



(12)发明专利

(10)授权公告号 CN 105703941 B

(45)授权公告日 2019.11.05

(21)申请号 201511029777.1

(22)申请日 2015.12.31

(65)同一申请的已公布的文献号
申请公布号 CN 105703941 A

(43)申请公布日 2016.06.22

(73)专利权人 山石网科通信技术股份有限公司
地址 215163 江苏省苏州市高新区景润路
181号

(72)发明人 渠海峡 崔应杰

(74)专利代理机构 北京康信知识产权代理有限
责任公司 11240

代理人 韩建伟 张永明

(51)Int.Cl.

H04L 12/24(2006.01)

(56)对比文件

CN 103457959 A,2013.12.18,
CN 103457959 A,2013.12.18,
CN 101178665 A,2008.05.14,
CN 101247401 A,2008.08.20,
US 2012/0147733 A1,2012.06.14,

审查员 穆剑

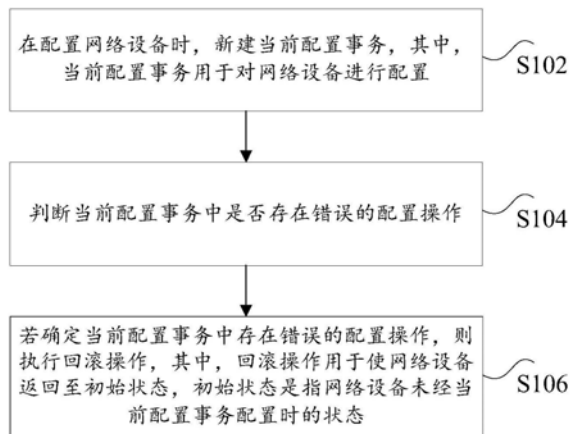
权利要求书2页 说明书8页 附图2页

(54)发明名称

配置事务的处理方法及装置

(57)摘要

本发明公开了一种配置事务的处理方法及装置。其中,该方法包括:在配置网络设备时,新建当前配置事务,其中,当前配置事务用于对网络设备进行配置;判断当前配置事务中是否存在错误的配置操作;若确定当前配置事务中存在错误的配置操作,则执行回滚操作,其中,回滚操作用于使网络设备返回至初始状态,初始状态是指网络设备未经当前配置事务配置时的状态。本发明解决了现有技术中网络设备配置事务处理成本较高的技术问题。



1. 一种配置事务的处理方法,其特征在于,包括:

在配置网络设备时,新建当前配置事务,其中,所述当前配置事务用于对所述网络设备进行配置;

判断所述当前配置事务中是否存在错误的配置操作;

若确定所述当前配置事务中存在所述错误的配置操作,则执行回滚操作,其中,所述回滚操作用于使所述网络设备返回至初始状态,所述初始状态是指所述网络设备未经所述当前配置事务配置时的状态,

其中,所述在配置网络设备时,所述方法还包括:通过事务日志记录所述当前配置事务中的每一个配置操作之间的配置关系,其中,所述配置关系至少包括从属关系、顺序关系和依赖关系,

执行回滚操作包括:

若当前配置事务用于对操作对象进行增加操作,则其对应的回滚操作为:操作对象不存在,仅记录当前配置数据,配置类型合并为删除操作;

若当前配置事务用于对操作对象进行删除操作,则其对应的回滚操作为:记录被删除的操作对象及其子对象,并将记录的数据合并为设置当前的操作对象及其子对象;

若当前配置事务用于对操作对象进行设置操作,则其对应的回滚操作为:若配置操作恢复为默认值,相当于增加操作;若配置操作恢复为除默认值之外的其他数值时,需要记录前值并将记录的数值合并为设置前值。

2. 根据权利要求1所述的方法,其特征在于,在判断所述当前配置事务中是否存在错误的配置操作之后,所述方法还包括:

若确定所述当前配置事务中不存在所述错误的配置操作,则根据所述当前配置事务配置所述网络设备;

在完成所述配置所述网络设备之后,释放所述当前配置事务所占用的所述网络设备的资源。

3. 根据权利要求1所述的方法,其特征在于,在所述新建当前配置事务之前,所述方法还包括:

对所述当前配置事务中的每一个配置操作和所述每一个配置操作所对应的配置数据接口进行抽象化处理。

4. 根据权利要求1所述的方法,其特征在于,所述在配置网络设备时,所述方法还包括:通过所述事务日志记录所述当前配置事务中正确的配置操作。

5. 根据权利要求1所述的方法,其特征在于,在所述新建当前配置事务之后,所述方法还包括:

若所述当前配置事务中存在被锁定的配置数据,则不允许新建配置事务;

若所述当前配置事务未结束配置所述网络设备,则不允许新建所述配置事务。

6. 根据权利要求1所述的方法,其特征在于,所述在配置网络设备时,所述方法还包括:若在预设时间段内未接收到根据所述当前配置事务而生成的网络响应,则结束配置所述网络设备。

7. 一种配置事务的处理装置,其特征在于,包括:

新建单元,用于在配置网络设备时,新建当前配置事务,其中,所述当前配置事务用于

对所述网络设备进行配置；

判断单元,用于判断所述当前配置事务中是否存在错误的配置操作；

第一处理单元,用于若确定所述当前配置事务中存在所述错误的配置操作,则执行回滚操作,其中,所述回滚操作用于使所述网络设备返回至初始状态,所述初始状态是指所述网络设备未经所述当前配置事务配置时的状态,

其中,所述在配置网络设备时,所述配置事务的处理装置还用于通过事务日志记录所述当前配置事务中的每一个配置操作之间的配置关系,其中,所述配置关系至少包括从属关系、顺序关系和依赖关系,

所述第一处理单元还用于:

若当前配置事务用于对操作对象进行增加操作,则其对应的回滚操作为:操作对象不存在,仅记录当前配置数据,配置类型合并为删除操作;

若当前配置事务用于对操作对象进行删除操作,则其对应的回滚操作为:记录被删除的操作对象及其子对象,并将记录的数据合并为设置当前的操作对象及其子对象;

若当前配置事务用于对操作对象进行设置操作,则其对应的回滚操作为:若配置操作恢复为默认值,相当于增加操作;若配置操作恢复为除默认值之外的其他数值时,需要记录前值并将记录的数值合并为设置前值。

8. 根据权利要求7所述的装置,其特征在于,所述装置还包括:

配置单元,用于若确定所述当前配置事务中不存在所述错误的配置操作,则根据所述当前配置事务配置所述网络设备;

释放单元,用于在完成所述配置所述网络设备之后,释放所述当前配置事务所占用的所述网络设备的资源。

9. 根据权利要求7所述的装置,其特征在于,所述装置还包括:

第二处理单元,用于对所述当前配置事务中的每一个配置操作和所述每一个配置操作所对应的配置数据接口进行抽象化处理。

配置事务的处理方法及装置

技术领域

[0001] 本发明涉及计算机领域,具体而言,涉及一种配置事务的处理方法及装置。

背景技术

[0002] FCAPS (Fault/Configuration/Accounting/Performance and Security,错误/配置/计算/性能和安全性)描述了网络管理系统所处理的五种不同的信息,FCAPS是网络管理的五种基本功能的缩写,也是网络管理工作目标的五个标准,其中,配置管理是网络管理中最常用的一项功能。

[0003] 目前,进行配置管理一般需要通过命令行接口(Command Line Interface,简称为CLI)或图形界面接口(Website User Interface,简称为Web UI)来实现,上述二者均为计算机操作系统或应用程序为用户提供的可视提示接口,但仍各有特点。具体地,命令行接口操作简单,每次可以配置一个或者相关联的多个参数,配置实时生效,如若在执行过程中发现错误,则配置不会生效;而图形界面接口相较于命令行接口,其界面更为友好,可用性更强,每次可以对一组下发参数进行配置。

[0004] 通过图形界面接口进行网络设备配置时,界面上通常会产生一批同时下发的参数,这批参数可以映射为一组命令行接口参数,进而通过由这些参数所构成的命令行,可以对网络设备配置数据库进行一系列的读、写操作。但在执行过程中,上述一条或者多条命令行可能会执行失败,此时,网络设备的配置会处于一个未知的中间状态,修复这个中间状态需要花费较长的时间,而这段时间很可能导致网络中断,或引发网络安全漏洞。此外,若存在多个管理员并发进行网络设备配置行为,则相互之间可能存在干扰。

[0005] 目前,为解决上述问题,现有技术中一般通过以下两种方式:

[0006] 方式一、保障网络设备配置过程中的隔离性,即引入事务(Transaction)的概念,将事务中的隔离性应用于网络设备配置,从而解决多个管理员并发进行网络设备配置时所存在的相互干扰的问题。

[0007] 方式二、通过二阶段提交(Two-phase Commit)方法查找网络设备在配置过程中出现的命令行执行失败的原因,但该方法需要对网络管理系统进行改造,其难度和工作量较大,因而导致成本较高。

[0008] 综上,现有技术中存在网络设备配置事务处理成本较高的技术问题。

[0009] 针对上述的问题,目前尚未提出有效的解决方案。

发明内容

[0010] 本发明实施例提供了一种配置事务的处理方法及装置,以至少解决现有技术中网络设备配置事务处理成本较高的技术问题。

[0011] 根据本发明实施例的一个方面,提供了一种配置事务的处理方法,该处理方法包括:在配置网络设备时,新建当前配置事务,其中,所述当前配置事务用于对所述网络设备进行配置;判断所述当前配置事务中是否存在错误的配置操作;若确定所述当前配置事务

中存在所述错误的配置操作,则执行回滚操作,其中,所述回滚操作用于使所述网络设备返回至初始状态,所述初始状态是指所述网络设备未经所述当前配置事务配置时的状态。

[0012] 进一步地,在判断所述当前配置事务中是否存在错误的配置操作之后,所述方法还包括:若确定所述当前配置事务中不存在所述错误的配置操作,则根据所述当前配置事务配置所述网络设备;在完成所述配置所述网络设备之后,释放所述当前配置事务所占用的所述网络设备的资源。

[0013] 进一步地,在所述新建当前配置事务之前,所述方法还包括:对所述当前配置事务中的每一个配置操作和所述每一个配置操作所对应的配置数据接口进行抽象化处理。

[0014] 进一步地,所述在配置网络设备时,所述方法还包括:通过事务日志记录所述当前配置事务中的每一个配置操作之间的配置关系,其中,所述配置关系至少包括从属关系、顺序关系和依赖关系。

[0015] 进一步地,所述在配置网络设备时,所述方法还包括:通过所述事务日志记录所述当前配置事务中正确的配置操作。

[0016] 进一步地,在所述新建当前配置事务之后,所述方法还包括:若所述当前配置事务中存在被锁定的配置数据,则不允许新建配置事务;若所述当前配置事务未结束配置所述网络设备,则不允许新建所述配置事务。

[0017] 进一步地,所述在配置网络设备时,所述方法还包括:若在预设时间段内未接收到根据所述当前配置事务而生成的网络响应,则结束配置所述网络设备。

[0018] 根据本发明实施例的另一个方面,还提供了一种配置事务的处理装置,该处理装置包括:新建单元,用于在配置网络设备时,新建当前配置事务,其中,所述当前配置事务用于对所述网络设备进行配置;判断单元,用于判断所述当前配置事务中是否存在错误的配置操作;第一处理单元,用于若确定所述当前配置事务中存在所述错误的配置操作,则执行回滚操作,其中,所述回滚操作用于使所述网络设备返回至初始状态,所述初始状态是指所述网络设备未经所述当前配置事务配置时的状态。

[0019] 进一步地,所述装置还包括:配置单元,用于若确定所述当前配置事务中不存在所述错误的配置操作,则根据所述当前配置事务配置所述网络设备;释放单元,用于在完成所述配置所述网络设备之后,释放所述当前配置事务所占用的所述网络设备的资源。

[0020] 进一步地,所述装置还包括:第二处理单元,用于对所述当前配置事务中的每一个配置操作和所述每一个配置操作所对应的配置数据接口进行抽象化处理。

[0021] 在本发明实施例中,采用在配置网络设备时,新建当前配置事务,其中,当前配置事务用于对网络设备进行配置;判断当前配置事务中是否存在错误的配置操作;若确定当前配置事务中存在错误的配置操作,则执行回滚操作,其中,回滚操作用于使网络设备返回至初始状态,初始状态是指网络设备未经当前配置事务配置时的状态的方式,通过在配置网络设备时新建当前配置事务,并判断当前配置事务中是否存在错误的配置操作,达到了若确定当前配置事务中存在错误的配置操作,则执行回滚操作的目的,从而实现了准确、完整、快速地发现并纠正网络设备配置错误,以及减少纠错成本的技术效果,进而解决了现有技术中网络设备配置事务处理成本较高的技术问题。

附图说明

[0022] 此处所说明的附图用来提供对本发明的进一步理解,构成本申请的一部分,本发明的示意性实施例及其说明用于解释本发明,并不构成对本发明的不当限定。在附图中:

[0023] 图1是根据本发明实施例的一种可选的配置事务的处理方法的流程示意图;

[0024] 图2是根据本发明实施例的另一种可选的配置事务的处理方法的流程示意图;

[0025] 图3是根据本发明实施例的一种可选的配置事务的处理装置的结构示意图;

[0026] 图4是根据本发明实施例的另一种可选的配置事务的处理装置的结构示意图;

[0027] 图5是根据本发明实施例的又一种可选的配置事务的处理装置的结构示意图。

具体实施方式

[0028] 为了使本技术领域的人员更好地理解本发明方案,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的实施例,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,都应当属于本发明保护的范围。

[0029] 需要说明的是,本发明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,以便这里描述的本发明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。

[0030] 实施例1

[0031] 根据本发明实施例,提供了一种配置事务的处理方法的实施例,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,并且在,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。

[0032] 图1是根据本发明实施例的一种可选的配置事务的处理方法的流程示意图,如图1所示,该方法包括如下步骤:

[0033] 步骤S102,在配置网络设备时,新建当前配置事务,其中,当前配置事务用于对网络设备进行配置;

[0034] 步骤S104,判断当前配置事务中是否存在错误的配置操作;

[0035] 步骤S106,若确定当前配置事务中存在错误的配置操作,则执行回滚操作,其中,回滚操作用于使网络设备返回至初始状态,初始状态是指网络设备未经当前配置事务配置时的状态。

[0036] 在本发明实施例中,通过在配置网络设备时新建当前配置事务,并判断当前配置事务中是否存在错误的配置操作,达到了若确定当前配置事务中存在错误的配置操作,则执行回滚操作的目的,从而实现了准确、完整、快速地发现并纠正网络设备配置错误,以及减少纠错成本的技术效果,进而解决了现有技术中网络设备配置事务处理成本较高的技术

问题。

[0037] 其中,当前配置事务是指当前正在执行的具备配置功能的事务,事务是访问并可能更新设备配置数据项的一个程序执行单元,也可以定义为一个由有限的配置操作序列构成的逻辑单元。在本申请中,当前配置事务主要为实现以下两个主要目标:

[0038] 目标一、使设备配置操作序列提供从错误状态(也可称为异常状态)恢复至正确状态(也可称为正常状态),以及使设备配置数据在错误状态(也可称为异常状态)仍能够保持一致性;

[0039] 目标二、当多个应用程序并发配置设备时,使应用程序之间相互隔离,从而防止多个应用程序彼此干扰。

[0040] 为实现上述目标,可以引入事务的属性来阐述当前配置事务的合理性。通常,事务具备以下四个属性,分别为原子性(Atomicity)、一致性(Consistency)、隔离性(Isolation)和持久性(Durability)。

[0041] 其中,原子性是指事务是一个不可分割的工作单位,事务中的操作要么全部发生,要么都不发生;一致性是指事务必须使设备配置数据从一个一致性状态切换到另一个一致性状态;隔离性是指多个用户并发更改设备配置时,系统为每一个用户开启的事务,不能被其他事务的操作数据所干扰,即多个并发事务之间要相互隔离;持久性是指一个事务一旦被提交,它对设备配置数据的改变就是永久性的,即便设备发生故障也不应对设备配置数据产生任何影响。

[0042] 可选地,当前配置事务可以由一系列配置操作组成,若该系列配置操作中不存在一条或多条配置操作执行失败的情况,则可以将该配置事务视为错误的配置事务。此外,当前配置事务可以用于对网络设备至少进行以下一种或几种配置操作:增加操作对象(ADD)、删除操作对象(DELETE)、设置操作对象(SET)、取消设置操作对象(UNSET)、调整操作对象(MOVE)以及对操作对象重命名或更改其key值(RENAME)。配置数据(即操作)的数据类型可以至少包括以下一种或几种:char、short、int和long。

[0043] 可选地,回滚(Rollback)是指程序或者数据发生错误时,将程序或者数据恢复至上一次正确状态的行为,通常,回滚包括程序回滚和数据回滚等类型。

[0044] 可选地,在判断当前配置事务中是否存在错误的配置操作之后,该方法还包括:

[0045] 步骤S12,若确定当前配置事务中不存在错误的配置操作,则根据当前配置事务配置网络设备;

[0046] 步骤S14,在完成配置网络设备之后,释放当前配置事务所占用的网络设备的资源。

[0047] 具体地,若当前配置事务中不存在错误的配置操作,则无需对当前配置事务执行回滚操作,在根据当前配置事务完成配置网络设备之后,通过释放当前配置事务所占用的网络设备的资源,可以为执行下一个事务(可以为配置事务,也可以为错误、计算、性能和安全等其他事务)提供处理资源,避免冗余。

[0048] 可选地,在新建当前配置事务之前,该方法还包括:

[0049] 步骤S22,对当前配置事务中的每一个配置操作和每一个配置操作所对应的配置数据接口进行抽象化处理。

[0050] 其中,对当前配置事务中的每一个配置操作进行抽象化处理,可以使每个配置操

作都可逆。

[0051] 可选地,在配置网络设备时,该方法还包括:

[0052] 步骤S32,通过事务日志记录当前配置事务中的每一个配置操作之间的配置关系,其中,配置关系至少包括从属关系、顺序关系和依赖关系。

[0053] 具体地,执行回滚操作可能需要利用错误的配置操作,也可能需要利用正确的配置操作。若存在某个回滚操作需要利用当前配置事务之前的某个事务的配置数据(描述配置操作的属性信息或其他信息),则需要记录当前配置事务之前的该事务的配置数据,该配置数据可以被称为前值数据。

[0054] 可选地,配置操作之间具备从属关系、顺序关系或依赖关系,例如,某个当前配置事务中的存在配置操作分别为A、B、C、D和E,其中,配置操作A和配置操作B之间的顺序关系为A→B,则表示先以配置操作A配置网络设备,在配置操作A配置完成之后以配置操作B配置网络设备。再例如,若配置操作C由两个子操作D和E构成,则配置操作C、D、E之间具备从属关系。需要说明的是,配置操作的配置关系是在执行当前配置事务的过程中逐步建立起来的,在执行回滚操作时,仅需依据该配置关系逆向执行即可,例如,配置操作A和配置操作B之间的顺序关系为A→B,则回滚操作的执行顺序为B→A。也就是说,为了确保配置操作之间的配置关系(也可称为依赖关系)的正确性,从当前配置事务的尾部开始执行回滚操作,从状态切换的角度来看,若之前的配置操作之间的配置关系是正确的,则按步骤切换回去理应是正确的。

[0055] 具体地,若当前配置事务用于对操作对象进行增加(ADD)操作,则其对应的回滚操作为:操作对象不存在,仅记录当前配置数据,配置类型合并为DELETE;

[0056] 若当前配置事务用于对操作对象进行删除(DELETE)操作,则其对应的回滚操作为:记录被删除的操作对象及其子对象,并将记录的数据合并为SET当前的操作对象及其子对象;

[0057] 若当前配置事务用于对操作对象进行设置(SET)操作,则其对应的回滚操作为:若配置操作恢复为默认值,相当于ADD;若配置操作为除默认值之外的其他数值时,需要记录前值并将记录的数值合并为SET前值;

[0058] 若当前配置事务用于对操作对象进行取消设置(UNSET)操作,则其对应的回滚操作为:若配置操作恢复为默认值,则记录恢复前的数值而无需记录当前配置,将记录的数值合并为SET前值;

[0059] 若当前配置事务用于对操作对象进行调整(MOVE)操作,则其对应的回滚操作为:记录配置操作的相对移动位置;

[0060] 若当前配置事务用于对操作对象进行重命名(RENAME)操作,则其对应的回滚操作为:记录配置操作名称变更前的名称。

[0061] 可选地,在配置网络设备时,方法还包括:

[0062] 步骤S42,通过事务日志记录当前配置事务中正确的配置操作,其中,正确的配置操作对于执行回滚操作具备极重要的支持作用。

[0063] 可选地,在新建当前配置事务之后,该方法还包括:

[0064] 步骤S52,若当前配置事务中存在被锁定的配置数据,则不允许新建配置事务;

[0065] 步骤S54,若所述当前配置事务未结束配置所述网络设备,则不允许新建所述配置

事务。

[0066] 可选地,若存在当前配置事务,则不允许除当前配置事务之外的配置事务对网络设备进行配置,即不能同时新建两个配置事务,也不能同时存在两个配置事务。此外,在根据当前配置事务配置网络设备时,不允许其他操作对当前配置事务中的配置数据进行锁定或解锁。当前配置事务理应被视为一个整体状态,不能被中断。

[0067] 可选地,在配置网络设备时,该方法还包括:

[0068] 步骤S62,若在预设时间段内未接收到根据当前配置事务而生成的网络响应,则结束配置网络设备。

[0069] 具体地,若当前配置事务的事务发起者(例如某个Web网页或某个程序)在预设时间段内未反馈因配置网络设备而产生的网络响应,则可以推测该事务发起者已单方面离开,并处于死锁(Dead lock)状态。此时立即结束配置网络设备,可以防止网络中断,避免引发网络安全上的漏洞。

[0070] 可选地,图2是根据本发明实施例的另一种可选的配置事务的处理方法的流程示意图,如图2所示,该方法包括如下步骤:

[0071] 步骤S202,进入空闲状态,例如,该空闲状态可以为Idle状态。

[0072] 步骤S204,执行当前事务,即执行当前配置事务。

[0073] 在执行完成步骤S204之后,可以执行步骤S206、步骤S208或步骤S210。

[0074] 步骤S206,执行超时操作,其过程为若在预设时间段内未接收到网络响应,则结束配置网络设备。

[0075] 步骤S208,执行回滚操作,其目的在于使网络设备返回至初始状态。

[0076] 在执行完成步骤S206或步骤S208之后,可执行步骤S210。

[0077] 步骤S210,完成当前事务,即已根据当前配置事务完成对网络设备的配置。

[0078] 在本发明实施例中,通过在配置网络设备时新建当前配置事务,并判断当前配置事务中是否存在错误的配置操作,达到了若确定当前配置事务中存在错误的配置操作,则执行回滚操作的目的,从而实现了准确、完整、快速地发现并纠正网络设备配置错误,以及减少纠错成本的技术效果,进而解决了现有技术中网络设备配置事务处理成本较高的技术问题。

[0079] 实施例2

[0080] 根据本发明实施例的另一方面,还提供了一种配置事务的处理装置,如图3所示,该装置包括:新建单元301、判断单元303、第一处理单元305。

[0081] 其中,新建单元301,用于在配置网络设备时,新建当前配置事务,其中,所述当前配置事务用于对所述网络设备进行配置;

[0082] 判断单元303,用于判断所述当前配置事务中是否存在错误的配置操作;

[0083] 第一处理单元305,用于若确定所述当前配置事务中存在所述错误的配置操作,则执行回滚操作,其中,所述回滚操作用于使所述网络设备返回至初始状态,所述初始状态是指所述网络设备未经所述当前配置事务配置时的状态。

[0084] 其中,当前配置事务是指当前正在执行的具备配置功能的事务,事务是访问并可能更新设备配置数据项的一个程序执行单元,也可以定义为一个由有限的配置操作序列构成的逻辑单元。在本申请中,当前配置事务主要为实现以下两个主要目标:

[0085] 目标一、使设备配置操作序列提供从错误状态(也可称为异常状态)恢复至正确状态(也可称为正常状态),以及使设备配置数据在错误状态(也可称为异常状态)仍能够保持一致性;

[0086] 目标二、当多个应用程序并发配置设备时,使应用程序之间相互隔离,从而防止多个应用程序彼此干扰。

[0087] 为实现上述目标,可以引入事务的属性来阐述当前配置事务的合理性。通常,事务具备以下四个属性,分别为原子性(Atomicity)、一致性(Consistency)、隔离性(Isolation)和持久性(Durability)。

[0088] 其中,原子性是指事务是一个不可分割的工作单位,事务中的操作要么全部发生,要么都不发生;一致性是指事务必须使设备配置数据从一个一致性状态切换到另一个一致性状态;隔离性是指多个用户并发更改设备配置时,系统为每一个用户开启的事务,不能被其他事务的操作数据所干扰,即多个并发事务之间要相互隔离;持久性是指一个事务一旦被提交,它对设备配置数据的改变就是永久性的,即便设备发生故障也不应对设备配置数据产生任何影响。

[0089] 可选地,当前配置事务可以由一系列配置操作组成,若该系列配置操作中存在一条或多条配置操作执行失败的情况,则可以将该配置事务视为错误的配置事务。此外,当前配置事务可以用于对网络设备至少进行以下一种或几种配置操作:增加操作对象(ADD)、删除操作对象(DELETE)、设置操作对象(SET)、取消设置操作对象(UNSET)、调整操作对象(MOVE)以及对操作对象重命名或更改其key值(RENAME)。配置数据(即操作)的数据类型可以至少包括以下一种或几种:char、short、int和long。

[0090] 可选地,回滚(Rollback)是指程序或者数据发生错误时,将程序或者数据恢复至上一次正确状态的行为,通常,回滚包括程序回滚和数据回滚等类型。

[0091] 可选地,如图4所示,该装置还包括:配置单元401、释放单元403。

[0092] 其中,配置单元401,用于若确定所述当前配置事务中不存在所述错误的配置操作,则根据所述当前配置事务配置所述网络设备;

[0093] 释放单元403,用于在完成所述配置所述网络设备之后,释放所述当前配置事务所占用的所述网络设备的资源。

[0094] 具体地,若当前配置事务中不存在错误的配置操作,则无需对当前配置事务执行回滚操作,在根据当前配置事务完成配置网络设备之后,通过释放当前配置事务所占用的网络设备的资源,可以为执行下一个事务(可以为配置事务,也可以为错误、计算、性能和安全等其他事务)提供处理资源,避免冗余。

[0095] 可选地,如图5所示,该装置还包括:第二处理单元501。

[0096] 其中,第二处理单元501,用于对所述当前配置事务中的每一个配置操作和所述每一个配置操作所对应的配置数据接口进行抽象化处理。

[0097] 其中,第二处理单元501对当前配置事务中的每一个配置操作进行抽象化处理,可以使每个配置操作都可逆。

[0098] 在本发明实施例中,通过在配置网络设备时新建当前配置事务,并判断当前配置事务中是否存在错误的配置操作,达到了若确定当前配置事务中存在错误的配置操作,则执行回滚操作的目的,从而实现了准确、完整、快速地发现并纠正网络设备配置错误,以及

减少纠错成本的技术效果,进而解决了现有技术中网络设备配置事务处理成本较高的技术问题。

[0099] 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。

[0100] 在本发明的上述实施例中,对各个实施例的描述都各有侧重,某个实施例中沒有详述的部分,可以参见其他实施例的相关描述。

[0101] 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,可以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。

[0102] 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。

[0103] 另外,在本发明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。

[0104] 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)执行本发明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。

[0105] 以上所述仅是本发明的优选实施方式,应当指出,对于本技术领域的普通技术人员来说,在不脱离本发明原理的前提下,还可以做出若干改进和润饰,这些改进和润饰也应视为本发明的保护范围。

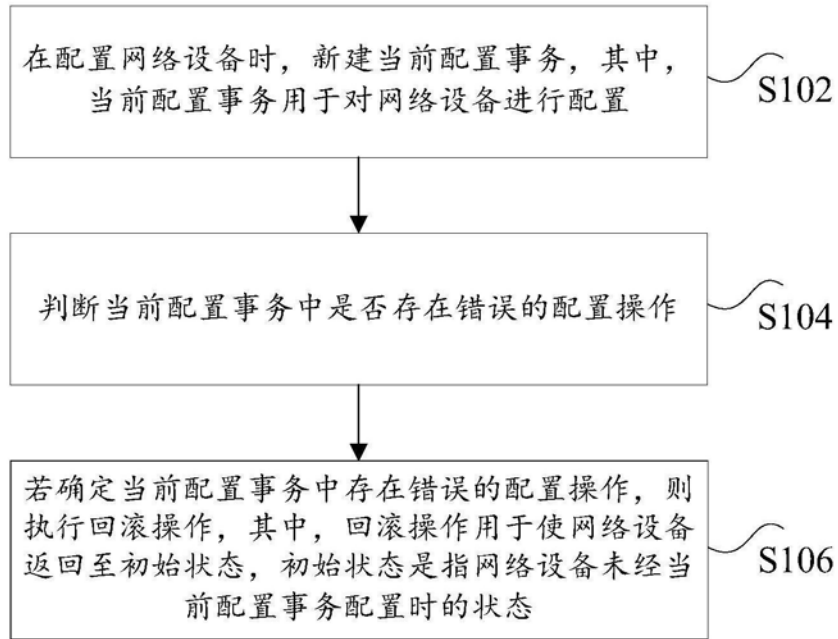


图1

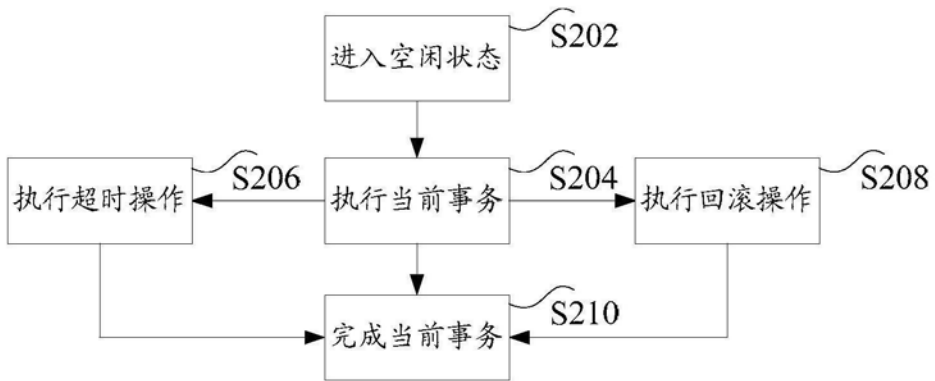


图2



图3

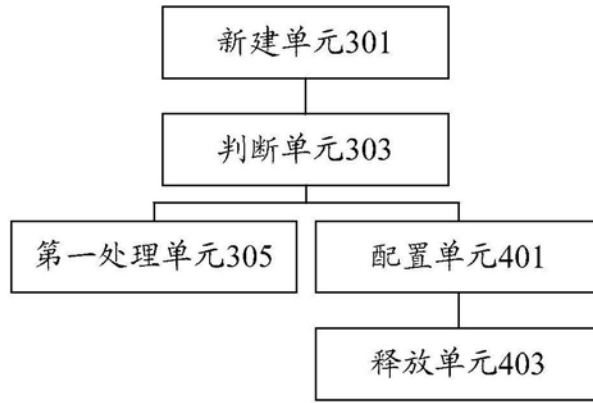


图4



图5