[1    引言 3](#_Toc13827)

[1.1     编写目的 3](#_Toc15657)

[1.2    背景 3](#_Toc12546)

[1.3    参考资料 3](#_Toc18311)

[1.4   术语定义及说明 3](#_Toc17400)

[2  设计概述 3](#_Toc3093)

[2.1   任务和目标 3](#_Toc17187)

[2.1.1      需求概述 4](#_Toc22713)

[2.1.2      输入与输出 4](#_Toc27879)

[2.1.3      条件与限制 5](#_Toc25081)

[2.1.4      详细设计方法和工具 5](#_Toc7328)

[3  模块详细分析 5](#_Toc21256)

[3.1创建更换leader控制服务服务 5](#_Toc8429)

[3.2 身份变更消息： 5](#_Toc14162)

[3.3 收到新区块到达和blockchain区块插入消息： 6](#_Toc14740)

[3.4 更换leader通知消息： 6](#_Toc20848)

[4   模块详细设计 6](#_Toc8483)

[4.1    模块结构设计 6](#_Toc21158)

[4.2    功能详细设计 6](#_Toc26874)

## 1    引言

### 1.1     编写目的

说明更换leader控制服务与其它模块的交互和设计考虑，为程序员编码提供依据。

### 1.2    背景

开发版本：0730

开发小组：算法组

模块描述：重选leader流程中，控制重选leader的流程，包括何时启动重选会话及最后更换结果的通知。

### 1.3    参考资料

以太坊源码

### 1.4   术语定义及说明

主节点：在重选leader时，本节点为leader

## 2  设计概述

### 2.1   任务和目标

1 详细说明更换leader控制服务与其它模块之间的接口。

2 详细说明更换leader控制服务的内部设计。

### 2.1.1      需求概述

控制重选leader的流程，包括何时启动重选会话及最后更换结果的通知。

### 2.1.2      输入与输出

* 输入：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | **消息码** | **消息体** | **来源** | **实现方式** | **备注** |
| **身份变更消息** | 待定 | 新身份枚举 | CA服务 | 消息订阅 | 身份变更通知 |
| **新区块Header准备完毕消息** | 待定 | header头 | 区块生成服务 | 消息订阅 | 矿工挖矿结果验证完毕后的消息 |
| **区块验证请求消息** | 待定 | header头 | CA服务 | 消息订阅 | 经过CA验证来自合法的validator |
| **区块验证结果消息** | 待定 | 消息体(leader、高度，结果) | 区块验证服务 | 消息订阅 | 目前只有成功消息 |
| **重选leader成功消息** | 待定 | 包含Leader差距值和投票列表 | 重选会话服务 | 消息订阅 | 主从节点会话都会发出该消息 |

* 输出：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **消息码** | **消息体** | **实现方式** | **备注** |
| **更换Leader成功消息** | 待定 | 节点信息 | 消息通知 | 节点信息为空表示当前无leader |
| **会话启动消息** | 待定 | 重选主节点信息 | 消息通知 |  |
| **会话停止接口** | 待定 | 无 | 消息通知 |  |

### 2.1.3      条件与限制

### 2.1.4      详细设计方法和工具

流程图。

## 3  模块详细分析

### 3.1创建更换leader控制服务

创建更换leader控制服务的实例，并对其中的成员进行初始化。进行身份变更消息的订阅。

最后启动update() 等待接收订阅的消息。

并返回该对象

### 3.2 身份变更消息：

收到从ca发来的身份变更消息。如果当前不是验证者节点，收到验证者消息，则启动更换leader控制服务协程；如果当前是验证者节点，收到非验证者消息，则关闭更换leader控制服务协程。

### 3.3 收到新区块到达和blockchain区块插入消息：

从消息中获取区块高度n和leaderId。如果当前轮次小于n+1，则初始化控制服务， 再判断n+1是否是广播轮次。如果是则设置当前状态为等待验证区块状态，并设置1.5s定时器。

### 3.4 **区块验证请求**消息：

判断当前状态是否为等待区块验证状态。如果是则通过接口获取当前leaderId。如果请求的发送方是当前leader，当前块高等于缓存的处理高度，则停止1.5s定时器，设置当前状态为等待区块验证结果消息，启动2s定时器。

### 3.5 **区块验证结果**消息：

判断当前状态是否为等待区块验证结果状态。如果是则通过接口获取当前leaderId。如果请求的发送方是当前leader，当前块高等于缓存的处理高度，则停止2s定时器，设置当前状态为空闲状态。

### 3.6 **重选leader开始**消息：

判断当前状态是否为重选leader状态。如果是则向主节点和从节点发送停止消息。然后判断当前验证者的角色。如果是leader则向主节点发送启动消息。如果是follower则向从节点发送启动消息，并重设2s定时器。

### 3.7 **重选leader成功**消息：

把当前状态置为等待区块验证状态。重选次数置0，停止定时器，开启1.5s定时器，向leader身份计算服务发出更换leader消息通知。

### 3.7 **超时**消息：

判断当前状态。如果是等待区块验证状态，则判断当前验证者是否是leader。如果不是则把当前设置为重选leader状态，向leader身份计算服务发出 更换leader结果消息；如果是等待验证区块结果状态，则把当前设置为重选leader状态，向leader身份计算服务发出 更换leader结果消息；其他状态不做处理。重选轮次加一。

## 4   模块详细设计

### 4.1    模块结构设计

### 4.2    功能详细设计

主要数据结构：

type leaderControlService struct {

newCACh chan newCAEvent //CA

newCASub event.Subscription

CurrentHeader \*types.Header

role int //-1: not found. 0: super miner 1:non-super miner 2:super verifier 3:non-super verifier

newHeaderCh chan newHeaderEvent //新区块Header准备完毕消息

chainHeadSub event.Subscription

insertBlockCh chan insertBlockEvent // blockChain 区块插入消息

insertBlockSub event.Subscription

validateBlockReqCh chan validateBlockReqEvent // 验证区块消息

validateBlockReqSub event.Subscription

posResultCh chan posResultEvent // pos 验证结果消息

posResultSub event.Subscription

reElectLeaderStartCh chan core.ChainHeadEvent // 重选leader 开始消息

reElectLeaderStartSub event.Subscription

reElectLeaderSuccCh chan reElectLeaderSuccEvent // 重选leader 成功消息

reElectLeaderSuccSub event.Subscription

ca \*testca.CFTEST

nodeId string

curNumber uint64 //上次选举leader时的块高

state uint

reelectCount uint //重选轮次

scope event.SubscriptionScope

leaderChangeFeed event.Feed

quitCh chan int

timer \*time.Timer

}

主要函数

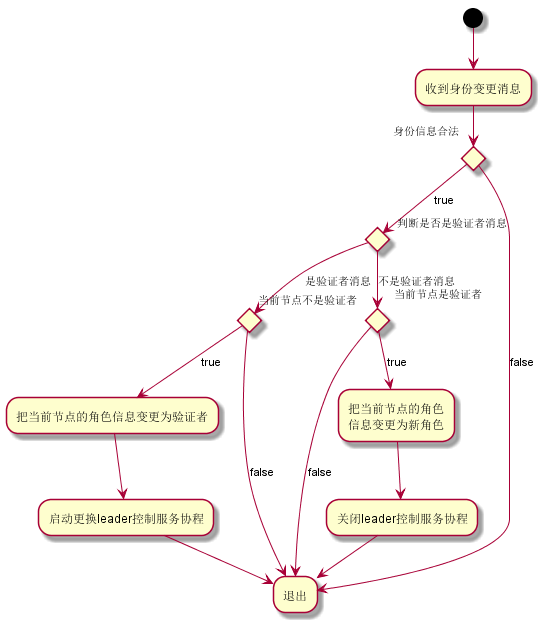
函数名称： newCaRoleHandle()

函数功能：收到身份变更消息时进行相应处理

输入：新身份枚举

输出：

流程图：



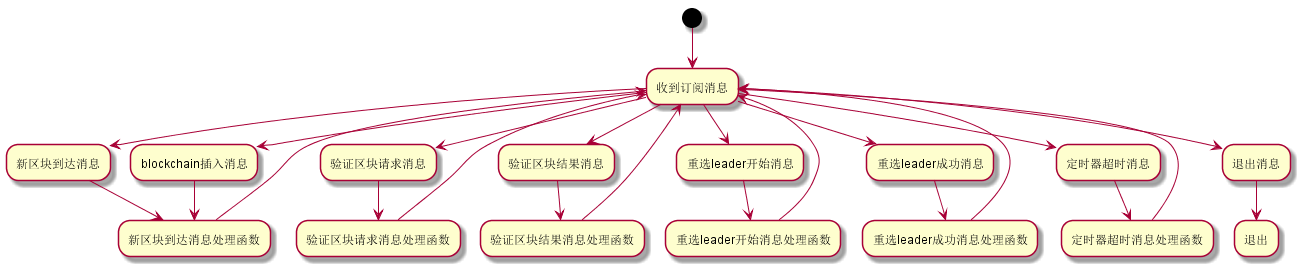
函数名称：leaderSwitchControl

函数功能：根据收到的订阅消息启动相应的处理流程

输入：

输出：

流程图：



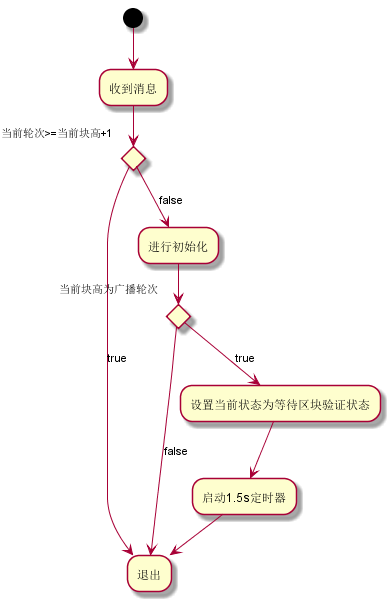
函数名称：newHeaderMsgHandle()

函数功能：对新区块到达消息和blockchain插入消息进行处理。

输入：块高、leader

输出：

流程图：



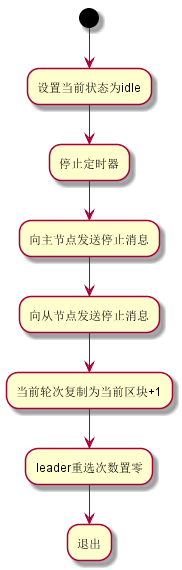
函数名称：init()

函数功能：对新区块到达消息和blockchain插入消息进行处理。

输入：块高、leader

输出：

流程图：



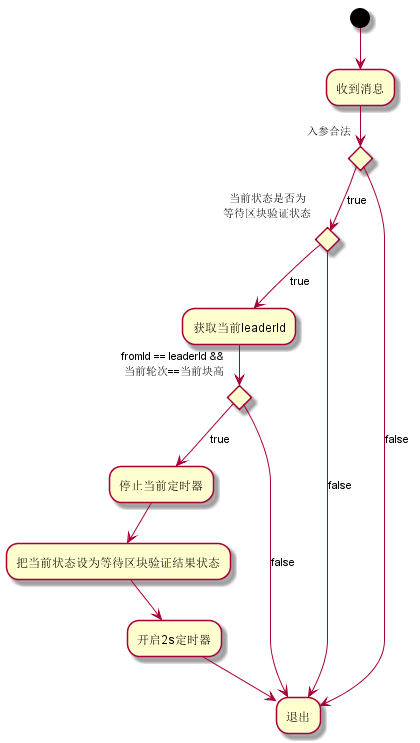
函数名称：validateBlockReqHandle()

函数功能：对区块验证请求消息进行处理。

输入：发送方id、块高

输出：

流程图：



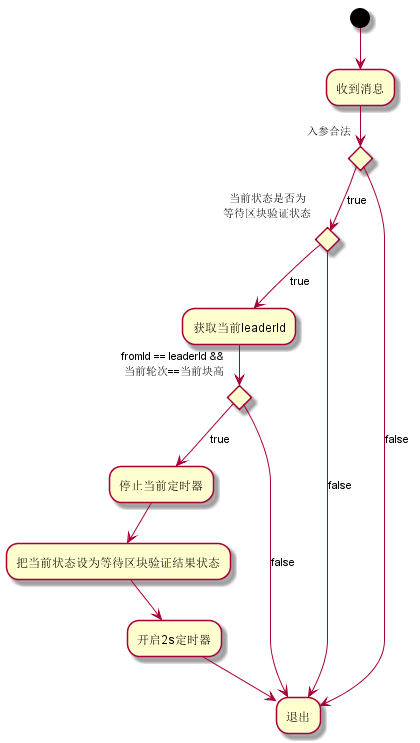
函数名称：posResultHandle()

函数功能：对区块验证结果消息进行处理。

输入：发送方id、块高

输出：

流程图：



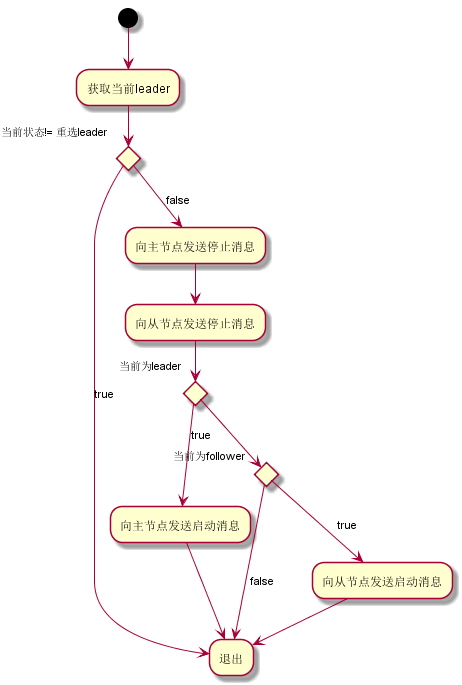
函数名称：reelectSessionMsgHandel()

函数功能：对重选leader开始消息进行处理。

输入：

输出：

流程图：



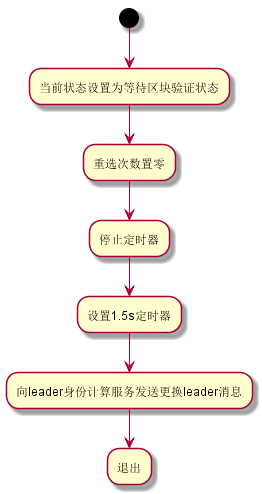
函数名称：reelectLeaderSuccessHandle()

函数功能：对重选leader成功消息进行处理。

输入：leaderId

输出：

流程图：



函数名称：timeoutHandle()

函数功能：对定时器消息进行处理。

输入：

输出：

流程图：

