我在论文里又找到有些有趣的数据，比如节点的数量只有十个到十五个，数据产生的速度是每分钟一个到三个，百分之十的节点随机的产生数据请求，每个数据大小是1mb，平均每个块的大小小于10kb，数据延迟每一挑模拟10ms。

Node varying form 10 to 50

Data items generated 1 to 3 per minute

Data are requested randomly by 10 percent of nodes

Data item size 1 MB

Average block size less than 10KB

Data dalay 10ms one hop

The transmission includes data request and transmission, data dissemination,and all blockchain broadcasting information

我下面按着数据产生到存储的顺序，分配一下每个小组的具体工作，

##### Producer:

1. Generate a data item 模拟产生数据（区块链支持）
2. Generate metadata 根据数据产生metadata，hash签名（区块链支持）
3. Boradcasts （网络）

##### Other nodes:

1. Receive the metadata （网络）
2. Cacluates which set of nodes wil store 资源分配算法
3. Mine block pos算法

##### The node which mines the next bolck

1. Pack all received metadata items 区块链支持

Along with the storing node information （cacluate， 资源分配算法）

Into blcok

1. Broadcasts 网络

##### Other nodes:

1. Receive the block 网络
2. If a node is chosen to be a storing node

Get the data from the producer and stores them 网络

3、if a node is chosen to be stored block ,store it

##### Customer:

1. Checks the metadata item 区块链支持
2. Fetches nodes that store it 网络  
   3. Sends the data request 网络  
   4. Receive data 网络

**资源分配算法：**

Input： 每个节点的负载（维护一个全局字典）

每个节点的迁移率（文中给出，固定值）

每两个节点之间的距离（维护一个距离矩阵，固定值）

Output： 需要存储该data的节点集

**New POS 算法：**

挖矿、验证

Input： S（维护一个全局字典）

Q（维护一个全局字典）

T (所有的节点共享一个时钟)

B（全局变量，由计算得到）

Hi

**Docker+websockets（区块链支持+网络）：**

部署节点

节点相互通信（异步）

文件传输

模拟网络

维护区块链

整个的代码分为三个大部分，资源分配算法、pos算法、区块链支持和网络，前两个算法偏理论实现，后边区块链部分偏工程，算法又资源分配算法较难，pos算法简单。

前两个算法与第三部分的联系并不是特别多，我想的是把数据结构统一，在统一的列出两个node和block节点，这样每个部分都可以根据自己需要构造数据，便于测试，同时后期对接的时候也方便。

我想的是咱们用一个腾讯文档，

1. 文档里列出数据结构模板和样例，

2. 每个组在自己要用到的数据结构后边做注释。

3. 每个组想要改动数据结构的时候，把改动同步到腾讯文档。

4. 改动数据结构的时候，不要破坏原有的结构，尽量使用新增字段新增函数，或者经过与对应组充分讨论之后再做修改。

下边是两个草案，我个人感觉，可能node各个组用的比较多，block就交给区块链组维护吧，大家分组之后，按自己的需求列出数据结构要有的字段和方法，**周三晚八点**咱们统计一下，给出一个大家都认可的模板和样例。

**两个重要的数据结构：**

所有全局变量都随block一起广播，接收到的节点解包并更新自己的值

Node{

固定值

节点的迁移率 range 字典

节点的距离 distence 字典

变量（每收到一个块或挖出一个块时更新）

节点的负载 storage 字典

节点拥有的token数量 token 字典

}

Block{

}