(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 104468302 A
(43)申请公布日 2015.03.25

(21)申请号 201410550612.8

(22)申请日 2014.10.16

(71)申请人 深圳市金证科技股份有限公司
地址 518057 广东省深圳市南山区科技园高新区南区高新南五道金证科技大楼8-9层

(72)发明人 何万刚

(74)专利代理机构 深圳中一专利商标事务所 44237
代理人 张全文

(51)Int.Cl.
H04L 12/433(2006.01)
H04L 12/703(2013.01)
H04L 12/757(2013.01)

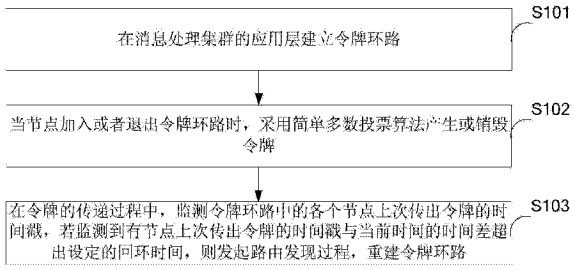
权利要求书2页 说明书10页 附图4页

(54)发明名称

一种令牌的处理方法、装置及消息处理集群

(57)摘要

本发明适用于网络技术领域，提供了一种令牌的处理方法、装置及消息处理集群，所述方法包括：在消息处理集群的应用层建立令牌环路；当节点加入或者退出令牌环路时，采用简单多数投票算法产生或销毁令牌；在令牌的传递过程中，监测令牌环路中的各个节点上次传出令牌的时间戳，若监测到有节点上次传出令牌的时间戳与当前时间的时间差超出设定的回环时间，则发起路由发现过程，重建令牌环路。本发明，避免了消息处理集群中出现2个或者2个以上的令牌环路；另外，对各个节点的令牌回环时间进行监视，解决了现有技术，令牌环路中令牌的丢失无法监管等难题的同时，大大提高了分布式集群系统的可靠性。



CN 104468302 A

1. 一种令牌的处理方法，其特征在于，所述方法包括：

在消息处理集群的应用层建立令牌环路；

当节点加入或者退出令牌环路时，采用简单多数投票算法产生或销毁令牌；

在令牌的传递过程中，监测令牌环路中的各个节点上次传出令牌的时间戳，若监测到有节点上次传出令牌的时间戳与当前时间的时间差超出设定的回环时间，则发起路由发现过程，重建令牌环路。

2. 如权利要求 1 所述的方法，其特征在于，所述简单多数投票算法产生或销毁令牌包括：

预设消息处理集群中的节点数量为 N，N 为大于等于 1 的整数；

当监测到有节点加入由消息处理集群中的 N 个节点组成的令牌环路或者退出所述令牌环路时，动态计算所述令牌环路内的节点数量 P；

当 P 大于 (N+2)/2 舍尾取整值时，若令牌环路内无令牌，则自动产生新的令牌；

若令牌环路内有令牌，则维持原令牌；

当 P 小于等于 (N+2)/2 舍尾取整值时，若令牌环路内有令牌，则销毁该令牌。

3. 如权利要求 1 或 2 所述的方法，其特征在于，在所述当 P 小于等于 (N+2)/2 舍尾取整值时，还包括：

若令牌环路内无令牌，则系统挂起，等待新的节点加入所述消息处理集群中。

4. 如权利要求 1 或 2 所述的方法，其特征在于，所述消息处理集群中的节点包括投票站节点、普通站节点和监控站节点。

5. 一种令牌的处理装置，其特征在于，所述装置包括：

令牌环路建立单元，用于在消息处理集群的应用层建立令牌环路；

令牌产生或者销毁单元，用于当节点加入或者退出令牌环路时，采用简单多数投票算法产生或销毁令牌；

令牌回环时间监测单元，用于在令牌的传递过程中，监测令牌环路中的各个节点上次传出令牌的时间戳，若监测到有节点上次传出令牌的时间戳与当前时间的时间差超出设定的回环时间，则发起路由发现过程，重建令牌环路。

6. 如权利要求 5 所述的装置，其特征在于，所述令牌产生或者销毁单元包括：

节点数量预设模块，用于预设消息处理集群中的节点数量为 N，N 为大于等于 1 的整数；

节点数量统计模块，用于当监测到有节点加入由消息处理集群中的 N 个节点组成的令牌环路或者退出所述令牌环路时，动态计算所述令牌环路内的节点数量 P；

令牌产生模块，用于当 P 大于 (N+2)/2 舍尾取整值时，若令牌环路内无令牌，则自动产生新的令牌；

令牌维持模块，用于若令牌环路内有令牌，则维持原令牌；

令牌销毁模块，用于当 P 小于等于 (N+2)/2 舍尾取整值时，若令牌环路内有令牌，则销毁该令牌。

7. 如权利要求 5 或 6 所述的装置，其特征在于，所述令牌产生或者销毁单元还包括：

节点加入等待模块，用于若令牌环路内无令牌，则系统挂起，等待新的节点加入所述消息处理集群中。

8. 如权利要求5或6所述的装置，其特征在于，所述装置内置于消息处理集群中的相应节点中，所述消息处理集群中的节点包括投票站节点、普通站节点和监控站节点。

9. 一种消息处理集群，其特征在于，所述消息处理集群中包括投票站节点、普通站节点和监控站节点，所述消息处理集群中的相应节点中内置有如权利要求5至8任一项所述的令牌的处理装置。

一种令牌的处理方法、装置及消息处理集群

技术领域

[0001] 本发明属于网络技术领域，尤其涉及一种令牌的处理方法、装置及消息处理集群。

背景技术

[0002] 随着网络规模的不断扩展，分布式系统被广泛应用，同时也带来了分布式协商一致性的问题。一致性问题主要表现为数据一致性、逻辑一致性和时序一致性，解决一致性问题主要是确保分布式系统在同时访问和控制一个资源时，能够协调一致，保证访问的正确性（无死锁和无饥饿现象）。互斥是分布式系统解决一致性问题的关键。通常分布式系统使用的互斥算法有：集中式算法、分布式算法和令牌环算法。

[0003] 集中式算法选择一个进程作为协调器，用于协调临界区的进入。

[0004] 特点：协调器在同一时间只允许一个进程进入临界区，故能保证互斥。因为请求消息是顺序排队得，不会出现“饿死”现象。但是单一的协调器是其瓶颈。

[0005] 分布式算法的思想是：对想进入临界区的进程首先建立一个消息，该消息包括待进入的临界区名、进程名和时间戳。一个进程接收到消息后，会有如下操作：如果不在临界区内且不想进入临界区则回复一个消息；如果在临界区内则不回复消息，将请求消息放入队列，如果不在临界区，但想进入临界区，则比较想进入临界区的进程的时间戳，时间戳小的进程进入临界区。

[0006] 特点：算法复杂，易出现“饿死”现象，且系统不健壮；但它从理论上表明了算法的可行性，必将发展出实际可行的算法。

[0007] 令牌环算法的思想是：整个系统只有一块令牌，只有令牌持有者才具有进入临界区的资格。当进程 i 从进程 i-1 接到令牌时，它检查是否想进入临界区，如果是，则进入，待其推出后，将令牌传递给进程 i+1；如果不想进入，则直接把令牌向下传递。

[0008] 特点：一是令牌易丢失，事实上检测令牌丢失是很困难的；二是容易出现进程故障，且进程故障比较不容易恢复。

发明内容

[0009] 本发明实施例提供了一种令牌的处理方法、装置及消息处理集群，旨在解决现有技术提供的互斥算法，令牌丢失难以监控、进程崩溃难以恢复的问题。

[0010] 一方面，提供一种令牌的处理方法，所述方法包括：

[0011] 在消息处理集群的应用层建立令牌环路；

[0012] 当节点加入或者退出令牌环路时，采用简单多数投票算法产生或销毁令牌；

[0013] 在令牌的传递过程中，监测令牌环路中的各个节点上次传出令牌的时间戳，若监测到有节点上次传出令牌的时间戳与当前时间的时间差超出设定的回环时间，则发起路由发现过程，重建令牌环路。

[0014] 进一步地，所述简单多数投票算法产生或销毁令牌包括：

[0015] 预设消息处理集群中的节点数量为 N，N 为大于等于 1 的整数；

[0016] 当监测到有节点加入由消息处理集群中的 N 个节点组成的令牌环路或者退出所述令牌环路时，动态计算所述令牌环路内的节点数量 P ；

[0017] 当 P 大于 (N+2)/2 舍尾取整值时，若令牌环路内无令牌，则自动产生新的令牌 ；

[0018] 若令牌环路内有令牌，则维持原令牌 ；

[0019] 当 P 小于等于 (N+2)/2 舍尾取整值时，若令牌环路内有令牌，则销毁该令牌。

[0020] 进一步地，在所述当 P 小于等于 (N+2)/2 舍尾取整值时，还包括 ：

[0021] 若令牌环路内无令牌，则系统挂起，等待新的节点加入所述消息处理集群中。

[0022] 进一步地，所述消息处理集群中的节点包括投票站节点、普通站节点和监控站节点。

[0023] 另一方面，提供一种令牌的处理装置，所述装置包括 ：

[0024] 令牌环路建立单元，用于在消息处理集群的应用层建立令牌环路 ；

[0025] 令牌产生或者销毁单元，用于当节点加入或者退出令牌环路时，采用简单多数投票算法产生或销毁令牌 ；

[0026] 令牌回环时间监测单元，用于在令牌的传递过程中，监测令牌环路中的各个节点上次传出令牌的时间戳，若监测到有节点上次传出令牌的时间戳与当前时间的时间差超出设定的回环时间，则发起路由发现过程，重建令牌环路。

[0027] 进一步地，所述令牌产生或者销毁单元包括 ：

[0028] 节点数量预设模块，用于预设消息处理集群中的节点数量为 N，N 为大于等于 1 的整数 ；

[0029] 节点数量统计模块，用于当监测到有节点加入由消息处理集群中的 N 个节点组成的令牌环路或者退出所述令牌环路时，动态计算所述令牌环路内的节点数量 P ；

[0030] 令牌产生模块，用于当 P 大于 (N+2)/2 舍尾取整值时，若令牌环路内无令牌，则自动产生新的令牌 ；

[0031] 令牌维持模块，用于若令牌环路内有令牌，则维持原令牌 ；

[0032] 令牌销毁模块，用于当 P 小于等于 (N+2)/2 舍尾取整值时，若令牌环路内有令牌，则销毁该令牌。

[0033] 进一步地，所述令牌产生或者销毁单元还包括 ：

[0034] 节点加入等待模块，用于若令牌环路内无令牌，则系统挂起，等待新的节点加入所述消息处理集群中。

[0035] 进一步地，所述装置内置于消息处理集群中的相应节点中，所述消息处理集群中的节点包括投票站节点、普通站节点和监控站节点。

[0036] 再一方面，提供一种消息处理集群，所述消息处理集群中包括投票站节点、普通站节点和监控站节点，所述消息处理集群中的相应节点中内置有如上所述的令牌的处理装置。

[0037] 在本发明实施例，在消息处理集群的应用层建立令牌环路后，当节点加入或者退出该令牌环路时，采用简单多数投票算法产生或销毁令牌，避免了消息处理集群中出现 2 个或者 2 个以上的令牌环路 ；另外，对各个节点的令牌回环时间进行监视，以及时发现令牌是否丢失，并在令牌丢失时，重建令牌环路，解决了现有技术，令牌环路中令牌的丢失无法监管等难题的同时，大大提高了分布式集群系统的可靠性。

附图说明

[0038] 图 1 是本发明实施例一提供的令牌的处理方法的实现流程图；

[0039] 图 2 是本发明实施例一提供的路由发现过程的实现流程图；

[0040] 图 3 是本发明实施例一提供的路由同步过程的实现流程图；

[0041] 图 4 是本发明实施例一提供的令牌检查过程的实现流程图；

[0042] 图 5 是本发明实施例一提供的令牌产生过程的实现流程图；

[0043] 图 6 是本发明实施例一提供的令牌传递过程的实现流程图；

[0044] 图 7 是本发明实施例二提供的令牌的处理装置的结构框图。

具体实施方式

[0045] 为了使本发明的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本发明进行进一步详细说明。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。

[0046] 在本发明实施例中，在消息处理集群的应用层建立令牌环路；当节点加入或者退出令牌环路时，采用简单多数投票算法产生或销毁令牌；在令牌的传递过程中，监测令牌环路中的各个节点上次传出令牌的时间戳，若监测到有节点上次传出令牌的时间戳与当前时间的时间差超出设定的回环时间，则发起路由发现过程，重建令牌环路。

[0047] 以下结合具体实施例对本发明的实现进行详细描述：

[0048] 实施例一

[0049] 图 1 示出了本发明实施例一提供的令牌的处理方法的实现流程，详述如下：

[0050] 在步骤 S101 中，在消息处理集群的应用层建立令牌环路。

[0051] 在步骤 S102 中，当节点加入或者退出令牌环路时，采用简单多数投票算法产生或销毁令牌。

[0052] 在本发明实施例中，为了避免消息处理集群分裂为两个或者两个以上的逻辑环路进行序列化工作的“脑裂”现象，因此采用“(N+2)/2 简单多数”的投票算法自动产生与销毁令牌，简单多数投票算法的算法如下：

[0053] 预设消息处理集群中的节点数量为 N，N 为大于等于 1 的整数，当监测到有节点加入由消息处理集群中的 N 个节点组成的令牌环路或者退出所述令牌环路时，动态计算所述令牌环路内的节点数量 P，当 P 大于 (N+2)/2 舍尾取整值时，若令牌环路内无令牌，则自动产生新的令牌；若令牌环路内有令牌，则维持原令牌；当 P 小于等于 (N+2)/2 舍尾取整值时，若令牌环路内有令牌，则销毁该令牌，若令牌环路内无令牌，则系统挂起，等待新的节点加入所述消息处理集群中。

[0054] 根据以上简单多数投票算法，奇数台主机（节点）容灾能力比奇数 +1 的偶数台主机还要强，所以在实际部署时一般采用奇数台主机组成消息处理集群。对于 3 台消息处理集群可容灾 1 台机器故障，5 台消息处理集群可容 1 台故障或 2 台同时故障，以此类推，此简单多数投票算法只容少数机器节点的灾难与故障，但相对单机来说，提升了可靠性。以下分别以 3 台、5 台、7 台为例计算消息处理集群的故障率。

[0055] 前提：

[0056] 1)、所有节点主机的出现故障的概率大致相同，即故障率为 q。

[0057] 2)、假设主机在 100 天内出现 1 次故障，则计算该主机的故障率 q 为 1%，依次类推。

[0058] 3)、假设所有主机的故障率 q 不超过 10%，即 10 天内出现小于 1 次故障。

[0059] 1 台故障率 $Q(1) = q$

[0060] 3 台故障率 $Q(3) = 3q^2(1-q)+q^3 = 3q^2-2q^3$

[0061] 5 台故障率 $Q(5) = 10q^3(1-q)^2+5q^4(1-q)+q^5 = 10q^3-15q^4+6q^5$

[0062] 7 台故障率 $Q(7) = 35q^4(1-q)^3+21q^5(1-q)^2+7q^6(1-q)+q^7 = 35q^4-84q^5+70q^6-20q^7$

[0063] 3 台主从仲裁故障率 $Q(仲) = 2q^2(1-q)+q^3 = 2q^2-q^3$

[0064] 表 1 示出了消息处理集群的故障率：

[0065]

Q(1)	Q(3)	Q(5)	Q(7)	Q(仲)
0.1	0.028	0.00856	0.002728	0.018
0.01	0.000298	9.8506E-06	3.4167E-07	0.000198
0.001	0.000002998	9.98501E-09	3.49161E-11	0.000001998

[0066] 表 1

[0067] 从上述的消息处理集群故障率看，故障率下降得很快，MPU 整体可靠性可提升到 99.97%～ 99.999999%。

[0068] 其中，消息处理集群中的所有节点都有统一的规划编码，并且静态配置有站点类型。不同类型的站点产生的方式、不同类型站点的数量、不同类型站点的职能描述以及不同类型站点所参与的流程详见步骤 S102 中的表 2，由表 2 可知，令牌的产生与销毁工作是由投票站、普通站和监控站配合完成。

[0069] 在步骤 S103 中，在令牌的传递过程中，监测令牌环路中的各个节点上次传出令牌的时间戳，若监测到有节点上次传出令牌的时间戳与当前时间的时间差超出设定的回环时间，则发起路由发现过程，重建令牌环路。

[0070] 在本发明实施例中，令牌的传递依赖于在各节点之间建立唯一、稳定、高效的令牌环路。对令牌环路的管理包括节点发现、路由同步、令牌检查、令牌产生、令牌传递、回环监视等基本流程以及监控站的选举、节点动态加入、节点动态退出等组合流程。消息处理集群内的各节点按对令牌的管理职责划分为监控站、普通站、投票站。

[0071] 表 2 示出了各个节点的职能分工

[0072]

分工	产生方式	数量	职能描述	参与流程
投票站	静态配置	0~1	仅参与令牌产生时的投票	路由发现
普通站	静态配置优先级，动态选举产生	多个	参与令牌投票、令牌传递、回环监视的工作，可发起环路路由发现流程	路由发现、路由同步、令牌检查、令牌产生、令牌传递、令牌回环监视
监控站	静态配置优先级，动态选举产生	正常环路有且只有 1 个	可参与令牌管理的所有工作以及令牌性能的监视工作，可发起路由发现、路由同步、令牌检查、令牌产生流程	路由发现、路由同步、令牌检查、令牌产生、令牌传递、令牌回环监视、令牌性能监视

[0073] 表 2

[0074] 令牌环路管理算法是建立在对消息处理集群中的所有节点进行统一规划编码的基础上并静态配置站点类型。

[0075] 令牌环路的建立是在路由发现、路由同步这两个基本处理流程之上完成的，令牌环路的建立、节点的加入、节点的退出、监控站选举等均遵循路由发现、路由同步，这两个基本流程。

[0076] 表 3 示出了令牌环路的路由发现、路由同步的发起者，发起条件以及执行的内容的描述：

[0077]

流程	发起者	发起条件	内容描述	
路由发现	普通站和监控站	1) 新节点加入 2) 后继节点异常 3) 超出回环时间 4) 系统重建回环指令	指令序号	发起者产生且唯一递增
			发起者	节点编码
			生命周期	最多能传递的节点数量，每传递一个节点该值减 1
			路由信息	每传递一个节点，把该节点信息增加到路由信息的

[0078]

				结构中
路由同步	监控站	1) 路由发现正常结束且指令序号为发布最大序号 2) 系统强制路由同步指令	指令序号	发起者产生且唯一递增
			发起者	节点编码
			生命周期	最多能传递的节点数量，每传递一个节点该值减 1
			路由信息	每传递一个节点，把该节点信息增加到路由信息的结构中

[0079] 表 3

[0080] 具体的，路由发现流程如图 2 所示，包括以下步骤：

[0081] <1>、新加入者、发现节点出现异常的节点、各路由发现回环时间内未接收到过令牌以及接收到系统控制路由重建命令，则发起路由发现指令。

[0082] <2>、发起者根据集群全局配置按照依次递增从其相邻节点开始尝试连接寻找可用后继者。

[0083] <3>、若尝试连接均未成功，该节点挂起，等待其他节点加入。

[0084] <4>、如连接后继者成功，更新路由表自身。

[0085] <5>、发起者发出的路由发现指令，传递给后继者。

[0086] <6>、后继者比较路由发现指令中的发起者 ID 与自身 ID。

[0087] <7>、若发起者 ID> 自身 ID，路由发现指令停止，且发起新的路由发现指令。

[0088] <8>、若发起者 ID< 自身 ID，把路由发现指令中的生命周期减 1，若仍然大于 0，则按步骤 <2>-<8> 继续传递路由发现指令，否则终止传递。

[0089] <9>、若发起者 ID ＝自身 ID，说明路由发现指令已经传递一圈，路由指令终止，更新其路由表，该节点即为集群监控站。

[0090] 路由同步流程如图 3 所示，包括以下步骤：

[0091] <1>、路由发现指令正常结束且指令序号为当前节点发布的最大序号，由 MPU 监控站发起路由同步指令。

[0092] <2>、后继者接收到路由同步指令后，判别指令发起者 ID 与自身 ID 的关系。

[0093] <3>、若发起者 ID> 自身 ID，终止该路由同步的传输，由该节点重新发起路由同步指令。

[0094] <4>、若发起者 ID< 自身 ID，更新自身路由表与监控站信息，把路由同步指令中的生命周期减 1，若仍然大于 0 则按步骤 <2>-><5> 继续传递，否则终止传递。

[0095] <5>、若发起者 ID ＝自身 ID，说明路由同步指令已经传递一圈，指令终止。

[0096] 表 4 示出了令牌环路的维护描述：

[0097]

环路维护	发起者	描述
逻辑建环	后启动的节点	节点启动时，均发起路由发现指令，但最终以监控站发出的指令为建环有效指令。
节点加入	新加入节点	新加入节点发起路由发现指令，但最终以监控站发出的路由发现指令与路由同步指令为有效入环指令。
节点退出	该节点的前驱	令牌传递时检查后继节点是否退出。节点发现不能把令牌交给后继者时，判定该后继节点已退出环路，且由它发起路由发现指令。

[0098] 表 4

[0099] 令牌工作管理包括令牌检查、令牌产生、令牌传递、令牌回环时间监视以及令牌性能监视。

[0100] 表 5 示出了令牌工作管理流程中的交互内容：

[0101]

流程	参与者	交互内容	内容说明
令牌检查	监控站和普通站	指令序号	发起者产生且唯一递增
		发起者	节点编码
		发起时间	发起令牌检查的时间戳
		令牌标志	令牌环路是否存在令牌
		生命周期	最多能传递的节点数量，每传递一个节点该值减 1

[0102]

令牌产生	监控站	最大序号询问	请求内容：发起时间 应答内容：发起时间、最大序号
		令牌通知	请求内容：上次最大序号、本次序号
令牌传递	监控站和普通站	创建者	令牌产生的节点编码
		创建时间	令牌产生的时间戳
		开始序号	令牌产生的起始序号
		监控站	监控站点编码
		传递时延	令牌传递一圈所消耗的时间(由监控站检测)
		序列化数量	令牌传递一圈所序列化指令的数量(由监控站检测)
		最大序号	令牌传递时上次更新的最大序号

[0103] 表 5

[0104] 其中，令牌检查流程如图 4 所示，包括以下步骤：

[0105] <1>、由监控站发起令牌检查指令，并传递给其后继者。

[0106] <2>、后继者接收到令牌检查指令，判别指令发起者 ID 与自身 ID 的关系。

[0107] <3>、若发起者 ID> 自身 ID，终止该令牌检查指令的传输，由该节点重新发起令牌检查指令。

[0108] <4>、若发起者 ID< 自身 ID，根据自身运行状况更新令牌检查指令中的令牌标志，并把指令中的生命周期减 1，若仍然大于 0 则按步骤 <1>-<5>) 继续传递指令，否则终止传递。

[0109] <5>、若发起者 ID ＝自身 ID，说明检查指令已经传递一圈，指令终止。

[0110] <6>、监控站通过接收到的令牌检查指令中的令牌标志，以及发起时间与最后一次接收到令牌的时间，判别消息处理集群中是否存在令牌。

[0111] 其中，令牌产生流程如图 5 所示，包括以下步骤：

[0112] <1>、监控站发起令牌检查流程，并且判别令牌环路不存在令牌。

[0113] <2>、判断路由表中有效节点数量已超过半数。

[0114] <3>、监控站向所有投票站发起最大序号询问。

[0115] <4>、监控站接收序号询问结果，在所有应答结果中取最大序号 M，并在此基础上增加一定的基数，形成新令牌的起始序号 N ＝ M+ 所有节点数 * 最大令牌步长。

[0116] <5>、监控站产生新令牌，并发布令牌更新通知给所有投票站。

[0117] 其中，令牌传递流程如图 6 所示，包括以下步骤：

[0118] <1>、节点在接收到传递的令牌后，把本机拥有令牌的标志置为 true，同时应答返回。

[0119] <2>、节点拥有令牌时，对处于发送缓冲区内不多于令牌步长的指令进行序列化，同时组合进行多播发送。

[0120] <3>、当令牌序列的指令达到令牌步长或者发送缓冲区无待发指令，则更新令牌信息并传递给其后继者，该节点进入待命状态，并记录进入待命状态的时间戳。

[0121] <4>、令牌传递给后继者时，若出现多次传递失败后，把该后继者在本机的路由表的状态改为‘N’不可用状态，并触发一次路由发现指令，再根据更新后路由表中的可用节点数是否达到最少工作主机数量，若少于该值令牌停止传递，并发出告警。

[0122] <5>、节点在待命状态，预取业务指令并监控等待时间是否超过回环时间，若超出则停止预取业务指令进入挂起状态，同时触发一次路由发现指令。

[0123] 具体的，令牌的监视是指对令牌环路中令牌的运行状况进行监视，包括令牌回环时间监视与令牌性能监视这两部分。令牌回环时间监视是指消息处理集群中的每个节点均记录上次传出令牌的时间戳，监控站定时判别该时间戳与当前时间的时间差是否超出设定的回环时间，若超出，则需重新发起路由发现过程，重建令牌环路。

[0124] 令牌性能监视是由监控站监测令牌传递一圈的延时与吞吐量，监测结果随令牌而传递。

[0125] 通过监控站对令牌回环时间的监视，可以监测到令牌是否丢失，如果丢失，则重建令牌环路，产生新的令牌，解决了现有技术，令牌环路中令牌的丢失无法监管等难题的同时，大大提高了分布式集群系统的可靠性。

[0126] 本实施例，参考令牌环网原理，在消息处理集群的应用层建立令牌环路后，当节点加入或者退出该令牌环路时，采用简单多数投票算法产生或销毁令牌，避免了消息处理集群中出现 2 个或者 2 个以上的令牌环路；另外，对各个节点的令牌回环时间进行监视，以及时发现令牌是否丢失，并在令牌丢失时，重建令牌环路，解决了现有技术，令牌环路中令牌的丢失无法监管等难题的同时，大大提高了分布式集群系统的可靠性。

[0127] 本领域普通技术人员可以理解实现上述各实施例方法中的全部或部分步骤是可以通过程序来指令相关的硬件来完成，相应的程序可以存储于一计算机可读取存储介质中，所述的存储介质，如 ROM/RAM、磁盘或光盘等。

[0128] 实施例二

[0129] 图 7 示出了本发明实施例二提供的令牌的处理装置的具体结构框图，为了便于说明，仅示出了与本发明实施例相关的部分。消息处理集群中包括投票站节点、普通站节点和监控站节点，所述消息处理集群中的相应节点中内置有所述令牌的处理装置，该令牌的处理装置 7 包括：令牌环路建立单元 71、令牌产生或者销毁单元 72 和令牌回环时间监测单元 73。

[0130] 其中，令牌环路建立单元 71，用于在消息处理集群的应用层建立令牌环路；

[0131] 令牌产生或者销毁单元 72，用于当节点加入或者退出令牌环路时，采用简单多数投票算法产生或销毁令牌；

[0132] 令牌回环时间监测单元 73，用于在令牌的传递过程中，监测令牌环路中的各个节点上次传出令牌的时间戳，若监测到有节点上次传出令牌的时间戳与当前时间的时间差超出设定的回环时间，则发起路由发现过程，重建令牌环路。

[0133] 具体的，所述令牌产生或者销毁单元 72 包括：

[0134] 节点数量预设模块，用于预设消息处理集群中的节点数量为 N；

[0135] 节点数量统计模块，用于当监测到有节点加入由消息处理集群中的 N 个节点组成的令牌环路或者退出所述令牌环路时，动态计算所述令牌环路内的节点数量 P；

[0136] 令牌产生模块，用于当 P 大于 (N+2)/2 舍尾取整值时，若令牌环路内无令牌，则自动产生新的令牌；

[0137] 令牌维持模块，用于若令牌环路内有令牌，则维持原令牌；

[0138] 令牌销毁模块，用于当 P 小于等于 (N+2)/2 舍尾取整值时，若令牌环路内有令牌，则销毁该令牌。

[0139] 进一步地，所述令牌产生或者销毁单元 72 还包括：

[0140] 节点加入等待模块，用于若令牌环路内无令牌，则系统挂起，等待新的节点加入所述消息处理集群中。

[0141] 进一步地，所述装置 7 内置于消息处理集群中的相应节点中，所述消息处理集群中的节点包括投票站节点、普通站节点和监控站节点。

[0142] 本发明实施例提供的令牌的处理装置可以应用在前述对应的方法实施例一中，详情参见上述实施例一的描述，在此不再赘述。

[0143] 值得注意的是，上述令牌的处理装置实施例中，所包括的各个单元只是按照功能逻辑进行划分的，但并不局限于上述的划分，只要能够实现相应的功能即可；另外，各功能单元的具体名称也只是为了便于相互区分，并不用于限制本发明的保护范围。

[0144] 以上所述仅为本发明的较佳实施例而已，并不用以限制本发明，凡在本发明的精神和原则之内所作的任何修改、等同替换和改进等，均应包含在本发明的保护范围之内。

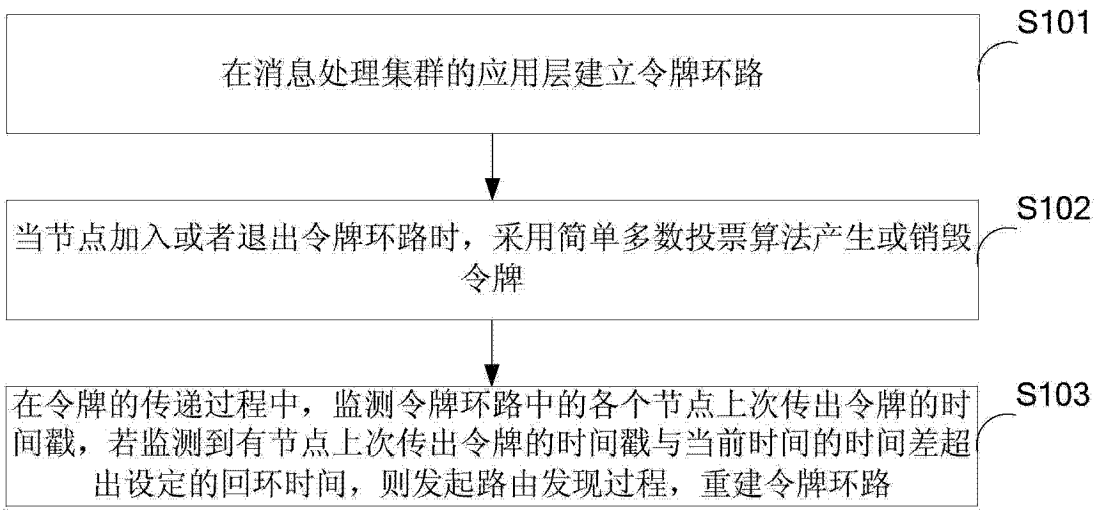

S101
在消息处理集群的应用层建立令牌环路
S102
当节点加入或者退出令牌环路时，采用简单多数投票算法产生或销毁令牌
S103
在令牌的传递过程中，监测令牌环路中的各个节点上次传出令牌的时间戳，若监测到有节点上次传出令牌的时间戳与当前时间的时间差超出设定的回环时间，则发起路由发现过程，重建令牌环路

图 1

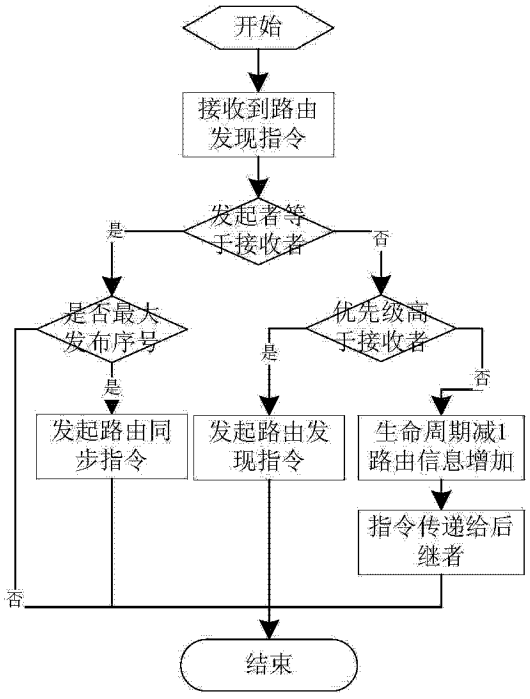

开始
接收到路由发现指令
发起者等于接收者
是
否
是否最大发布序号
优先级高于接收者
是
否
是
否
发起路由同步指令
发起路由发现指令
生命周期减1路由信息增加
指令传递给后继者
否
结束

图 2

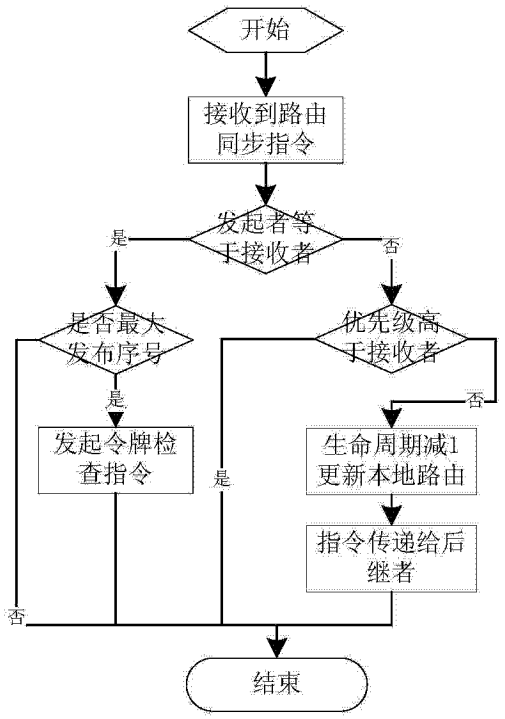

开始
接收到路由同步指令
发起者等于接收者
是
否
是否最大发布序号
优先级高于接收者
是
否
是
否
发起令牌检查指令
生命周期减1更新本地路由
指令传递给后继者
否
结束

图 3

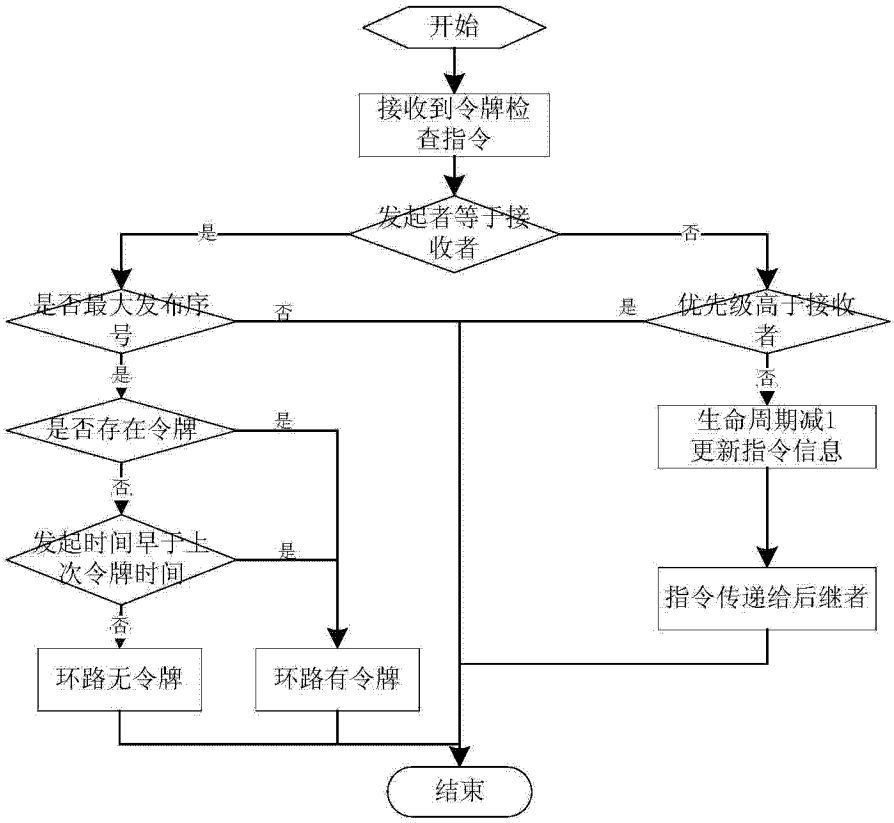

开始
接收到令牌检查指令
发起者等于接收者
是
否
是否最大发布序号
否
是
优先级高于接收者
是
否
是否存在令牌
是
否
生命周期减1
更新指令信息
发起时间早于上次令牌时间
是
否
指令传递给后继者
环路无令牌
环路有令牌
结束

图 4

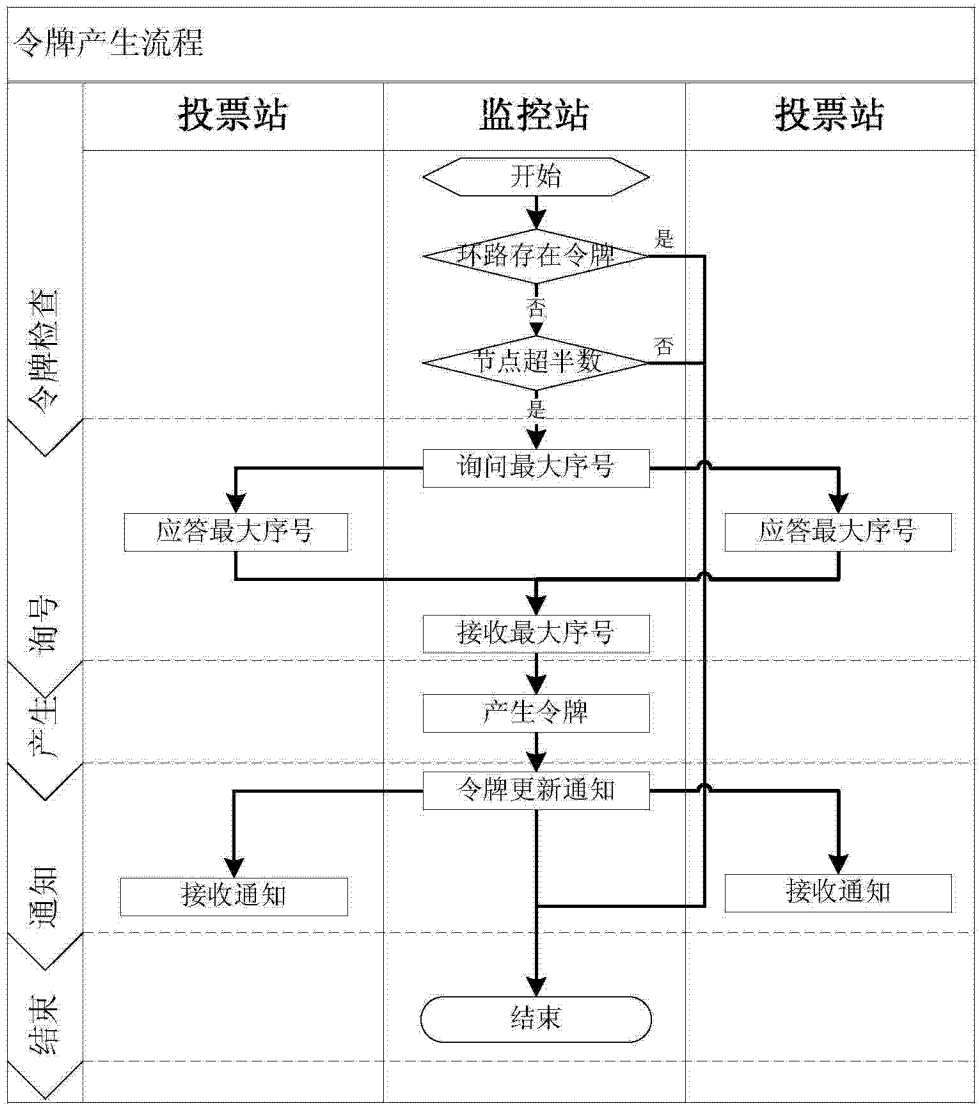

令牌产生流程
投票站
监控站
投票站
令牌检查
询号
产生
通知
结束
开始
环路存在令牌
是
否
节点超半数
否
是
询问最大序号
应答最大序号
应答最大序号
接收最大序号
产生令牌
令牌更新通知
接收通知
接收通知
结束

图 5

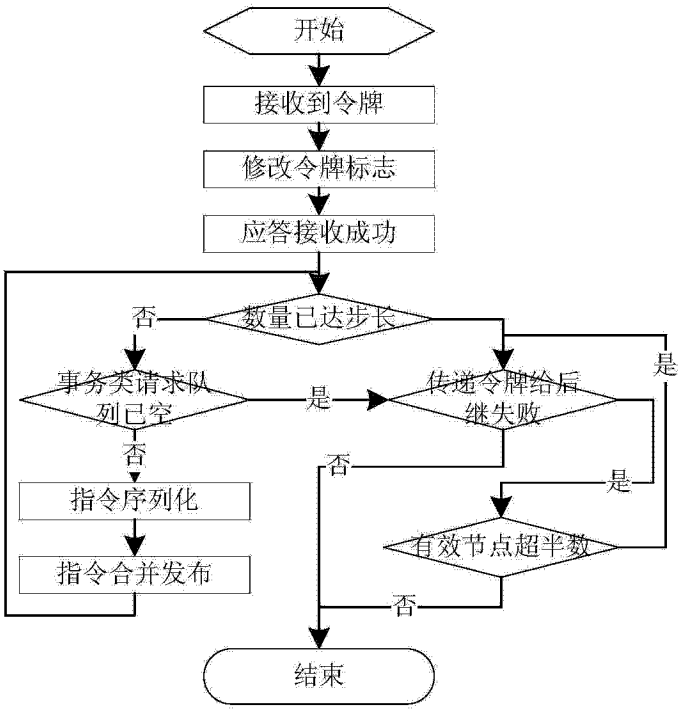

开始
接收到令牌
修改令牌标志
应答接收成功
否
数量已达步长
是
事务类请求队列已空
是
传递令牌给后继失败
否
否
是
指令序列化
有效节点超半数
指令合并发布
否
结束

图 6

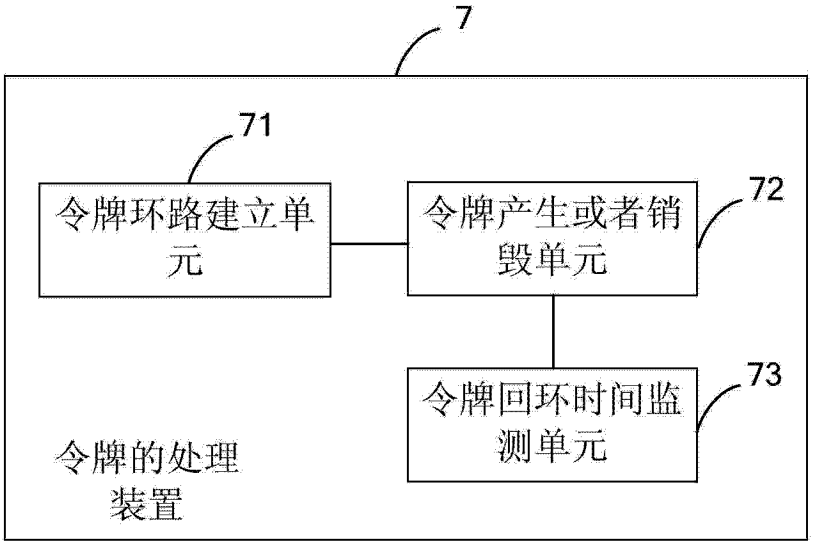

7
71
令牌环路建立单元
令牌产生或者销毁单元
72
令牌回环时间监测单元
73
令牌的处理装置

图 7