# peer和channel介绍

## peer

官网解释：

一个网络实体，维护ledger并运行Chaincode容器来对ledger执行read-write操作。peer由Member拥有和维护。

自我理解：

一个网络实体，主要的任务就是用来维护账本，可以通过chaincode对账本完成记账、执行交易的操作。每个节点（peer）都需要隶属一个org（组织），因为隶属于org，所以在此节点上只允许隶属org的用户进行对channel、chaincode等操作。

## channel

官网解释：

通道是构建在“Fabric”网络上的私有区块链，实现了数据的隔离和保密。通道特定的账本在通道中是与所有对等节点共享的，并且交易方必须通过该通道的正确验证才能与账本进行交互。通道是由一个“配置块”来定义的。

创建通道是为了限制信息传播的范围，是和某一个账本关联的。每个交易都是和唯一的通道关联的。这会明确地定义哪些实体（组织及其成员）会关注这个交易。

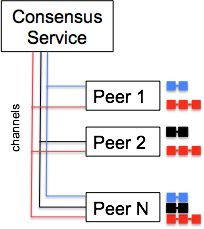
自我理解：

channel的主要作用就是数据的隔离和保密，chaincode必须依附于channel。节点需要加入channel之后才能执行位于channel上的交易。节点是否有权限加入channel，是根据channel的配置文件来定义的，配置文件定义了此channel上允许那些org的节点能加入。

## peer与channel的关联

如下图所示，一个peer可以订阅多个channel，channel可以被多个peer订阅。peer 1,2和N订阅红色通道，并共同维护红色账本; peer 1和N订阅蓝色通道并维护蓝色账本; 类似地，peer 2和peer N在黑色通道上并维护黑色账本。

在这个例子中，peer N在订阅了所有通道，我们看到每个通道都有一个相关的账本。 一般来说，我们称不涉及所有peer的账本为子账本，而涉及所有peer的账本另一种是系统账本，即全账本。



# peer种类

## 排序节点（orderer peer）

提供排序服务的节点，此节点不进行记账。在交易流程中，当背书节点执行背书之后，将交易的结果发送给排序节点，节点对结果进行一系列的验证和排序。之后再将验证和排序的结果发送给各个背书节点。

在一个Fabric网络中至少有一个orderer节点，orderer节点是是网络启动时就定义好的。orderer节点并不处理交易的数据，只负责对接受的数据进行正确性确认和排序操作。所以无法通过控制orderer节点的方式来对交易的数据进行篡改。

## 背书节点（endorser peer）

一笔交易需要成功运行并修改账本，必须要通过背书节点的背书操作。背书操作就可以认为此节点对此结果的一个认可，交易定义背书策略时，可能需要多个节点进行背书，意味着这笔交易需要多个节点对交易的结果认可之后，交易的结果才能真正的修改账本。

非背书节点上，执行修改账本的交易是不会修改账本的。即如果想修改账本，就必须在背书节点上执行交易。背书节点是通过背书策略中的算法进行选举，暂时还不清楚选举背书节点的规则。

## 记账节点（comitter peer）

负责进行记账的节点，当排序节点将验证排序之后的结果发送给主节点之后，将验证通过的结果发送至记账节点。记账节点将对发送来的结果进行记账，更新本身节点上对应的账本。

## 锚节点（Anchor Peer）

锚节点是通道中能被所有对等节点探测、并能与之进行通信的一种对等节点。通道中的每个成员都有一个（或多个，以防单点故障）锚节点，允许属于不同成员身份的节点来发现通道中存在的其它节点。也就是在channel中的每个org中至少有一个节点，能作为所属org位于此channel的代表，可以与其他org的节点进行通信。

## 主节点（leader peer）

每一个Member在其订阅的channel上可以拥有多个peer，其中一个peer会作为channel的leading peer代表该Member与ordering service通信。ordering service将block传递给leading peer，该peer再将此block分发给同一member下的其他peer。