选地,所述调用信息包括调用链ID和跨度ID,所述生成模块还用于:
- [0036] 基于所述调用链ID,确定所述异常响应对应的服务节点;
- [0037] 基于所述跨度ID,确定所述服务节点的调用顺序以及所述服务节点的层级关系;
- [0038] 基于所述服务节点、所述调用顺序和所述层级关系,生成所述异常响应对应的调用结果。
- [0039] 优选地,所述调用信息包括响应ID,所述生成模块还用于:
- [0040] 基于所述跨度ID,确定所述服务节点的调用顺序,

- [0041] 基于所述响应ID,确定所述服务节点的响应方;
- [0042] 基于所述调用顺序和所述响应方,确定所述服务节点的层级关系。
- [0043] 优选地,所述生成模块还用于:
- [0044] 基于所述层级关系,确定所述服务节点的调用深度;
- [0045] 基于所述调用深度、所述服务节点和所述调用顺序,生成所述异常响应对应的调用结果。
- [0046] 优选地,所述生成模块还用于:
- [0047] 基于所述调用顺序,确定所述服务节点中的根服务节点;
- [0048] 基于所述层级关系和所述根服务节点,将所述服务节点生成调用树,并将所述调用树确定为所述异常响应对应的调用结果。
- [0049] 优选地,所述异常定位装置还包括跟踪模块,所述跟踪模块用于:
- [0050] 若检测到所述服务请求,则基于预设代理组件,生成所述服务请求的所述调用链ID,以及所述服务请求的第一跨度ID;
- [0051] 基于所述预设代理组件,确定所述服务请求调用的所述服务节点,并基于所述服务节点,生成所述服务请求的第二跨度ID,所述跨度ID包括所述第一跨度ID和所述第二跨度ID;
- [0052] 将所述跨度ID和所述调用链ID关联保存。
- [0053] 优选地,所述第二确定模块还用于:
- [0054] 识别所述调用结果中的异常节点,并基于所述异常节点的日志信息,确定所述异常节点的异常对象;
- [0055] 确定所述异常节点在所述调用结果的异常位置;
- [0056] 基于所述异常位置,将所述异常对象显示在显示界面。
- [0057] 此外,为实现上述目的,本发明还提供一种异常定位系统,所述异常定位系统包括:存储器、处理器及存储在所述存储器上并可在所述处理器上运行的异常定位程序,所述异常定位程序被所述处理器执行时实现如上所述的异常定位方法的步骤。
- [0058] 此外,为实现上述目的,本发明还提供一种计算机可读存储介质,所述计算机可读存储介质上存储有异常定位程序,所述异常定位程序被处理器执行时实现如上所述的异常定位方法的步骤。
- [0059] 本发明提出的异常定位方法,若检测到异常响应,则确定所述异常响应的服务请求,并基于所述服务请求,确定所述异常响应对应的调用信息;基于所述调用信息,生成所述异常响应对应的调用结果;基于所述调用结果,确定所述异常响应对应的异常对象,并将所述异常对象显示在显示界面。本发明在检测到异常响应时,不需要登录多个系统一一查找异常的根源,而是通过将异常响应对应的调用信息,生成调用结果,再从调用结果定位出异常对象,使得异常对象可直接显示在显示界面上,相较现有技术仅能确定异常在某一条链路上,而无法生成直观的调用结果,将异常直接定位出来,本发明方案在提高异常定位的便捷性的同时,实现异常的快速定位。

## 附图说明

- [0060] 图1是本发明实施例方案涉及的硬件运行环境的系统结构示意图;

[0061] 图2为本发明异常定位方法第一实施例的流程示意图。

[0062] 本发明目的的实现、功能特点及优点将结合实施例,参照附图做进一步说明。

### 具体实施方式

[0063] 应当理解,此处所描述的具体实施例仅仅用以解释本发明,并不用于限定本发明。

[0064] 如图1所示,图1是本发明实施例方案涉及的硬件运行环境的系统结构示意图。

[0065] 本发明实施例系统包括移动终端或服务器设备。

[0066] 如图1所示,该系统可以包括:处理器1001,例如CPU,网络接口1004,用户接口1003,存储器1005,通信总线1002。其中,通信总线1002用于实现这些组件之间的连接通信。用户接口1003可以包括显示屏(Display)、输入单元比如键盘(Keyboard),可选用户接口1003还可以包括标准的有线接口、无线接口。网络接口1004可选的可以包括标准的有线接口、无线接口(如WI-FI接口)。存储器1005可以是高速RAM存储器,也可以是稳定的存储器(non-volatile memory),例如磁盘存储器。存储器1005可选的还可以是独立于前述处理器1001的存储装置。

[0067] 本领域技术人员可以理解,图1中示出的系统结构并不构成对异常定位系统的限定,可以包括比图示更多或更少的部件,或者组合某些部件,或者不同的部件布置。

[0068] 如图1所示,作为一种计算机存储介质的存储器1005中可以包括操作系统、网络通信模块、用户接口模块以及异常定位程序。

[0069] 其中,操作系统是管理和控制异常定位系统与软件资源的程序,支持网络通信模块、用户接口模块、异常定位程序以及其他程序或软件的运行;网络通信模块用于管理和控制网络接口1002;用户接口模块用于管理和控制用户接口1003。

[0070] 在图1所示的异常定位系统中,所述异常定位系统通过处理器1001调用存储器1005中存储的异常定位程序,并执行下述异常定位方法各个实施例中的操作。

[0071] 基于上述硬件结构,提出本发明异常定位方法实施例。

[0072] 参照图2,图2为本发明异常定位方法第一实施例的流程示意图,所述方法包括:

[0073] 步骤S10,若检测到异常响应,则确定所述异常响应的服务请求,并基于所述服务请求,确定所述异常响应对应的调用信息;

[0074] 步骤S20,基于所述调用信息,生成所述异常响应对应的调用结果;

[0075] 步骤S30,基于所述调用结果,确定所述异常响应对应的异常对象,并将所述异常对象显示在显示界面。

[0076] 本实施例异常定位方法运用于理财机构或者银行系统等金融机构的异常定位系统中,为描述方便,异常定位系统以定位系统简称。本实施例的定位系统包括前端服务、代理组件和后端服务,用户可通过前端服务窗口,如网页等,发起服务请求,代理组件在接收到服务请求时,将其发送至对应的后端服务,其中,代理组件的代理可为正向代理或者反向代理,本实施例优选Nginx代理组件,为适应分布式调用场景,采用反向代理的方式代后端服务收发消息,之后,跟踪服务请求在后端服务的调用过程。定位系统在检测到异常响应时,可根据之前跟踪的调用过程,获取调用信息,从而生成调用结果,再根据调用结果定位异常。

[0077] 本实施例可适用多系统协调响应同一请求的应用场景,也适用多系统协调响应多

个请求的应用场景,如把整体的业务拆分成多个具有特定明确功能的服务节点的微服务架构等,微服务架构通过分散的服务节点之间的配合,响应服务请求。在微服务构架中,大部分的服务节点都是单独部署运行的,彼此通过总线交互,均为无状态的服务,处于上述微服务架构的环境中,前后端的服务请求会经过多个微服务,也即多个服务节点的处理和传递。然而,在服务请求的处理过程中,用户只能得知服务请求的最终处理结果,而无法得知服务请求的处理顺序与经过的每个服务节点的处理结果,若服务请求出现了错误或异常,则需要登录到各个服务节点的服务器查看日志文件来查找和定位问题,耗费的时间太多,导致查找和定位问题的效率太低。

[0078] 本实施例在检测到异常响应时,通过异常响应的调用信息,生成异常响应对应的调用结果,通过分析调用结果,确定异常响应对应的异常对象,不需要逐一登录各服务节点的服务器查看日志文件等信息,最后将异常对象显示在显示界面,使得用户可快速直观的定位异常。

[0079] 以下将对各个步骤进行详细说明:

[0080] 步骤S10,若检测到异常响应,则确定所述异常响应的服务请求,并基于所述服务请求,确定所述异常响应对应的调用信息。

[0081] 在本实施例中,若检测到异常响应,则先确定异常响应的服务请求,也即确定当前异常响应是哪一服务请求引起的,然后,通过服务请求,确定异常响应对应的调用信息,具体的,在确定服务请求之后,根据服务请求的业务ID,或者服务请求的服务名称等,确定异常响应对应的调用信息,其中,调用信息指的是为响应服务请求所调用的服务节点信息,也即,在检测到异常响应时,确定处理服务请求时所调用的服务节点信息,如服务请求为下单请求,若用户在发起下单请求的过程中出现异常,定位系统则需要先确定在响应下单请求过程中的服务节点信息,具体包括是哪些服务节点响应了服务请求,如下单服务节点和付款服务节点等,以及各个服务节点的调用时间等。

[0082] 步骤S20,基于所述调用信息,生成所述异常响应对应的调用结果。

[0083] 在本实施例中,在确定了调用信息,也即确定了服务节点信息之后,根据服务节点信息,生成异常响应对应的调用结果,其中,服务节点信息包括调用时间,调用名称、调用类型,以及各服务节点的日志信息等,具体按照调用时间的顺序,将各调用名称对应的服务节点串联起来,并为每一服务节点附加调用类型和日志信息,生成调用结果。

[0084] 进一步地,在一实施例中,步骤S20包括:

[0085] 步骤a1,基于所述调用链ID,确定所述异常响应对应的服务节点;

[0086] 在一实施例中,调用信息包括调用链ID(trace ID)和跨度ID(span ID),其中,trace表示一个由多个span组成的有向无环图(DAG图),而span可以是一次方法调用,一个程序块的调用,或者一次RPC/数据库访问,只要是一个具有完整时间周期的程序访问,都可被认为是一个span,因此,trace可认为是调用链路,也即服务请求传递流转的链路。

[0087] 在确定调用信息后,定位系统根据trace ID确定异常响应对应的调用链路,再根据调用链路确定异常响应对应的服务节点,也即先确定异常响应发生在哪一条调用链路上,然后,确定当前调用链路上都由哪些服务节点组成。

[0088] 步骤a2,基于所述跨度ID,确定所述服务节点的调用顺序以及所述服务节点的层级关系;

[0089] 接着,根据span ID,确定服务节点的调用顺序以及服务节点的层级关系,其中,服务节点的调用顺序可直接根据span ID确定,也即不同的调用顺序的服务节点的span ID不同,如前端服务节点的span ID为span1,后端服务节点的span ID为span2等,可确定调用顺序为前端服务节点→后端服务节点等。

[0090] 需要说明的是,span ID包括parentspan ID,也即父跨度ID,因此,可根据parentspan ID确定服务节点之间的层级关系,也即父子关系,如前端服务节点与后端服务节点存在父子关系,则前端服务节点的span ID为span1,后端服务节点的span ID为span2,parentspan ID为span1等。

[0091] 此外,还可以配置每个spanID包含所有父及前辈节点的span ID,在具体实施时,可使用圆点符号作为分隔符,如上所述例子,后端服务节点的span ID为span2.1,表示后端服务节点的span ID为span2,parentspan ID为span1,此时不再需要ParentSpan ID字段。

[0092] 步骤a3,基于所述服务节点、所述调用顺序和所述层级关系,生成所述异常响应对应的调用结果。

[0093] 在一实施例中,定位系统根据确定的服务节点、服务节点的调用顺序以及服务节点的层级关系,生成异常响应对应的调用结果,具体根据服务节点的调用顺序和层级关系,将各个服务节点串联起来,形成调用结果。

[0094] 具体的,步骤a3包括:

[0095] 步骤a31,基于所述调用顺序,确定所述服务节点中的根服务节点;

[0096] 在该步骤中,定位系统先根据各个服务节点的调用顺序,确定服务节点中的根服务节点,也即调用链路的起始服务节点。

[0097] 步骤a32,基于所述层级关系和所述根服务节点,将所述服务节点生成调用树,并将所述调用树确定为所述异常响应对应的调用结果。

[0098] 然后,以根服务节点为起点,将各个服务节点按照层级关系,依次将各服务节点与根服务节点串联,从而生成调用树,并将调用树作为异常响应对应的调用结果,也即,以树状形式,以根服务节点为地点,其他服务节点为枝点,生成调用树。

[0099] 需要说明的是,调用结果除了是调用树之外,还可以是时序图,也即,根据调用信息,生成异常响应对应的调用时序图,并将调用时序图确定为调用结果,具体的,按照调用顺序和层级关系,将各服务节点排列成调用时序图等。

[0100] 步骤S30,基于所述调用结果,确定所述异常响应对应的异常对象,并将所述异常对象显示在显示界面。

[0101] 在本实施例中,在得到调用结果后,在调用结果上确定异常对象,从而将异常对象显示在显示界面上。

[0102] 具体的,步骤S30包括:

[0103] 步骤b1,识别所述调用结果中的异常节点,并基于所述异常节点的日志信息,确定所述异常节点的异常对象;

[0104] 在本实施例中,定位系统通过识别调用结果中的异常标识,确定调用结果中的异常节点,然后读取异常节点的日志信息,确定异常节点的异常对象,也即在明确哪个节点异常后,针对性的读取该节点的日志信息,从而确定具体的异常对象是什么,不再需要进入各个服务节点的服务器去查找,实现异常的快速定位。

[0105] 其中,异常标识可通过调用超时时间表征,如某一调用结果的服务节点的调用时间超过预设时间,则确定当前服务节点异常,则对当前服务节点进行异常标记,也即当前服务节点携带有异常标识,因此,可通过识别调用结果中的异常标识,确定调用结果中的异常节点。

[0106] 在具体实施时,可先确定各服务节点对应的预设时间,其中,预设时间为各服务节点历史正常调用的平均时间范围,每一个服务节点对应的预设时间可以相同,也可以不相同,然后,确定每一个服务节点为响应当前服务请求的调用时间是否超过其对应的预设时间,若是,则确定当前服务节点异常;若否,则确定当前服务节点无异常。

[0107] 步骤b2,确定所述异常节点在所述调用结果的异常位置;

[0108] 在本实施例中,接着确定异常节点在调用结果的异常位置,也即确定在哪一节点出现异常,在具体实施时,可以起始服务节点作为起点,当前异常节点相对起始服务节点的相对位置即为异常位置。

[0109] 步骤b3,基于所述异常位置,将所述异常对象显示在显示界面。

[0110] 在本实施例中,根据异常位置,将异常对象显示在显示界面上,供用户直观查看,在具体实施时,可将异常对象在调用结果的异常位置用亮色显示,并显示异常对象,也即显示界面既显示包含异常位置的调用结果,还显示具体的异常对象,便于用户直观定位异常。

[0111] 本实施例若检测到异常响应,则确定所述异常响应的服务请求,并基于所述服务请求,确定所述异常响应对应的调用信息;基于所述调用信息,生成所述异常响应对应的调用结果;基于所述调用结果,确定所述异常响应对应的异常对象,并将所述异常对象显示在显示界面。本发明在检测到异常响应时,不需要登录多个系统一一查找异常的根源,而是通过将异常响应对应的调用信息,生成调用结果,再从调用结果定位出异常对象,使得异常对象可直接显示在显示界面上,在提高异常定位的便捷性的同时,实现异常的快速定位。

[0112] 进一步地,基于本发明异常定位方法第一实施例,提出本发明异常定位方法第二实施例。

[0113] 异常定位方法的第二实施例与异常定位方法的第一实施例的区别在于,基于所述跨度ID,确定所述服务节点的调用顺序以及所述服务节点的层级关系的步骤包括:

[0114] 步骤a21,基于所述跨度ID,确定所述服务节点的调用顺序,

[0115] 步骤a22,基于所述响应ID,确定所述服务节点的响应方;

[0116] 步骤a23,基于所述调用顺序和所述响应方,确定所述服务节点的层级关系。

[0117] 本实施例考虑到在异步调用场景下,如请求方在发送一笔服务请求时,下游服务节点存在多个响应方接收该笔服务请求,此时的请求方与响应方存在1:N的关系,若仅用span ID来区分一次调用,则会将多笔响应方接收调用标记为同一次,但实际场景却是多次响应,且有可能每次响应的的时间不一样,可见,仅用span ID来记录每一次调用,会存在调用层级关系不清晰的问题,因此,本实施例引入响应ID(response ID),使用span ID+response ID的方式来唯一标识一次完整的调用过程,从而响应方多次拉取服务请求的多次消费,都能以一个独立的标识来构建,最终体现多次消费过程,还原真实的调用情况。

[0118] 以下将对各个步骤进行详细说明:

[0119] 步骤a21,基于所述跨度ID,确定所述服务节点的调用顺序。

[0120] 在本实施例中,调用信息包括调用链ID(trace ID)、跨度ID(span ID)和响应ID

(response ID),定位系统在根据trace ID确定异常响应对应的所有服务节点之后,根据span ID,确定各个服务节点的调用顺序,也即确定服务请求的传递顺序,具体过程与上述实施例类似,在此不再赘述。

[0121] 步骤a22,基于所述响应ID,确定所述服务节点的响应方。

[0122] 在本实施例中,定位系统再根据响应ID,确定服务节点的响应方,也即消费服务请求的消费对象,在异步调用场景下,一笔服务请求很可能会被同一服务节点的多个响应方消费,或者同一响应方多次消费,如用户发起取款请求,对应的取款服务节点既要取将取款信息发送至用户终端告知用户取款情况,也要将取款请求的额度上报内部审计服务节点,以供内部审计服务节点审核,也即,在当前情形下,同一取款请求,取款服务节点两次消费,若仅用span ID,仅能表示取款服务节点参与调用,但无法知道取款服务节点消费了两次,因此,本实施例增加response ID,同一服务节点的多次消费以response ID区别,如上所述例子,取款服务节点可以span ID+response ID1,span ID+response ID2表示,也即span ID一样,response ID不一样,用以区别多次消费。

[0123] 步骤a23,基于所述调用顺序和所述响应方,确定所述服务节点的层级关系。

[0124] 在本实施例中,定位系统根据调用顺序和响应方,确定服务节点的层级关系,也即,本实施例为得到各服务节点清晰可靠的层级关系,增加response ID区分异步多次消费的情况,使得各服务节点的层级关系更加明朗清晰。

[0125] 本实施例通过增加响应ID,以跨度ID和响应ID结合的方式,标识一次完整的调用过程,从而响应方多次消费的特殊场景也能适用,还原真实的调用过程,从而得到各服务节点清晰的层级关系,以使后续生成的调用结果更加准确,同时,也在后续定位异常的过程中,可以更加精准的定位出异常对象,也即异常对象可以具体到响应方,如取款服务节点异常,进一步确定是取款服务节点将取款信息发送至用户终端出现异常,还是取款服务节点将取款请求的额度上报内部审计服务节点出现异常等。

[0126] 进一步地,基于本发明异常定位方法第一、第二实施例,提出本发明异常定位方法第三实施例。

[0127] 异常定位方法的第三实施例与异常定位方法的第一、第二实施例的区别在于,基于所述服务节点、所述调用顺序和所述层级关系,生成所述异常响应对应的调用结果的步骤包括:

[0128] 步骤a33,基于所述层级关系,确定所述服务节点的调用深度;

[0129] 步骤a34,基于所述调用深度、所述服务节点和所述调用顺序,生成所述异常响应对应的调用结果。

[0130] 本实施例为提高调用结果的生成速度,在生成调用结果的过程中,先确定各服务节点的调用深度,再根据调用深度,将服务节点按照调用顺序,生成调用结果。

[0131] 以下将对各个步骤进行详细说明:

[0132] 步骤a33,基于所述层级关系,确定所述服务节点的调用深度。

[0133] 在本实施例中,定位系统先根据层级关系,确定各个服务节点的调用深度,也即依次确定当前服务节点之后还有几层服务节点,如A服务节点之后还有2层,具体为B服务节点和C服务节点,B服务节点之后还有D服务节点,也即A服务节点之后的第一层为B服务节点和C服务节点,第二层为D服务节点,此时A服务节点的调用深度为3层等。

[0134] 步骤a34,基于所述调用深度、所述服务节点和所述调用顺序,生成所述异常响应对应的调用结果。

[0135] 在本实施例中,定位系统根据调用深度、服务节点和服务节点的调用顺序,生成异常响应对应的调用结果。

[0136] 具体的,基于调用深度,对各个服务节点进行排序,在具体实施时,以调用深度的大小,按从小到大进行排序,然后,以调用深度最小的服务节点作为调用结果的基础,再在调用深度最小的服务节点的基础上,将调用深度大一层的的服务节点搭建进来,以此类推,最后将调用深度最大的调用节点放置在顶部,此后,按照调用顺序,调整各层级的服务节点的位置,最终生成调用结果。

[0137] 如上述例子,A服务节点的调用深度为3,B服务节点的调用深度为2,C服务节点的调用深度为2,D服务节点的调用深度为1,则在生成调用结果的过程中,先以D服务节点作为基础,再在D服务节点的基础上,将B服务节点和C服务节点搭建进来,最后将A服务节点置于顶部,也即,先缓存调用深度小的服务节点的调用结果,然后在生成大一层调用深度的服务节点的调用结果时,直接利用已搭建好的、调用深度小的服务节点的调用结果,以此加快最终调用结果的生成速度。

[0138] 本实施例在生成调用结果的过程中,以调用深度作为生成的参数之一,优先生成调用深度小的服务节点的调用结果,再在生成调用深度大的服务节点的调用结果时,直接利用已搭建好的,调用深度小的服务节点的调用结果,以此加快最终调用结果的生成速度。

[0139] 进一步地,基于本发明异常定位方法第一、第二、第三实施例,提出本发明异常定位方法第四实施例。

[0140] 异常定位方法的第四实施例与异常定位方法的第一、第二、第三实施例的区别在于,异常定位方法还包括:

[0141] 步骤c,若检测到所述服务请求,则基于预设代理组件,生成所述服务请求的所述调用链ID,以及所述服务请求的第一跨度ID;

[0142] 步骤d,基于所述预设代理组件,确定所述服务请求调用的所述服务节点,并基于所述服务节点,生成所述服务请求的第二跨度ID,所述跨度ID包括所述第一跨度ID和所述第二跨度ID;

[0143] 步骤e,将所述跨度ID和所述调用链ID关联保存。

[0144] 本实施例在检测到服务请求时,通过生成调用链ID和跨度ID,跟踪服务请求的流转传递,并保存,便于后续出现异常响应时,定位异常。

[0145] 以下将对各个步骤进行详细说明:

[0146] 步骤c,若检测到所述服务请求,则基于预设代理组件,生成所述服务请求的所述调用链ID,以及所述服务请求的第一跨度ID。

[0147] 本实施例的定位系统包括代理组件,具体可为Nginx代理组件,用于在代理实体服务Tomcat(部署容器,也即服务节点)部署的Restful(满足约束条件和原则的应用程序或设计)接口时,为每一次实际代理触发的服务请求,按照预设规范标准的规范,如OpenTracing,生成全局唯一的trace ID,同时生成第一笔调用的span ID,也即第一跨度ID(第一span ID)。

[0148] 在具体实施过程中,首先,代理组件会将该笔调用的详细信息,如调用的接口名

称、调用时间、返回码、是否有异常,以及生成的trace ID和第一span ID一起输出到代理组件原有的日志中,如Nginx代理组件的access.log中,其中,检测异常响应,即判断是否有异常的异常信息可以通过两种方式获取:1)通过定位系统的过滤器Filter拦截服务请求,捕获如Druid(数据库连接池)、MyBatis(持久层框架)抛出的异常,并解析其中的错误返回码和错误异常信息上报,其中,过滤器Filter内置预设拦截规则和预设过滤规则,在具体实施时,可将异常的日志信息作为过滤标准,以此拦截过滤异常,并按照过滤规则对应的解析方式,解析其中的错误返回码和错误异常信息;2)在业务的公共框架内,若公共框架自己识别出业务失败和逻辑失败,可以根据Opentracing的spanContext(跨越进程边界,传递到下级span的状态)获取trace ID和Span ID,主动上报异常信息,也即定位系统接收业务框架主动上报的异常信息,如SpanContext的状态为异常时,定位系统接收当前服务节点上报的trace ID和Span ID。

[0149] 其次,代理组件将生成的trace ID作为服务节点的服务请求头Header的一部分,然后传递到服务节点Restful接口的HttpRequest的服务请求头内,也即,将trace ID封装进服务请求的头部。

[0150] 步骤d,基于所述预设代理组件,确定所述服务请求调用的所述服务节点,并基于所述服务节点,生成所述服务请求的第二跨度ID,所述跨度ID包括所述第一跨度ID和所述第二跨度ID。

[0151] 在本实施例中,基于预设代理组件,先确定服务请求调用的服务节点,具体可根据服务请求的服务类型确定,也即不同的服务类型对应不同的服务节点,可事先建立服务类型-服务节点映射表来确定。服务节点在接收代理组件代理的请求后,首先解析HttpRequest请求头的信息,确认是否有符合预设规范标准,如OpenTracing规范的trace ID和span ID等信息,若有,则将HttpRequest请求头的信息构建为第一个父span,而Restful接口的span作为父span的子来构建,也即当前服务节点的span为子span,也即第二span ID,因此,后续在定位异常的过程中,span ID包含了第一span ID和第二span ID。

[0152] 步骤e,将所述跨度ID和所述调用链ID关联保存。

[0153] 最后将span ID和trace ID按照预设上报方式,上报到对应的数据库关联保存,如MSS服务端的Hbase数据库中关联保存。

[0154] 此外,还需要说明的是,在跟踪服务请求的流转传递过程中,依靠定位系统中MSS(MessageServer,消息服务器)系统的日志采集组件,采集代理组件产生的日志,需要说明的是,日志采集组件需先部署到代理组件对应的服务器上。在采集日志过程中,若遇到上述产生的预设规范标准,如OpenTracing规范的日志,则将其采集上报到MSS服务端的Hbase数据库,也即在采集日志的过程中还需要判断当前采集的日志是否为OpenTracing规范的日志,若是,则将其上报到MSS服务端的Hbase数据库;若不是,则可将其采集到预设通用数据库中,也即不同的日志分别存储,便于后续根据不同的日志定位异常。

[0155] 最后,依靠MSS服务端MSS-LOGVIEW(网页)从Hbase数据库中捞出代理组件生成的父span和服务节点的子span,并通过统一的trace ID来查询并展示,从而构建出一笔实际服务请求调用经过代理组件,最终调用服务节点的整个过程,便于后续出现异常响应时,可以快速获取到相关的调用信息,从而生成调用结果,再根据调用结果定位异常,实现异常的快速定位。

[0156] 需要说明的是,由于服务节点中可能存在多个响应方对同一请求进行消费,因此,在跟踪确定服务请求调用的服务节点时,还需依次跟踪确定当前服务节点中消费当前服务请求的响应方,也即,在生成服务节点的span ID时,还需要进一步跟踪服务请求在当前服务节点的消费动向。

[0157] 若检测到当前服务节点存在响应方消费当前服务请求,则生成当前响应方对应的响应ID(response ID)。

[0158] 在一实施例中,response ID的生成过程包括:确定响应方的业务功能,如同一服务节点中的取款通知和取款审计等业务功能,然后,基于响应方的业务功能,生成唯一的response ID,这是由于同一服务节点的不同业务功能是由不同的响应方执行的,若是相同的业务功能,理论上在设计时,会将相同的业务功能交由同一响应方执行,因此,可通过响应方的业务功能来生成响应方的唯一response ID,用于区别同一服务节点的不同响应方。

[0159] 在另一实施例中,response ID的生成过程包括:确定响应方的位置信息,具体为响应方所在的链路信息,如trace ID,以及所在服务节点的节点信息,如span ID,然后,基于预设规则,如随机数,或者特定增量的有序数列等,以及位置信息,也即根据预设规则、trace ID和span ID,生成唯一的response ID,预设规则用于区别同一服务节点的不同响应方,位置信息用于区别不同服务节点相同业务功能的响应方。

[0160] 如当前服务节点有甲响应方和乙响应方,为体现同一服务节点对服务请求的多次消费,根据甲响应方所在的链路的链路信息,如trace ID,以及当前服务节点的节点信息,如span ID,以及预设规则(如随机数M),得到甲响应方的response ID1,然后,根据乙响应方所在的链路的链路信息,如trace ID,以及当前服务节点的节点信息,如span ID,以及预设规则(在上一数值M的基础上增加预设阈值m),得到乙响应方的response ID2,此时同一服务节点的不同响应方的response ID是不同的,且也不会跟其他不同服务节点相同业务功能的响应方相同。

[0161] 本实施例在实施异常定位之前,在检测到服务请求时,通过生成调用链ID和跨度ID,跟踪服务请求的流转传递,并保存,也即事先储存服务请求的调用过程,便于后续出现异常响应时,可以获取相关调用信息,从而生成调用结果,再根据调用结果定位异常,实现异常的快速定位。

[0162] 本发明还提供一种异常定位装置。本发明异常定位装置包括:

[0163] 第一确定模块,用于若检测到异常响应,则确定所述异常响应的服务请求,并基于所述服务请求,确定所述异常响应对应的调用信息;

[0164] 生成模块,用于基于所述调用信息,生成所述异常响应对应的调用结果;

[0165] 第二确定模块,用于基于所述调用结果,确定所述异常响应对应的异常对象,并将所述异常对象显示在显示界面。

[0166] 优选地,所述调用信息包括调用链ID和跨度ID,所述生成模块还用于:

[0167] 基于所述调用链ID,确定所述异常响应对应的服务节点;

[0168] 基于所述跨度ID,确定所述服务节点的调用顺序以及所述服务节点的层级关系;

[0169] 基于所述服务节点、所述调用顺序和所述层级关系,生成所述异常响应对应的调用结果。

[0170] 优选地,所述调用信息包括响应ID,所述生成模块还用于:

- [0171] 基于所述跨度ID,确定所述服务节点的调用顺序,
- [0172] 基于所述响应ID,确定所述服务节点的响应方;
- [0173] 基于所述调用顺序和所述响应方,确定所述服务节点的层级关系。
- [0174] 优选地,所述生成模块还用于:
- [0175] 基于所述层级关系,确定所述服务节点的调用深度;
- [0176] 基于所述调用深度、所述服务节点和所述调用顺序,生成所述异常响应对应的调用结果。
- [0177] 优选地,所述生成模块还用于:
- [0178] 基于所述调用顺序,确定所述服务节点中的根服务节点;
- [0179] 基于所述层级关系和所述根服务节点,将所述服务节点生成调用树,并将所述调用树确定为所述异常响应对应的调用结果。
- [0180] 优选地,所述异常定位装置还包括跟踪模块,所述跟踪模块用于:
- [0181] 若检测到所述服务请求,则基于预设代理组件,生成所述服务请求的所述调用链ID,以及所述服务请求的第一跨度ID;
- [0182] 基于所述预设代理组件,确定所述服务请求调用的所述服务节点,并基于所述服务节点,生成所述服务请求的第二跨度ID,所述跨度ID包括所述第一跨度ID和所述第二跨度ID;
- [0183] 将所述跨度ID和所述调用链ID关联保存。
- [0184] 优选地,所述第二确定模块还用于:
- [0185] 识别所述调用结果中的异常节点,并基于所述异常节点的日志信息,确定所述异常节点的异常对象;
- [0186] 确定所述异常节点在所述调用结果的异常位置;
- [0187] 基于所述异常位置,将所述异常对象显示在显示界面。
- [0188] 本发明还提供一种计算机可读存储介质。
- [0189] 本发明计算机可读存储介质上存储有异常定位程序,所述异常定位程序被处理器执行时实现如上所述的异常定位方法的步骤。
- [0190] 其中,在所述处理器上运行的异常定位程序被执行时所实现的方法可参照本发明异常定位方法各个实施例,此处不再赘述。
- [0191] 需要说明的是,在本文中,术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、物品或者系统不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、物品或者系统所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括该要素的过程、方法、物品或者系统中还存在另外的相同要素。
- [0192] 上述本发明实施例序号仅仅为了描述,不代表实施例的优劣。
- [0193] 通过以上的实施方式的描述,本领域的技术人员可以清楚地了解到上述实施例方法可借助软件加必需的通用硬件平台的方式来实现,当然也可以通过硬件,但很多情况下前者是更佳实施方式。基于这样的理解,本发明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如上所述的一个存储介质(如ROM/RAM、磁碟、光盘)中,包括若干指令用以使得一台终端设备(可以是手机,

计算机,服务器,空调器,或者网络设备等)执行本发明各个实施例所述的方法。

[0194] 以上仅为本发明的优选实施例,并非因此限制本发明的专利范围,凡是利用本发明说明书与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,均同理包括在本发明的专利保护范围内。

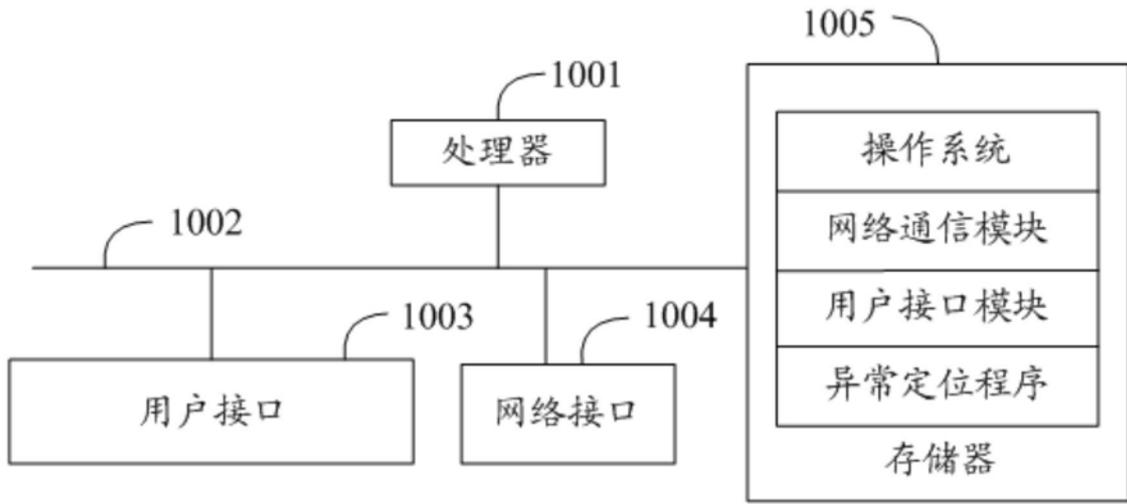


图1

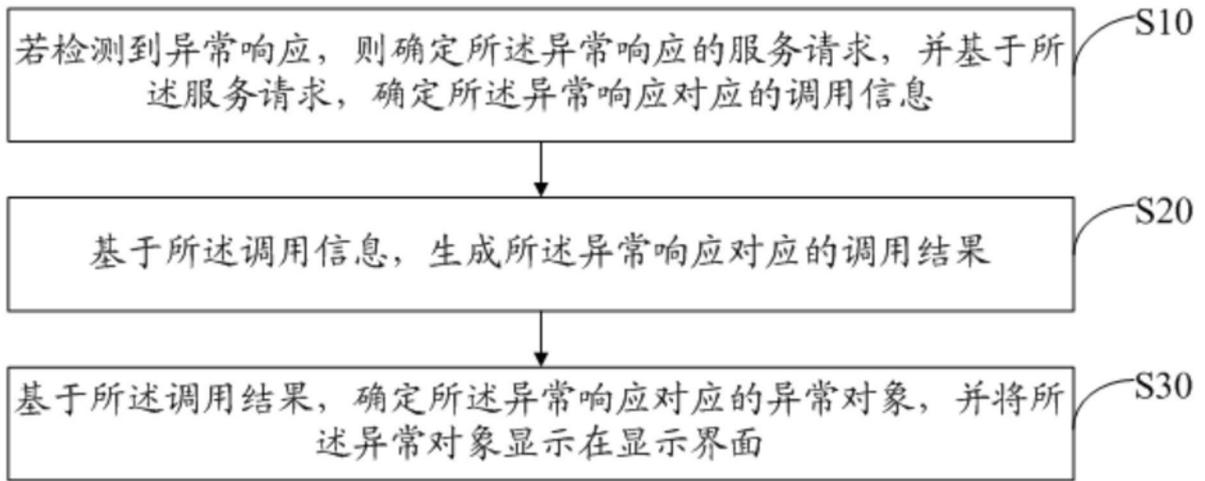


图2