



(12) 发明专利申请

(10) 申请公布号 CN 112241355 A

(43) 申请公布日 2021.01.19

(21) 申请号 202011119826.1

(22) 申请日 2020.10.19

(71) 申请人 恩亿科(北京)数据科技有限公司  
地址 100192 北京市海淀区西小口路66号  
东升科技园·北领地D-1楼1层136A  
(东升地区)

(72) 发明人 黄峰

(74) 专利代理机构 青岛清泰联信知识产权代理  
有限公司 37256  
代理人 张媛媛

(51) Int.Cl.  
G06F 11/34 (2006.01)  
G06F 11/30 (2006.01)

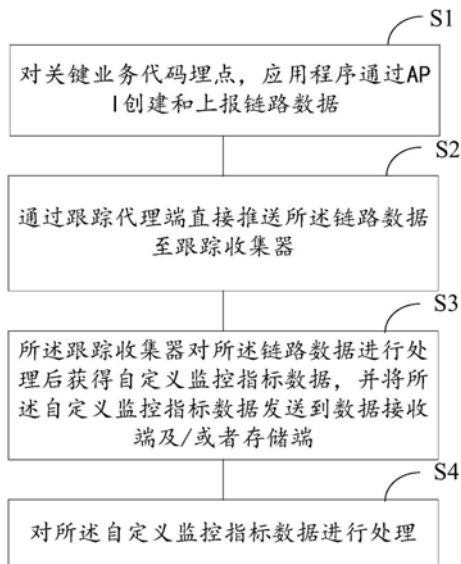
权利要求书1页 说明书7页 附图4页

(54) 发明名称

链路追踪方法、系统、计算机可读存储介质及电子设备

(57) 摘要

本发明公开了一种链路追踪方法、系统、计算机可读存储介质及电子设备,链路追踪方法基于Prometheus的自定义监控服务功能,链路追踪方法包括:步骤S1:对关键业务代码埋点,应用程序通过API创建和上报链路数据;步骤S2:通过跟踪代理端直接推送链路数据至跟踪收集器;步骤S3:跟踪收集器对链路数据进行处理后获得自定义监控指标数据,并将自定义监控指标数据发送到数据接收端及/或者存储端。



1. 一种链路追踪方法,其特征在于,基于Prometheus的自定义监控服务功能,所述链路追踪方法包括:

步骤S1:对关键业务代码埋点,应用程序通过API创建和上报链路数据;

步骤S2:通过跟踪代理端直接推送所述链路数据至跟踪收集器;

步骤S3:所述跟踪收集器对所述链路数据进行处理后获得自定义监控指标数据,并将所述自定义监控指标数据发送到数据接收端及/或者存储端。

2. 如权利要求1所述的链路追踪方法,其特征在于,所述步骤S1中包括,所述跟踪代理端通过UDP的方式接收所述链路数据。

3. 如权利要求1所述的链路追踪方法,其特征在于,所述步骤S3中包括:

步骤S31:通过所述跟踪收集器解析链路数据标签;

步骤S32:通过所述跟踪收集器对所述链路数据进行校验;

步骤S33:通过所述跟踪收集器将完成校验的所述链路数据包装成所述自定义监控指标数据。

4. 如权利要求1所述的链路追踪方法,其特征在于,还包括,

步骤S4:对所述自定义监控指标数据进行处理。

5. 如权利要求4所述的链路追踪方法,其特征在于,所述步骤S4包括:

步骤S41:通过Pushgateway接收所述自定义监控指标数据;

步骤S42:通过计数器将所述自定义监控指标数据聚合生成统计指标数据;

步骤S43:通过图形编辑器根据所述自定义监控指标数据获得展示图形。

6. 一种链路追踪系统,其特征在于,基于Prometheus的自定义监控服务功能,所述链路追踪系统包括:

跟踪客户端,对关键业务代码埋点,应用程序通过API创建和上报链路数据;

跟踪代理端,接收并直接推送所述链路数据;

跟踪收集器,对所述链路数据进行处理后获得自定义监控指标数据,并将所述自定义监控指标数据发送到数据接收端及/或者存储端。

7. 如权利要求6所述的链路追踪系统,其特征在于,所述跟踪收集器解析链路数据标签;所述跟踪收集器对所述链路数据进行校验;所述跟踪收集器将完成校验的所述链路数据包装成所述自定义监控指标数据。

8. 如权利要求6所述的链路追踪系统,其特征在于,还包括数据接收端,所述数据接收端通过Pushgateway接收所述自定义监控指标数据;所述数据接收端通过计数器将所述自定义监控指标数据聚合生成统计指标数据,所述数据接收端通图形编辑器根据所述自定义监控指标数据获得展示图形。

9. 一种计算机可读存储介质,其上存储有计算机程序,其特征在于,该程序被处理器执行时实现如权利要求1至5中任一项所述的链路追踪方法。

10. 一种电子设备,包括存储器、处理器以及存储在所述存储器上并可在所述处理器上运行的计算机程序,其特征在于,所述处理器执行所述计算机程序时实现如权利要求1至5中任一项所述的链路追踪方法。

## 链路追踪方法、系统、计算机可读存储介质及电子设备

### 技术领域

[0001] 本发明涉及一种链路追踪方法、系统、计算机可读存储介质及电子设备,具体地说,尤其涉及一种基于Prometheus的自定义监控服务功能实现链路追踪方法、系统、计算机可读存储介质及电子设备。

### 背景技术

[0002] 在实际生产环境中,当有一个请求里包含有多个服务单元,如果出现执行缓慢延迟过高时,想要查看每个服务单元的调用执行情况,在没有明显异常和报错的情况下很难定位是哪个具体服务点出了问题,这时就需要链路追踪。通过链路追踪可以获取一个请求的完整执行过程,UI图示能够展示每个服务单元的调用关系和执行信息。

[0003] 传统的监控系统,通过在业务代码中埋点实现对业务数据的监控,但是只能对某短时间的指标数据进行统计和聚合,无法实现对单个请求进行完整的链路跟踪。

[0004] 现有的链路追踪系统基本架构为client发送请求,collector或者server处理请求,data store做数据存储,最后提供简单的UI进行查询和展示,但是无法直接结合其他监控指标分析请求数据,只能提供有限的参考作用。现有技术的缺点如下:

[0005] 1、链路追踪系统UI比较简单,一些个性化的分析需求无法满足;

[0006] 2、独立的追踪系统和监控系统,会增加部署和维护成本;

[0007] 3、链路追踪功能和系统监控功能分离,对于复杂问题的分析和定位问题需要频繁切换系统;

[0008] 4、Prometheus通过埋点实现监控业务指标,实现链路追踪需要重复埋点。

[0009] 因此对于有类似功能的链路追踪技术和数据监控技术,开发人员希望能够尽量集中在一套系统中,充分利用链路追踪图和监控统计图之间的辅助作用,减少系统之间切换的繁复操作,通过一套系统的数据展示来达到问题的快速分析和定位。

### 发明内容

[0010] 针对上述问题,本发明提供一种链路追踪方法,其中,基于Prometheus的自定义监控服务功能,所述链路追踪方法包括:

[0011] 步骤S1:对关键业务代码埋点,应用程序通过API创建和上报链路数据;

[0012] 步骤S2:通过跟踪代理端直接推送所述链路数据至跟踪收集器;

[0013] 步骤S3:所述跟踪收集器对所述链路数据进行处理后获得自定义监控指标数据,并将所述自定义监控指标数据发送到数据接收端及/或者存储端。

[0014] 上述的链路追踪方法,其中,所述步骤S1中包括,所述跟踪代理端通过UDP的方式接收所述链路数据。

[0015] 上述的链路追踪方法,其中,所述步骤S3中包括:

[0016] 步骤S31:通过所述跟踪收集器解析链路数据标签;

[0017] 步骤S32:通过所述跟踪收集器对所述链路数据进行校验;

[0018] 步骤S33:通过所述跟踪收集器将完成校验的所述链路数据包装成所述自定义监控指标数据。

[0019] 上述的链路追踪方法,其中,还包括,

[0020] 步骤S4:对所述自定义监控指标数据进行处理。

[0021] 上述的链路追踪方法,其中,所述步骤S4包括:

[0022] 步骤S41:通过Pushgateway接收所述自定义监控指标数据;

[0023] 步骤S42:通过计数器将所述自定义监控指标数据聚合生成统计指标数据;

[0024] 步骤S43:通过图形编辑器根据所述自定义监控指标数据获得展示图形。

[0025] 本发明还提供一种链路追踪系统,其中,基于Prometheus的自定义监控服务功能,所述链路追踪系统包括:

[0026] 跟踪客户端,对关键业务代码埋点,应用程序通过API创建和上报链路数据;

[0027] 跟踪代理端,接收并直接推送所述链路数据;

[0028] 跟踪收集器,对所述链路数据进行处理后获得自定义监控指标数据,并将所述自定义监控指标数据发送到数据接收端及/或者存储端。

[0029] 上述的链路追踪系统,其中,所述跟踪收集器解析链路数据标签;所述跟踪收集器对所述链路数据进行校验;所述跟踪收集器将完成校验的所述链路数据包装成所述自定义监控指标数据。

[0030] 上述的链路追踪系统,其中,还包括数据接收端,所述数据接收端通过Pushgateway接收所述自定义监控指标数据;所述数据接收端通过计数器将所述自定义监控指标数据聚合生成统计指标数据,所述数据接收端通图形编辑器根据所述自定义监控指标数据获得展示图形。

[0031] 本发明还提供一种计算机可读存储介质,其上存储有计算机程序,其特征在于,该程序被处理器执行时实现如上述中任一项所述的链路追踪方法。

[0032] 本发明还提供一种电子设备,包括存储器、处理器以及存储在所述存储器上并可在所述处理器上运行的计算机程序,其中,所述处理器执行所述计算机程序时实现如上述中任一项所述的链路追踪方法。

[0033] 综上所述,本发明相对于现有技术其功效在于:

[0034] 1) 通过现有Prometheus系统监控的环境中可以直接实现链路追踪减少开发成本;

[0035] 2) 使得在一套系统中实现系统监控和链路追踪两种功能;

[0036] 3) 省去不同系统之间来回切换的繁复操作;

[0037] 4) 方便开发人员在统一的监控图中结合追踪信息和监控信息快速分析定位问题;

[0038] 5) 提高问题发现和定位的查找效率;

[0039] 6) 减少了多套系统的部署和运维成本。

[0040] 本发明的其它特征和优点将在随后的说明书中阐述,并且,部分地从说明书中变得显而易见,或者通过实施本发明而了解。本发明的目的和其他优点可通过在说明书、权利要求书以及附图中所指出的结构来实现和获得。

## 附图说明

[0041] 为了更清楚地说明本发明实施例或现有技术中的技术方案,下面将对实施例或现

有技术描述中所需要使用的附图作一简单地介绍,显而易见地,下面描述中的附图是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

- [0042] 图1为本发明的链路追踪方法的流程图;
- [0043] 图2为图1中步骤S3的流程图;
- [0044] 图3为图1中步骤S4的流程图;
- [0045] 图4为本发明的链路追踪系统的结构示意图;
- [0046] 图5为本发明的链路追踪系统的应用示意图;
- [0047] 图6为本发明电子设备的结构示意图。

### 具体实施方式

[0048] 为使本发明实施例的目的、技术方案和优点更加清楚,下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地说明,显然,所描述的实施例是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,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,都属于本发明保护的范围。

[0049] 本发明的示意性实施例及其说明用于解释本发明,但并不作为对本发明的限定。另外,在附图及实施方式中所使用相同或类似标号的元件/构件是用来代表相同或类似部分。

[0050] 关于本文中所使用的“第一”、“第二”、“S1”、“S2”、…等,并非特别指称次序或顺位的意思,也非用以限定本发明,其仅为了区别以相同技术用语描述的元件或操作。

[0051] 关于本文中所使用的方向用语,例如:上、下、左、右、前或后等,仅是参考附图的方向。因此,使用的方向用语是用来说明并非用来限制本创作。

[0052] 关于本文中所使用的“包含”、“包括”、“具有”、“含有”等等,均为开放性的用语,即意指包含但不限于。

[0053] 关于本文中所使用的“及/或”,包括所述事物的任一或全部组合。

[0054] 关于本文中的“多个”包括“两个”及“两个以上”;关于本文中的“多组”包括“两组”及“两组以上”。

[0055] 关于本文中所使用的用语“大致”、“约”等,用以修饰任何可以微变化的数量或误差,但这些微变化或误差并不会改变其本质。一般而言,此类用语所修饰的微变化或误差的范围在部分实施例中可为20%,在部分实施例中可为10%,在部分实施例中可为5%或是其他数值。本领域技术人员应当了解,前述提及的数值可依实际需求而调整,并不以此为限。

[0056] 某些用以描述本申请的用词将于下或在此说明书的别处讨论,以提供本领域技术人员在有关本申请的描述上额外的引导。

[0057] Prometheus作为监控系统,通过在业务代码中埋点实现对指定业务数据的监控,能对某短时间的指标数据进行统计和展示,无法实现对完整请求链路的跟踪。

[0058] 从实现过程和功能来看,链路追踪和Prometheus的自定义监控有相互重叠的部分,它们的实现原理都必须经过代码埋点,数据存储和查询展示这三步。如果可以将链路追踪添功能加到Prometheus中,方便开发人员通过一套系统快速分析和定位问题,有效减少冗余工作。

[0059] 请参照图1,图1为本发明的链路追踪方法的流程图。如图1所示,本发明的链路追踪方法基于Prometheus的自定义监控服务功能,链路追踪方法包括:

[0060] 步骤S1:对关键业务代码埋点,应用程序通过API创建和上报链路数据,具体地说,在跟踪客户端(Tracing Client)对关键业务代码埋点,应用程序通过API创建和上报链路(span)数据,并把跟踪(trace)信息发送给跟踪代理端(Tracing Agent)。

[0061] 步骤S2:通过跟踪代理端直接推送所述链路数据至跟踪收集器;其中,为了保持数据的快速传输,跟踪代理端采用UDP的方式接收从跟踪客户端(Tracing Client)发出的链路(span)数据,允许少量链路(span)数据的丢失,不影响业务和整个链路追踪。

[0062] 步骤S3:所述跟踪收集器(Tracing Collector)对所述链路数据进行处理后获得自定义监控指标数据,并将所述自定义监控指标数据发送到数据接收端及/或者存储端。

[0063] 步骤S4:对所述自定义监控指标数据进行处理。

[0064] 请参照图2,图2为图1中步骤S3的流程图。如图2所示,所述步骤S3中包括:

[0065] 步骤S31:通过所述跟踪收集器解析链路数据标签;

[0066] 步骤S32:通过所述跟踪收集器对所述链路数据进行校验;

[0067] 步骤S33:通过所述跟踪收集器将完成校验的所述链路数据包装成所述自定义监控指标数据。

[0068] 请参照图3,图3为图1中步骤S4的流程图。如图3所示,所述步骤S4包括:

[0069] 步骤S41:通过Pushgateway接收所述自定义监控指标数据;

[0070] 步骤S42:通过计数器将所述自定义监控指标数据聚合生成统计指标数据;

[0071] 步骤S43:通过图形编辑器根据所述自定义监控指标数据获得展示图形。

[0072] 由此,本发明基于Prometheus的自定义监控服务功能实现链路追踪,无需重复埋点,对多次请求计数统计的同时对单次请求进行链路追踪;同时打通了链路追踪和系统监控的技术隔阂;并且提供链路追踪信息和监控统计指标的统一图形化展示;更拥有更加丰富的图形展示,为技术人员分析问题提供依据参考。

[0073] 请参照图4,图4为本发明的链路追踪系统的结构示意图。如图4所示,本发明的链路追踪系统,基于Prometheus的自定义监控服务功能,所述链路追踪系统包括:

[0074] 跟踪客户端(Tracing Client) 11,对关键业务代码埋点,应用程序通过API创建和上报链路数据;

[0075] 跟踪代理端(Tracing Agent) 12,接收并直接推送所述链路数据;

[0076] 跟踪收集器(Tracing Collector) 13,对所述链路数据进行处理后获得自定义监控指标数据,并将所述自定义监控指标数据发送到数据接收端14及/或者存储端15,其中,所述跟踪收集器13解析链路数据标签;所述跟踪收集器对所述链路数据进行校验;所述跟踪收集器将完成校验的所述链路数据包装成所述自定义监控指标数据。

[0077] 具体地说,跟踪客户端(Tracing Client) 11:OpenTracing协议,OpenTracing是一个轻量级的标准化层,位于应用程序和追踪或日志分析程序之间,解决了不同的分布式追踪系统API不兼容的问题;

[0078] 跟踪代理端(Tracing Agent) 12:作为跟踪客户端(Tracing Client) 11的一个代理将跟踪客户端(Tracing Client) 11和跟踪收集器(Tracing Collector) 13解耦,屏蔽了跟踪客户端(Tracing Client) 11和跟踪收集器(Tracing Collector) 13之间的连接细节,

直接推送从跟踪客户端 (Tracing Client) 11收集上报的链路数据给跟踪收集器 (Tracing Collector) 13;

[0079] 跟踪收集器 (Tracing Collector) 13:负责接收跟踪代理端 (Tracing Agent) 12发送的链路数据,解析span标签,通过检验、转换和包装等处理得到符合规范的自定义监控指标数据 (metric),以推送 (push) 方式自定义监控指标数据 (metric) 发送到指定的数据接收端14或者存储到指定存储端15中。

[0080] 进一步地,链路追踪系统还包括数据接收端14及/或者存储端15,所述数据接收端14通过Pushgateway接收所述自定义监控指标数据,所述数据接收端14通过计数器将所述自定义监控指标数据聚合生成统计指标数据,所述数据接收端14通过图形编辑器根据所述自定义监控指标数据获得展示图形;存储端15接收所述自定义监控指标数据进行存储备份。

[0081] 具体地说,数据接收端14:对接Pushgateway,接收跟踪收集器 (Tracing Collector) 13发送来的自定义监控指标数据 (metric) 然后等待服务监控系统 (Prometheus) 拉取数据;存储端 (Data Store) 15:存储链路数据,支持写入Elastic Search等主流数据库中。

[0082] 请参照图5,图5为本发明的链路追踪系统的应用示意图。以下结合图5,具体说明本发明链路追踪系统的工作过程,其中在本实施例中,数据接收端14包括服务监控系统 (Prometheus) 141及图形编辑器 (grafana) 142。具体说明如下:

[0083] 服务监控系统 (Prometheus) 141采取了一种新的模型,将采集时序数据作为整个系统的核心,无论是告警还是构建监控图表,都是通过操纵时序数据来实现的。Prometheus通过指标的名称以及label (key/value) 的组合来识别时序数据,每个label代表一个维度,可以增加或者减少label来控制所选择的时序数据。Prometheus借助多维度的数据模型,以及强大的查询语言满足了微服务架构底下对监控的要求:既能知道服务整体的运行情况,也能够保持足够的粒度,知道某个组件的运行情况。

[0084] (1) 在跟踪客户端 (Tracing Client) 11对关键业务代码埋点,应用程序通过API创建和上报链路数据,把跟踪信息发送给跟踪代理端 (Tracing Agent) 12;

[0085] (2) 跟踪代理端 (Tracing Agent) 12是数据从跟踪客户端 (Tracing Client) 11转发到跟踪收集器 (Tracing Collector) 13的中转站。为了保持数据的快速传输,采用UDP的方式接收从跟踪客户端 (Tracing Client) 11发出的链路数据,允许少量链路数据的丢失,不影响业务和整个链路追踪;

[0086] (3) 跟踪收集器 (Tracing Collector) 13对数据进行校验和处理,比如发送到下游Pushgateway的数据格式是否正确,时间范围是否符合规定,校验过的数据包装成Pushgateway能够识别和接收的自定义监控指标数据 (metric),例如链路数据经过解析后形成链路追踪自定义监控指标数据 (metric),同时复制一份在缓存中用于服务监控系统 (Prometheus) 141的计数器聚合生成统计指标数据,下游可以储到指定后端存储或者直接以推送的方式发向Pushgateway;

[0087] (4) Pushgateway允许任何客户端向其push符合规范的自定义监控指标,只需要接收数据,服务监控系统 (Prometheus) 141会周期性的定时拉取数据;

[0088] (5) 图形编辑器 (grafana) 142从服务监控系统 (Prometheus) 141或者存储端 (Data Store) 15中查询数据,通过自定义dashboard可以绘制满足个性化需求的展示图形。

[0089] 其中,在本实施例中跟踪客户端(Tracing Client)11及跟踪代理端(Tracing Agent)12可部署与主机(Host)或容器(Container)中。

[0090] 本发明还提供一种计算机可读存储介质,其上存储有计算机程序,其中,该程序被处理器执行时实现如上述中任一项所述的链路追踪方法。

[0091] 请参照图6,图6为本发明电子设备的结构示意图。如图6所示,本发明的电子设备,电子设备可以包括处理器21以及存储有计算机程序指令的存储器22。

[0092] 具体地,上述处理器21可以包括中央处理器(CPU),或者特定集成电路(Application Specific Integrated Circuit,简称为ASIC),或者可以被配置成实施本申请实施例的一个或多个集成电路。

[0093] 其中,存储器22可以包括用于数据或指令的大容量存储器。举例来说而非限制,存储器22可包括硬盘驱动器(Hard Disk Drive,简称为HDD)、软盘驱动器、固态驱动器(Solid State Drive,简称为SSD)、闪存、光盘、磁光盘、磁带或通用串行总线(Universal Serial Bus,简称为USB)驱动器或者两个或更多个以上这些的组合。在合适的情况下,存储器22可包括可移除或不可移除(或固定)的介质。在合适的情况下,存储器22可在数据处理装置的内部或外部。在特定实施例中,存储器22是非易失性(Non-Volatile)存储器。在特定实施例中,存储器22包括只读存储器(Read-Only Memory,简称为ROM)和随机存取存储器(Random Access Memory,简称为RAM)。在合适的情况下,该ROM可以是掩模编程的ROM、可编程ROM(Programmable Read-Only Memory,简称为PROM)、可擦除PROM(Erasable Programmable Read-Only Memory,简称为EPROM)、电可擦除PROM(Electrically Erasable Programmable Read-Only Memory,简称为EEPROM)、电可改写ROM(Electrically Alterable Read-Only Memory,简称为EAROM)或闪存(FLASH)或者两个或更多个以上这些的组合。在合适的情况下,该RAM可以是静态随机存取存储器(Static Random-Access Memory,简称为SRAM)或动态随机存取存储器(Dynamic Random Access Memory,简称为DRAM),其中,DRAM可以是快速页模式动态随机存取存储器(Fast Page Mode Dynamic Random Access Memory,简称为FPMDRAM)、扩展数据输出动态随机存取存储器(Extended Data Out Dynamic Random Access Memory,简称为EDODRAM)、同步动态随机存取内存(Synchronous Dynamic Random-Access Memory,简称SDRAM)等。

[0094] 存储器22可以用来存储或者缓存需要处理和/或通信使用的各种数据文件,以及处理器21所执行的可能的计算机程序指令。

[0095] 处理器21通过读取并执行存储器22中存储的计算机程序指令,以实现上述实施例中的任意一种链路追踪方法。

[0096] 在其中一些实施例中,电子设备还可包括通信接口23和总线20。其中,如图5所示,处理器21、存储器22、通信接口23通过总线20连接并完成相互间的通信。

[0097] 通信端口23可以实现与其他部件例如:外接设备、图像/数据采集设备、数据库、外部存储以及图像/数据处理工作站等之间进行数据通信。

[0098] 总线20包括硬件、软件或两者,将电子设备的部件彼此耦接在一起。总线20包括但不限于以下至少之一:数据总线(Data Bus)、地址总线(Address Bus)、控制总线(Control Bus)、扩展总线(Expansion Bus)、局部总线(Local Bus)。举例来说而非限制,总线20可包括图形加速接口(Accelerated Graphics Port,简称为AGP)或其他图形总线、增强工业标



准架构(Extended Industry Standard Architecture, 简称为EISA)总线、前端总线(Front Side Bus, 简称为FSB)、超传输(Hyper Transport, 简称为HT)互连、工业标准架构(Industry Standard Architecture, 简称为ISA)总线、无线带宽(InfiniBand)互连、低引脚数(Low Pin Count, 简称为LPC)总线、存储器总线、微信道架构(Micro Channel Architecture, 简称为MCA)总线、外围组件互连(Peripheral Component Interconnect, 简称为PCI)总线、PCI-Express(PCI-X)总线、串行高级技术附件(Serial Advanced Technology Attachment, 简称为SATA)总线、视频电子标准协会局部(Video Electronics Standards Association Local Bus, 简称为VLB)总线或其他合适的总线或者两个或更多个以上这些的组合。在合适的情况下,总线20可包括一个或多个总线。尽管本申请实施例描述和示出了特定的总线,但本申请考虑任何合适的总线或互连。

[0099] 该电子设备可以执行上述中的任一种链路追踪方法。

[0100] 以上所述实施例的各技术特征可以进行任意的组合,为使描述简洁,未对上述实施例中的各个技术特征所有可能的组合都进行描述,然而,只要这些技术特征的组合不存在矛盾,都应当认为是本说明书记载的范围。

[0101] 综上所述,本发明基于Prometheus的自定义监控服务功能实现链路追踪,无需重复埋点,对多次请求计数统计的同时对单次请求进行链路追踪;同时打通了链路追踪和系统监控的技术隔阂;并且提供链路追踪信息和监控统计指标的统一图形化展示;更拥有更加丰富的图形展示,为技术人员分析问题提供依据参考。

[0102] 尽管参照前述实施例对本发明进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本发明各实施例技术方案的精神和范围。

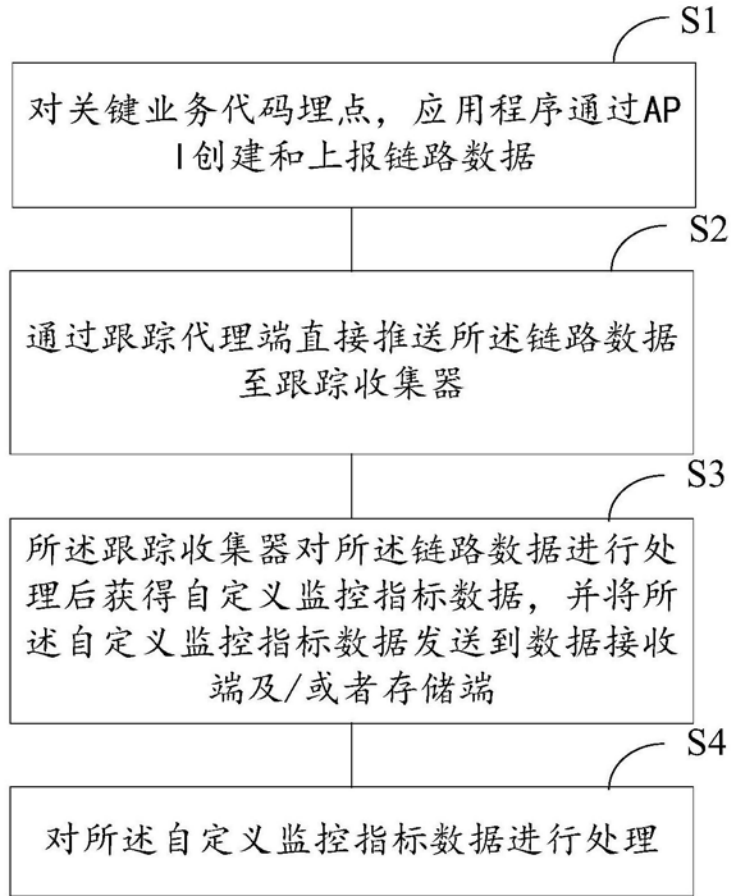


图1

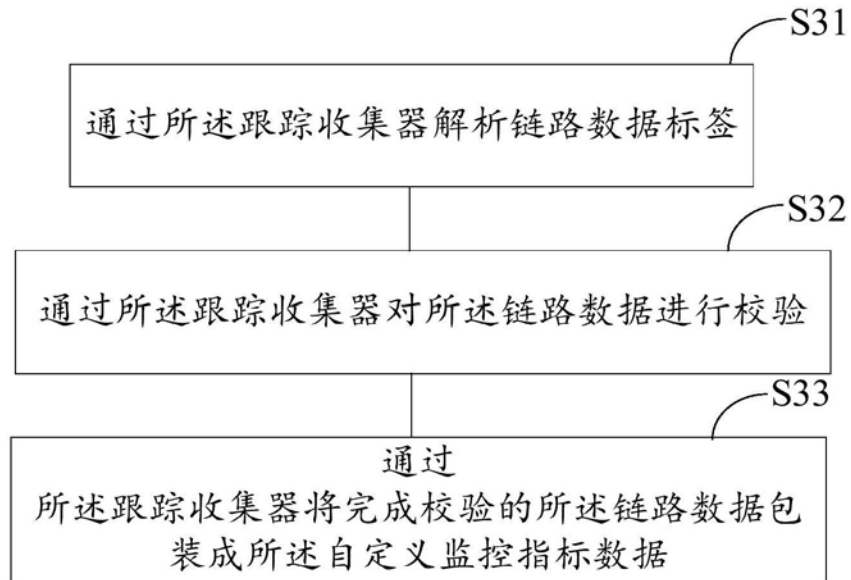


图2

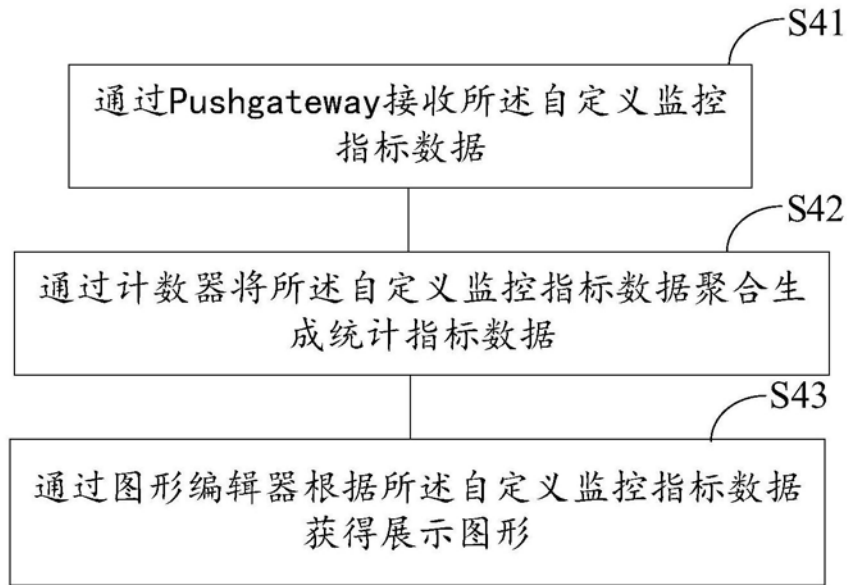


图3

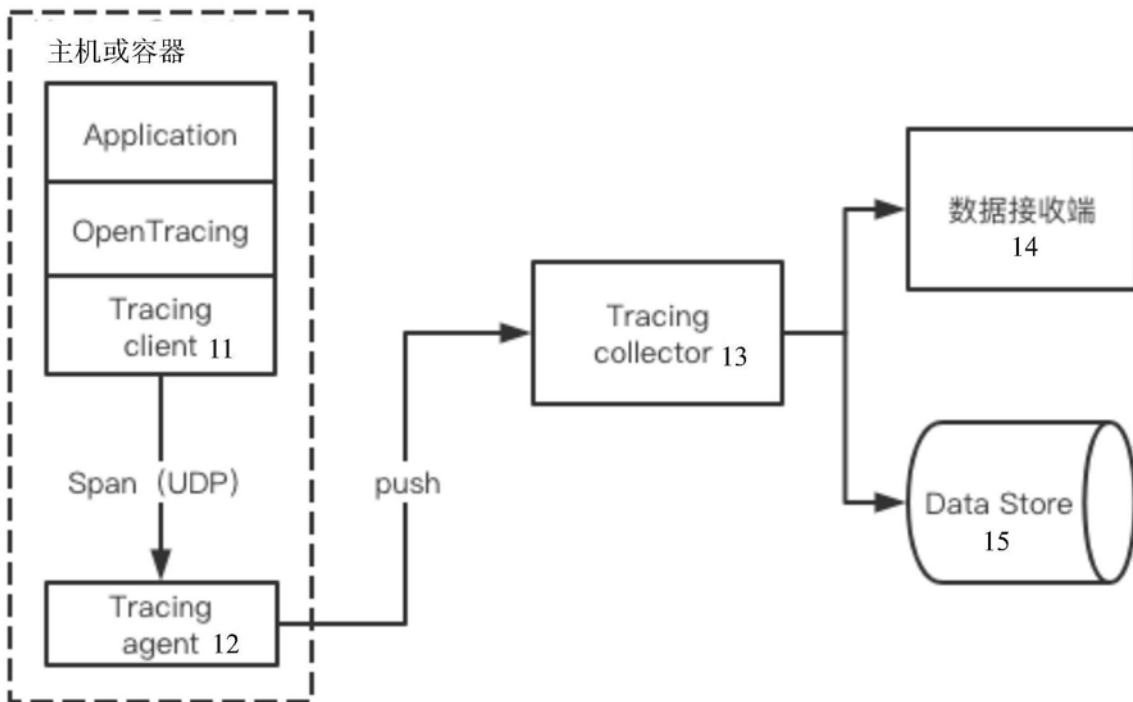


图4

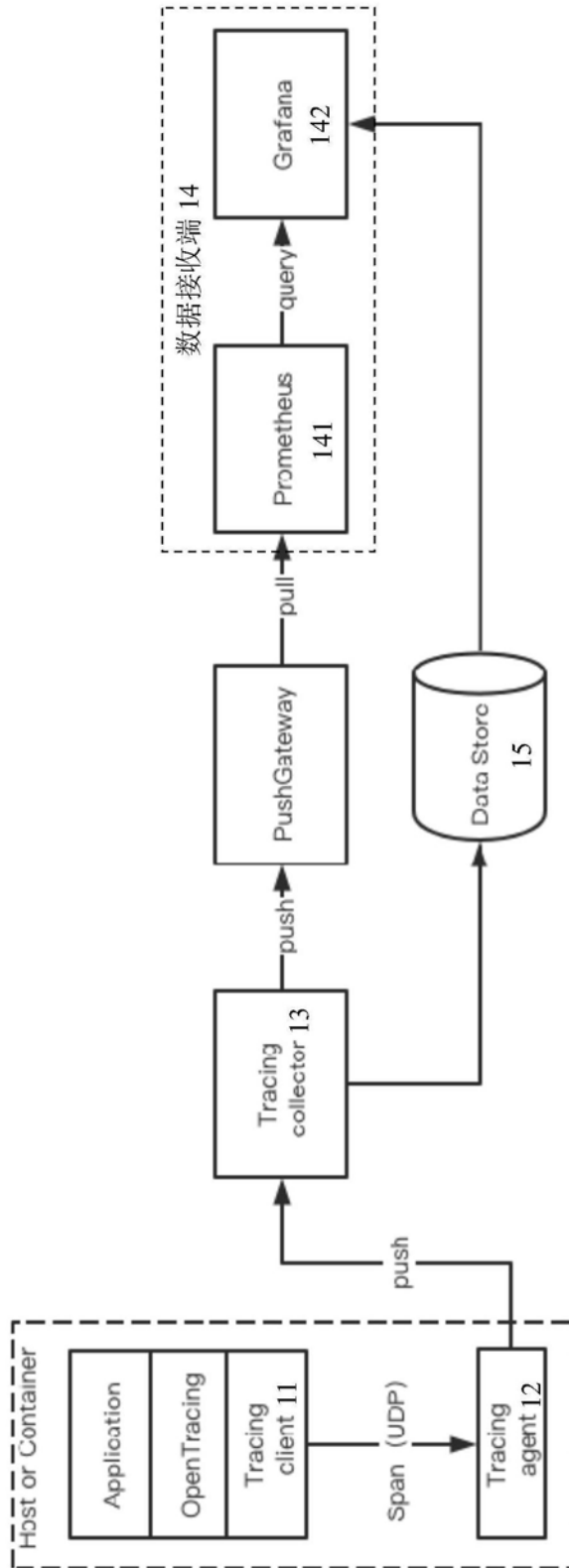


图5

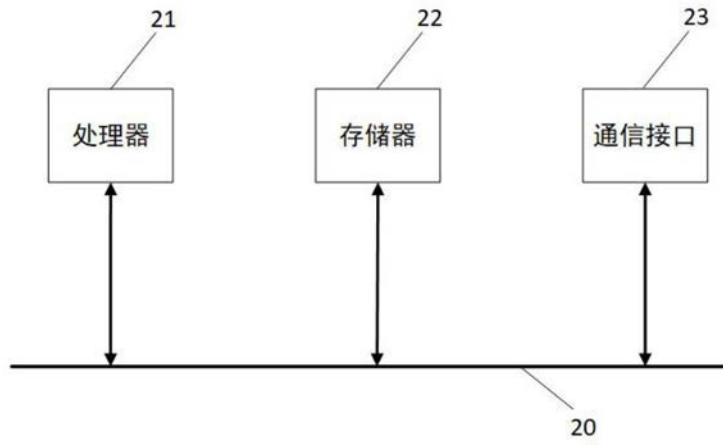


图6