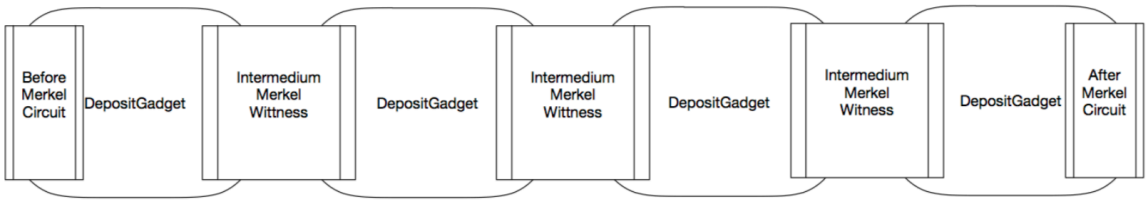
Circuit Understanding

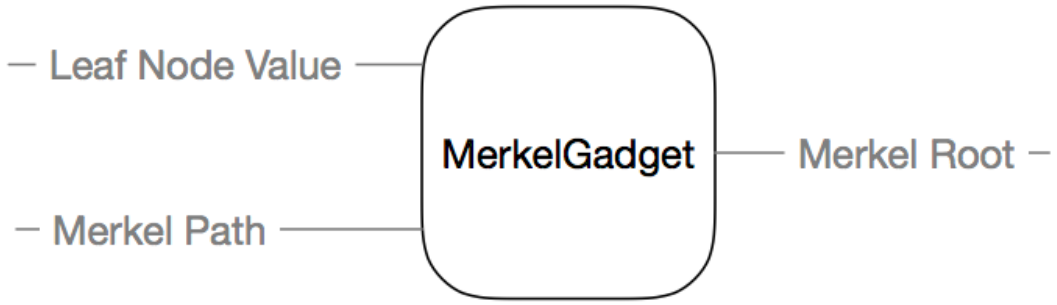
综合讨论

1. 为什么之前提出的节省中间验证的优化不可行?

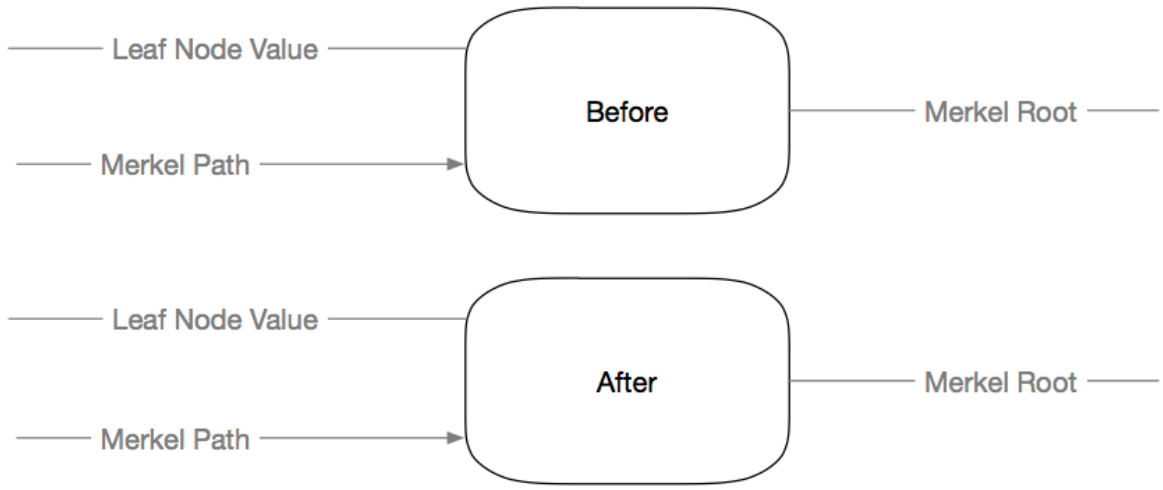
   之前优化的想法是这样的，中间状态不验证，而只有block出入的merkel tree root进行验证。



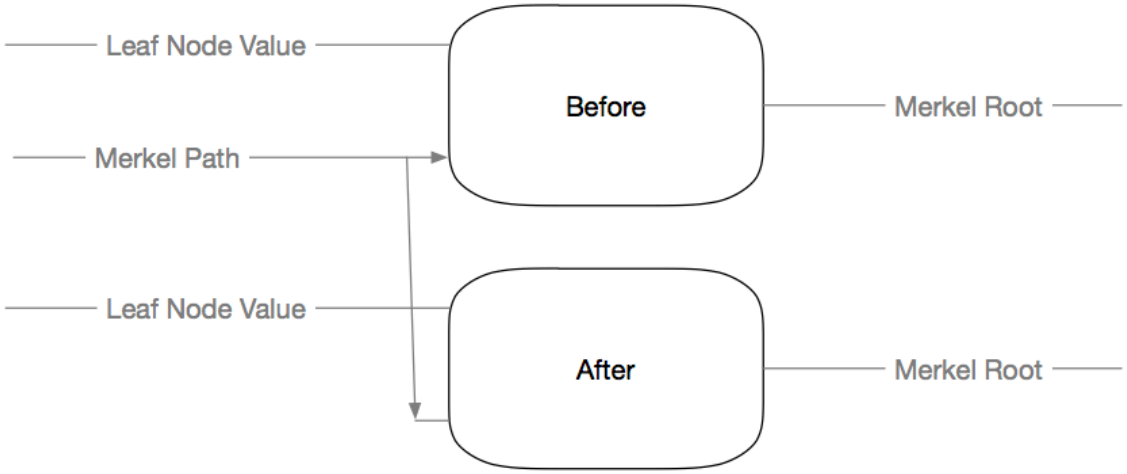
   该方案不可行是因为必须保证Merkel Path的一致性。如下图所示，本来Merkel Root验证电路本身比较简单，输入是path和leaf，输出是merkel root。



   对于Loopring的电路设计，我们有before和after两个merkel tree需要被验证，于是有：



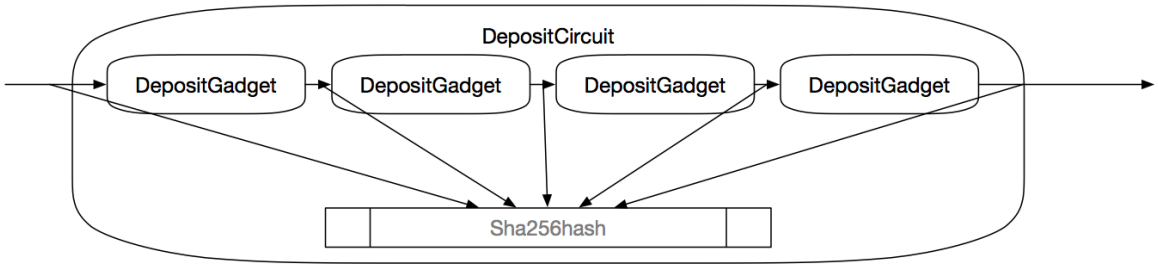
   这里有个问题，因为before和after完全没有联系，那么任意构造的merkel树都可以单独通过before或者after的验证，也就是说before和after可以来自完全不同的两颗merkel树，且各自符合merkel逻辑。所以为了保证内部的操作在同一棵merkel tree上完成，也就必须让before和after产生联系，目前看来只有这样一种方案，如下图所示：



即对before和after使用同一份merkel path做验证，这样保证了对节点状态的改动始终在同一棵merkel树上，但是此方案的副作用就是每次只能用来验证一个节点的一次改动，因为合并的多个节点的改动不可能有公用的merkel path。

2. 为什么需要sha256在电路的最后（这里指deposit/withdraw，ringsettle似乎没有这个计算），直接使用merkel root行不行？为什么合约还要计算sha256，直接拿statement行不行？如果合约不用计算的话，换个hash行不行？

   答案都是不行，或者说当前设计下不行。首先看看当前的设计：



   基本上有两块，上面一行是对merkel树的验证，下面一行是以block为单位的一个hash。可以对比以太坊来理解这个模型，上面一行depositGadget是对eth stateDB的操作，将每一个request对应到一个eth的tx，而下面这个sha256hash就是对应到eth block hash。

   1. 只使用merkel root不行的原因：

      类比eth，因为对statedb的操作和对block上tx记录的操作无关，如果没有这个block hash的check，那么只要statedb的check能过，block里面包含的任意tx和statedb的改动无关也能通过验证。这样block的状态和statedb的状态就不能保持一致了。

   2. 为什么合约要计算sha256hash这个statement，直接拿statement不是也能通过验证么？

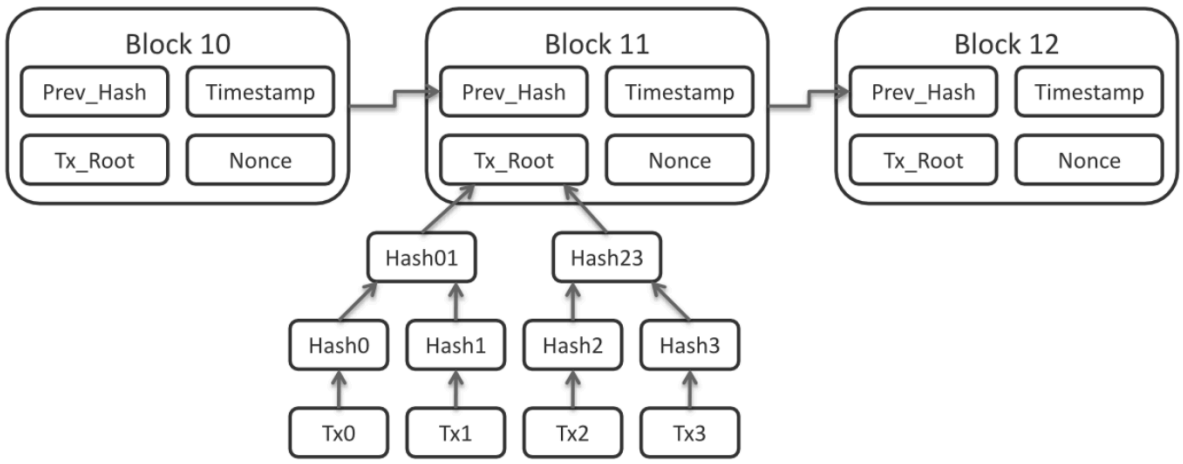
      直接拿statement确实能正确地通过验证，但是这里没办法保证通过验证的就是当前提交的block的，比如有两个block有待验证，合法的block0和非法的block1，在调用合约的时候，我们提交block1的data和block0的proof，合约验证block0的proof肯定是通过的，但显然不能按照block1的tx去执行，也就是需要一个block和proof的绑定过程，这里可能是零知识证明设计上的一个需要注意的地方，就是不能仅依赖zksnark proof，还必须有业务层上的一些附加的逻辑。

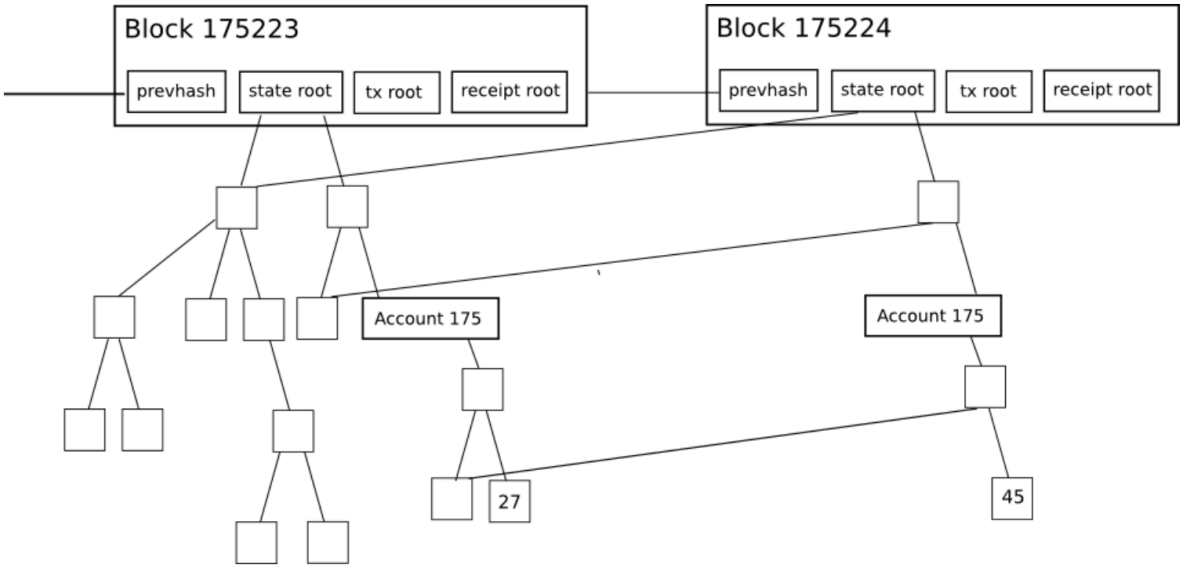
   3. 因为合约必须对这个statement进行检查，所以不可以取消或者换算法（sha256在合约里面便宜）。

3. 讨论理解电路设计过程。

Loopring的电路设计可以隐约看到以太坊的逻辑（个人猜测是受了ethsnark的影响，有时间可以过一下ethsnark的代码逻辑），原因可能是因为request本身已经组织成了一个blockchain，基本上就是request对应tx，block对应block，merkel树对应stateDB。

下面两张图是eth的block内部两个主要数据结构tx_root和state_root的组织方式





基本对应关系如下：

Ethereum	**Loopring circuit**	**验证逻辑**	**补充说明**
Transaction	Request	Signature(public key)	
tx root	PublicDataHash	Hash equivalence	eth tx root采用的merkel组织方式。但loopring这边不需要这么麻烦，直接计算full data hash效果一样。
state root	Merkel Root Before/After	Merkel root equivalence	ETH只有一棵MPT用于存储stateDB，但loopring需要证明所有的操作都在同一棵merkel树上。

这样在电路设计上的考量就比较清楚了，从0开始设计电路的流程大概是一个选型的过程，比如是用UTXO还是Account，中间状态用什么存储，状态变化用什么描述?

	账户模型	世界状态模型	状态变化描述	补充说明
BTC	UTXO	？？	tx	
ETH	Account	MPT	tx	
Loopring	Account	layered Merkel	request	由于zksnark，因此还有一些需要业务层的设计来保证，如前面提到的Layered Merkel树唯一，以及request block和block proof绑定等等。
Loopring-derivative :)	??	??	??	