### 接收到广播区块的处理过程

见源码org.ethereum.manager.BlockLoader#blockWork。

1）接收到其他节点广播来的区块。

2）检查其区块头。

3）检查其是否已经在区块缓存里有了，有则返回。

4）检查其父是否已经有了，有则继续。

5）检查交易根哈希是否就是交易列表对应的MPT根哈希。

6）检查所有交易的发送者nonce是否合规，除了与库一致，还要在列表内是合规的。

7）读取库当前的根哈希，保留后用。

8）开始执行区块，即逐条处理每条交易。

9）根据库（区块级快照）创建交易级快照。

10）根据交易级快照创建合约级快照。

11）执行交易，这个过程中涉及到如检查交易gasLimit累计是否可能超过区块gasLimit，交易执行中gas是否足够，nonce是否符合及递增，余额是否足够及转账，创建合约，改变合约状态，保存合约代码，改变账户的状态等等。

12）执行交易完成后，提交合约级快照的状态改变刷到交易级快照。

13）整理交易执行结果，将退款和费用分别转给发送者和矿工，删除需要删除的账户。

14）提交交易级快照的状态改变刷到区块级快照。

15）将交易执行结果和收据添加到列表。

16）所有交易执行完毕后，计算对矿工和其他的奖励，至此区块执行完毕。

17）根据交易收据列表计算得到MTP根哈希，检查与广播区块的是否一致。

18）根据交易收据列表里的布隆过滤器计算区块的布隆过滤器值，检查与广播区块的是否一致。

19）获取库此时的根哈希，检查与广播区块的是否一致。如果不一致，则执行过程与其他节点不一致，就是共识未达成。库的状态恢复为区块执行前的，之前保留的根哈希用上了。若一致就提交，也就是将数据刷到数据库。

20）接着将区块及难度及是否分叉、交易序号及区块哈希及收据持久化，并重置最优区块，重置库的根哈希为该区块上的根哈希。

21）上面处理过程中，对待最优链处理和分叉情况稍有不同。完成后，接下来就是处理分叉，省略。

### 创建新区块的处理过程

见源码org.ethereum.mine.BlockMiner#BlockMiner。

1）

<http://www.sohu.com/a/150828885_99901444>

## 技术点：

### 版权和许可申明

Copyright (c) 年份 公司或个人

<http://www.gnu.org/licenses/>

<https://choosealicense.com/licenses/gpl-3.0/>

<http://www.lupaworld.com/article-21618-1.html>

GNU General Public License v3.0，使用其代码的应用可以商用不开源，但若开源分发则必须仍以此许可为许可。

### Springwork

org.ethereum.cli.CLIInterface

org.springframework.stereotype.Component