



(12)发明专利申请

(10)申请公布号 CN 108282531 A

(43)申请公布日 2018.07.13

(21)申请号 201810068918.8

(22)申请日 2018.01.24

(71)申请人 深圳市欧辰技术有限公司

地址 518000 广东省深圳市南山区粤海街道高新中一道2号长园新材料港1栋4楼南

(72)发明人 余童

(74)专利代理机构 深圳力拓知识产权代理有限公司 44313

代理人 龚健

(51)Int.Cl.

H04L 29/08(2006.01)

H04L 29/12(2006.01)

G06F 17/30(2006.01)

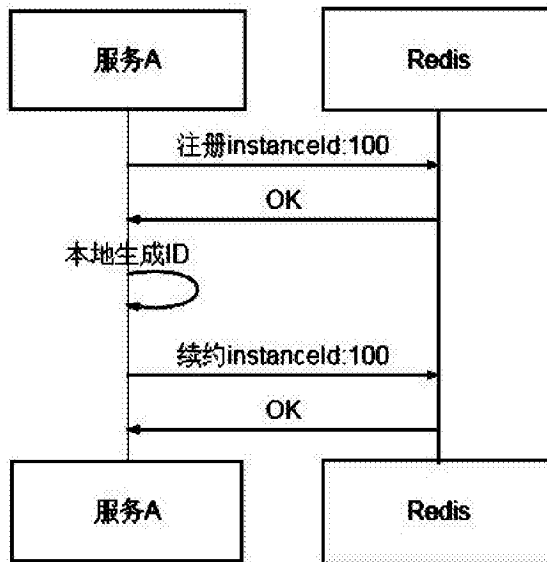
权利要求书1页 说明书2页 附图1页

(54)发明名称

一种基于分布式系统的唯一性ID生成方法

(57)摘要

本发明公开了一种基于分布式系统的唯一性ID生成方法,包含以下步骤:A、首先项目应用启动未完成之后,开启同步定时任务,定期调用redis注册,续约instanceId;B、然后注册一个特定时长的ID服务生成器来生成服务实例,之后归并入步骤A中的同步定时任务中去,当ID服务生成器实例存在时,会对服务进行续约,如果续约不成功进入一个短暂的等待阶段,之后会重新生成一个新的ID服务生成器,避免业务的间断情况,当ID服务生成器实例不存在时,会立即重新生成新的ID服务实例生成器,本发明有效解决了使用分布式系统时,系统架构面临全局ID生成可能会出现重复ID,导致整个业务无法正常进行的客观问题。



1. 一种基于分布式系统的唯一性ID生成方法,其特征在于,包含以下步骤:

首先项目应用启动未完成之后,开启同步定时任务,定期调用redis注册,续约instanceId;

然后注册一个特定时长的ID服务生成器来生成服务实例,之后归并入步骤A中的同步定时任务中去,当ID服务生成器实例存在时,会对服务进行续约,如果续约不成功进入一个短暂的等待阶段,之后会重新生成一个新的ID服务生成器,避免业务的间断情况,当ID服务生成器实例不存在时,会立即重新生成新的ID服务实例生成器。

2. 根据权利要求1所述的基于分布式系统的唯一性ID生成方法,其特征在于,ID服务实例生成器的生成是根据所在网关的IPV4地址,进行机器唯一服务id的获取工作。

3. 根据权利要求1所述的基于分布式系统的唯一性ID生成方法,其特征在于,ID的具体生成策略:ID为64位的2进制数,最后会将其转化为14位的10进制长整型数。

4. 根据权利要求1所述的基于分布式系统的唯一性ID生成方法,其特征在于,在生成64位2进制数时,最高位的字节会被标记为不可用,前41位字节是以微秒为单位的timestamp,紧接着10个字节为redis根据IPV4生成的机器ID,最后12个字节是累加计数器,该计数器在系统内。

一种基于分布式系统的唯一性ID生成方法

技术领域

[0001] 本发明涉及一种ID生成方法,具体是一种基于分布式系统的唯一性ID生成方法。

背景技术

[0002] 当前为解决分布式系统的唯一ID有以下解决办法:

- 1.使用UUID生成唯一ID;
- 2.使用数据库sequence (序列,一种ID自增策略)
- 3.使用twitter的雪花算法生成全局ID;
- 4.使用MongoDB的objectId来生成全局唯一ID。

[0003] 上述方法存在以下缺点:

UUID:无序,过长,难以建立索引。

[0004] MongoDB ObjectId:过长,非纯数字。

[0005] 数据库自增序列:中心生成,有性能瓶颈。

[0006] 雪花算法:中心生成,独立部署ID生成器。

发明内容

[0007] 本发明的目的在于提供一种基于分布式系统的唯一性ID生成方法,以解决上述背景技术中提出的问题。

[0008] 为实现上述目的,本发明提供如下技术方案:

一种基于分布式系统的唯一性ID生成方法,包含以下步骤:

A、首先项目应用启动未完成之后,开启同步定时任务,定期调用redis注册,续约instanceId;

B、注册一个特定时长的ID服务生成器来生成服务实例,之后归并入步骤A中的同步定时任务中去,当ID服务生成器实例存在时,会对服务进行续约,如果续约不成功进入一个短暂的等待阶段,之后会重新生成一个新的ID服务生成器,避免业务的间断情况,当ID服务生成器实例不存在时,会立即重新生成新的ID服务实例生成器。

[0009] 作为本发明再进一步的方案:ID服务实例生成器的生成是根据所在网关的IPV4地址,进行机器唯一服务id的获取工作。

[0010] 作为本发明再进一步的方案:ID的具体生成策略:ID为64位的2进制数,最后会将其转化为14位的10进制长整型数。

[0011] 作为本发明再进一步的方案:在生成64位2进制数时,最高位的字节会被标记为不可用,前41位字节是以微秒为单位的timestamp,紧接着10个字节为redis根据IPV4生成的机器ID,最后12个字节是累加计数器,该计数器在系统内。

[0012] 与现有技术相比,本发明的有益效果是:1、效率高,速度快;2、综合了其他技术的优点,有效解决了使用分布式系统时,系统架构面临全局ID生成可能会出现重复ID,导致整个业务无法正常进行的客观问题。

附图说明

[0013] 图1本发明的工作流程图。

具体实施方式

[0014] 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、完整地描述,显然,所描述的实施例仅仅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。

[0015] 请参阅图1,本发明实施例中,一种基于分布式系统的唯一性ID生成方法,包含以下步骤:

A、首先项目应用启动未完成之后,开启同步定时任务,定期调用redis注册,续约instanceId;

B、然后注册一个特定时长的ID服务生成器来生成服务实例。之后归并入步骤1中的同步定时任务中去,当ID服务生成器实例存在时,会对服务进行续约,如果续约不成功进入一个短暂的等待阶段,之后会重新生成一个新的ID服务生成器,避免业务的间断情况。当ID服务生成器实例不存在时,会立即重新生成新的ID服务实例生成器。

[0016] C、ID服务实例生成器的生成,是根据所在网关的IPV4地址,进行机器唯一服务id的获取工作;

D、ID的具体生成策略:ID为64位的2进制数,最后会将其转化为14位的10进制长整型数

E、在生成64位2进制数时,最高位的字节会被标记为不可用,前41位字节是以微秒为单位的timestamp,紧接着10个字节为redis根据IPV4生成的机器ID,最后12个字节是累加计数器,该计数器在软件内。

[0017] F、因为计算机存放数字都是存放数字的补码,正数的源码、补码、反码都一样,负数的补码是其反码加一。符号为拉取反操作时不变,做逻辑与、或、非、异或操作时要参与运算。

[0018] 对于本领域技术人员而言,显然本发明不限于上述示范性实施例的细节,而且在不背离本发明的精神或基本特征的情况下,能够以其他的具体形式实现本发明。因此,无论从哪一点来看,均应将实施例看作是示范性的,而且是非限制性的,本发明的范围由所附权利要求而不是上述说明限定,因此旨在将落在权利要求的等同要件的含义和范围内的所有变化囊括在本发明内。不应将权利要求中的任何附图标记视为限制所涉及的权利要求。

[0019] 此外,应当理解,虽然本说明书按照实施方式加以描述,但并非每个实施方式仅包含一个独立的技术方案,说明书的这种叙述方式仅仅是为清楚起见,本领域技术人员应当将说明书作为一个整体,各实施例中的技术方案也可以经适当组合,形成本领域技术人员可以理解的其他实施方式。

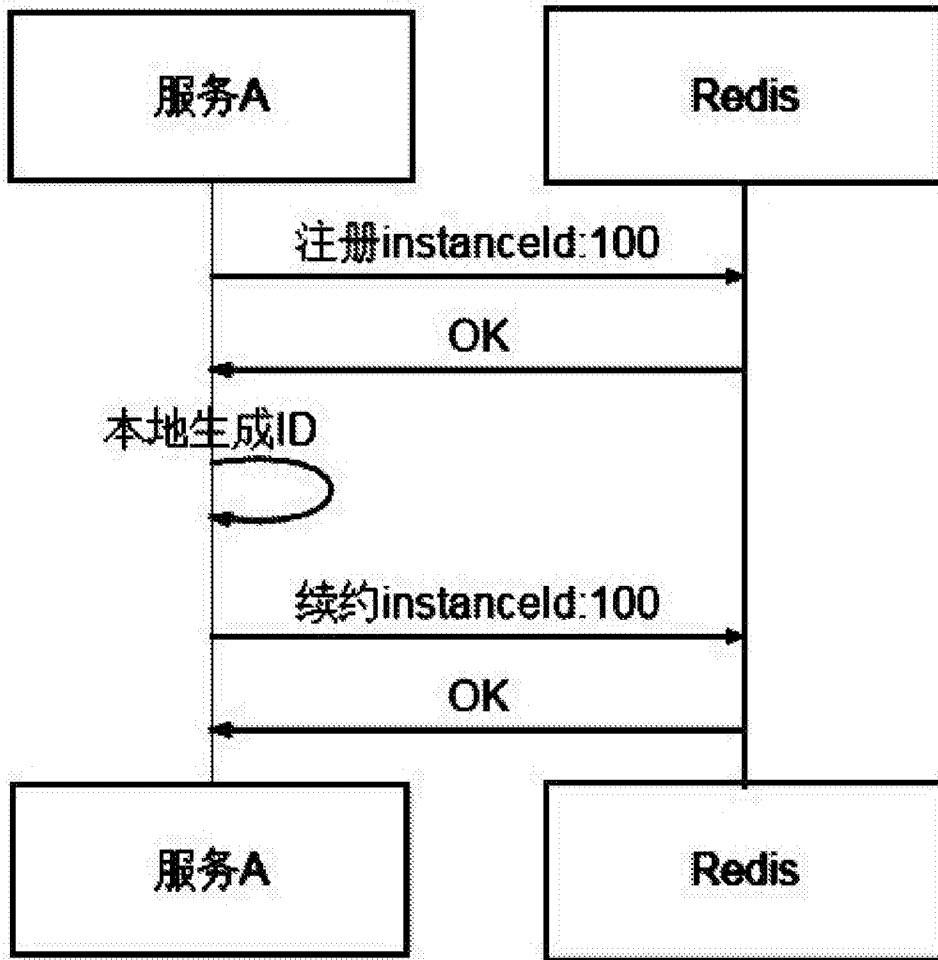


图1