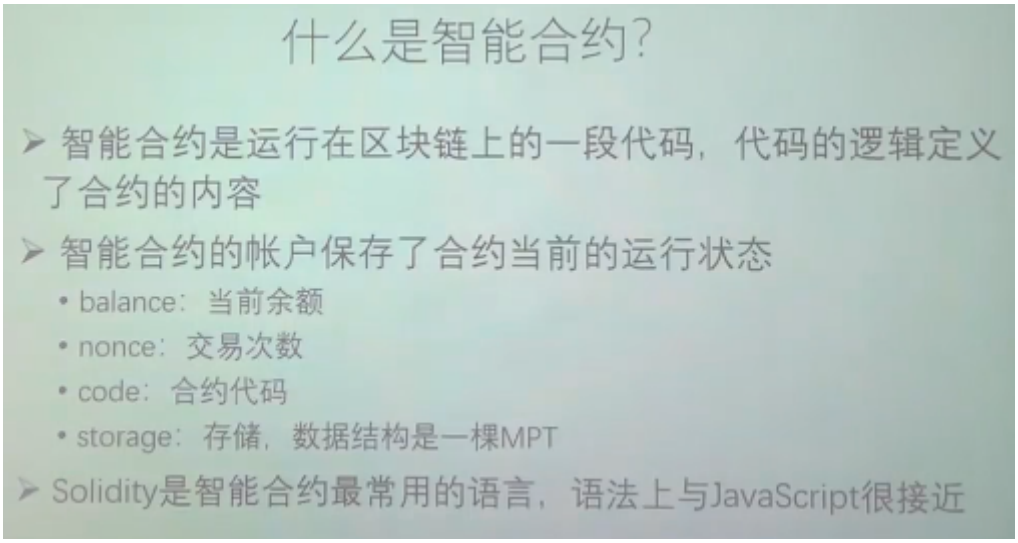
智能合约

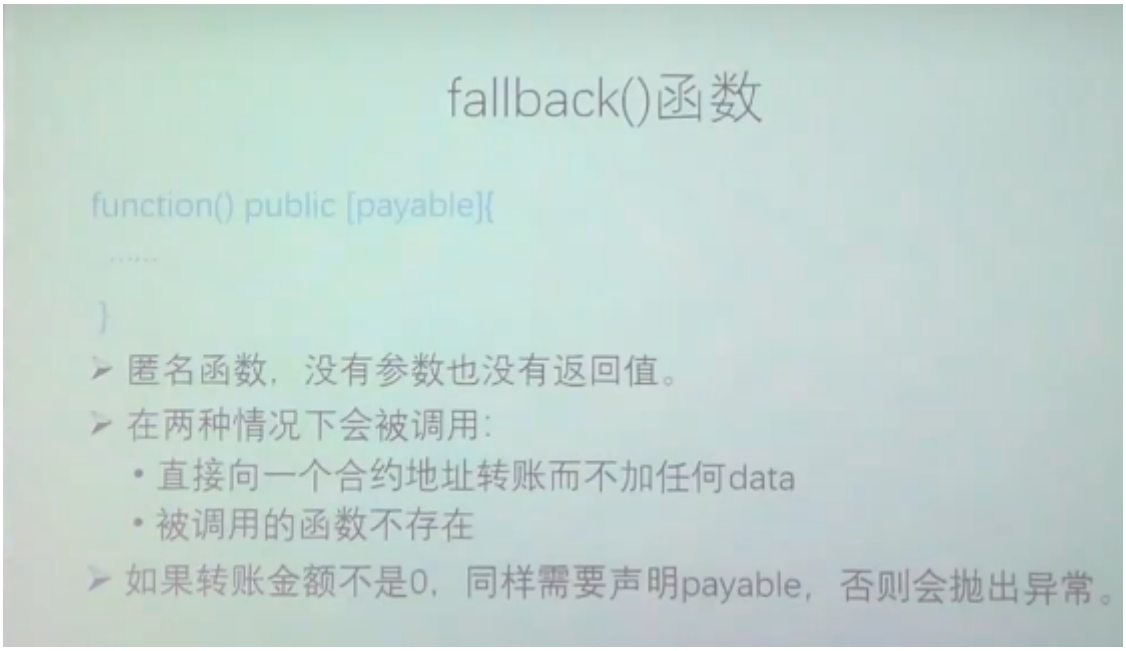


使用语言：solidity

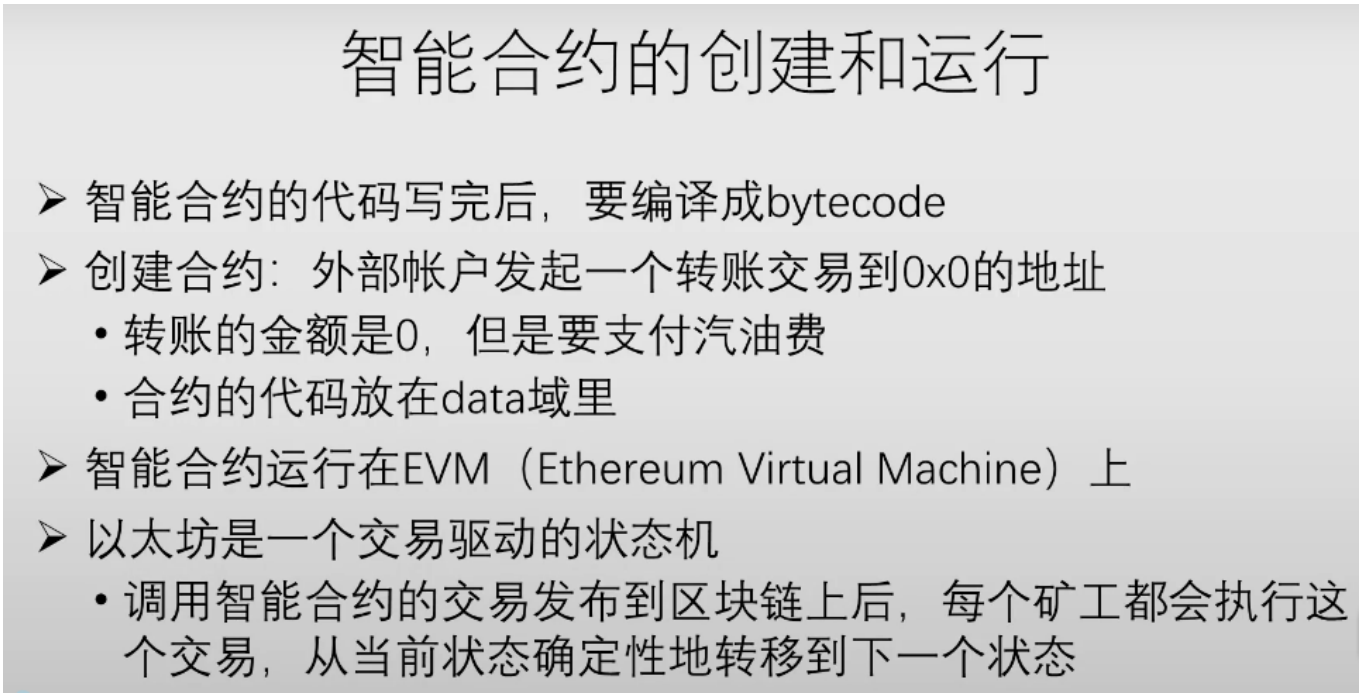
一个合约如何调用另一个合约：

- 直接调用 ，如果失败，全部回滚
- 使用地址类型的call()函数 ，只回滚被调用的函数
- 代理调用 delegate call

payable：标志此函数接受转入帐
fallback函数（仅合约账户有）：



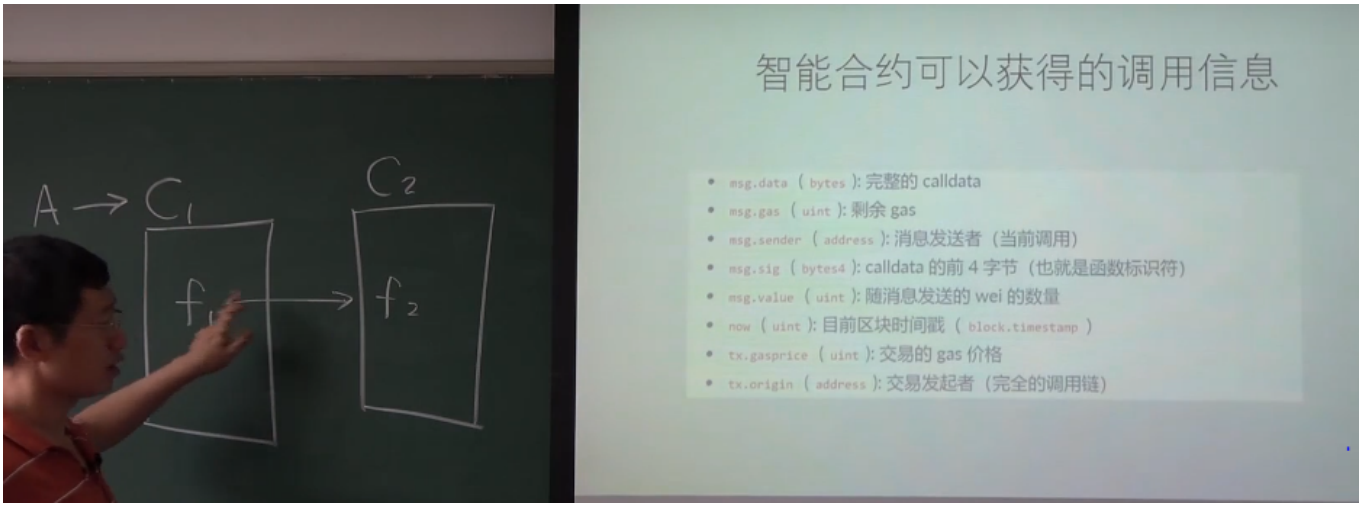
合约的创建与运行



EVM的好处：寻址空间256位，普通计算机只有64位

所有的全节点都是每次先执行将要打包的交易（包括合约操作），然后执行后修改本地维护的状态树。全部执行完后，算出state tree root放在区块头里，再计算nonce值挖矿。一旦一个全节点算出了符合target的值（即挖到矿），就将自己这一区块所打包的交易广播出去，所有全节点听到后都独立将所有的交易在本地执行一遍，算出三棵树的根hash值，并验证挖到矿的节点nonce值是否正确。若正确，则维护更新本地的状态树。

由于出于验证的角度，所有节点在这个状态机下运行完后都要处于同一个状态，所以智能合约是不支持多线程的，同时也没有真的的随机函数及获取节点计算机系统参数的函数。



msg.sender不同于tx.origin ，如上图，对于合约C2，它的f2函数的msg.sender是f1,tx.origin是A。

三种转账方式：transfer，send，call.value
区别：transfer出错不支持回滚，另外两个支持。call.value被调用的同时，移交所有剩余的汽油费，另外两个只转移2300，大概仅支持写一条log的汽油费。