BTCRelay

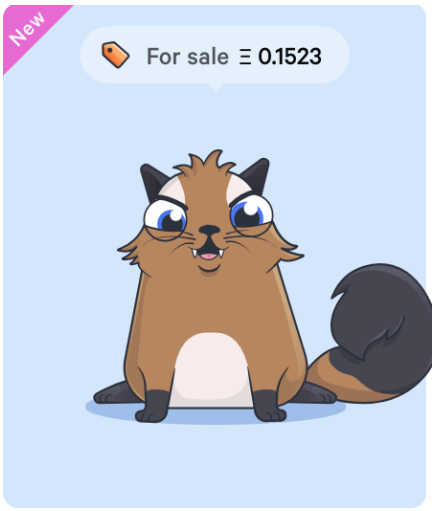
1. BTC Relay 简介

最初BTC Relay是在以太坊基金会下构思和资助的，当BTC的开发人员Joseph Chow采纳并得到ConsenSys的支持，此项目也随之加入ConsenSys。2016年5月2日 ConsenSys 团队宣布正式推出 BTCRelay，此项目被认为是第一个侧链项目，其中比特币为主链，以太坊为侧链。从技术的角度上讲，BTC Relay 是以太坊上的一个智能合约，它使用 Serpent 语言实现了BTC的 SPV 功能。通过以太坊的智能合约功能验证比特币网络上的交易。把以太坊网络与比特币网络以一种安全、去中心化的方式连接起来。

BTCRelay 背后的思想是在比特币和以太坊节点之间建立一种信任机制，如果一笔比特币交易发送给以太坊的一个全节点，它能够识别、验证这笔交易，而且根据预先的设定调用相关的智能合约。

在此之前，以太坊的节点只能验证以太坊的交易请求，比特币的节点也只能验证比特币的交易请求。这两个不同的区块链系统之间没有一种可信的通信机制。这种限制的根源在于： 一个节点要验证一笔交易，必须依赖账本的历史数据，否则无法判断该交易是否被确认。区块链本质上是一个分布式账本，通过p2p通信机制，在每一个全节点上有一个全量账本数据的副本。每一个节点在本地拥有足够的历史数据验证链内的交易，但是无法验证链外的交易。比如对于一个以太坊的节点来讲，它自身没有比特币的区块数据，即使能够接受比特币的交易，也无法验证其正确性。

如果有一种信任机制能令以太坊能够验证比特币的交易数据，进而可以触发以太坊上的智能合约。Dapp开发者可以在客户端设置一个“比特币支付”按钮，让比特币的持有者也能直接使用基于以太坊的Dapp，不需要将比特币兑换为以太币。 一个潜在的应用场景是使用比特币购买加密猫。加密猫是以太坊上的 ERC721 token，只能通过以太币购买。

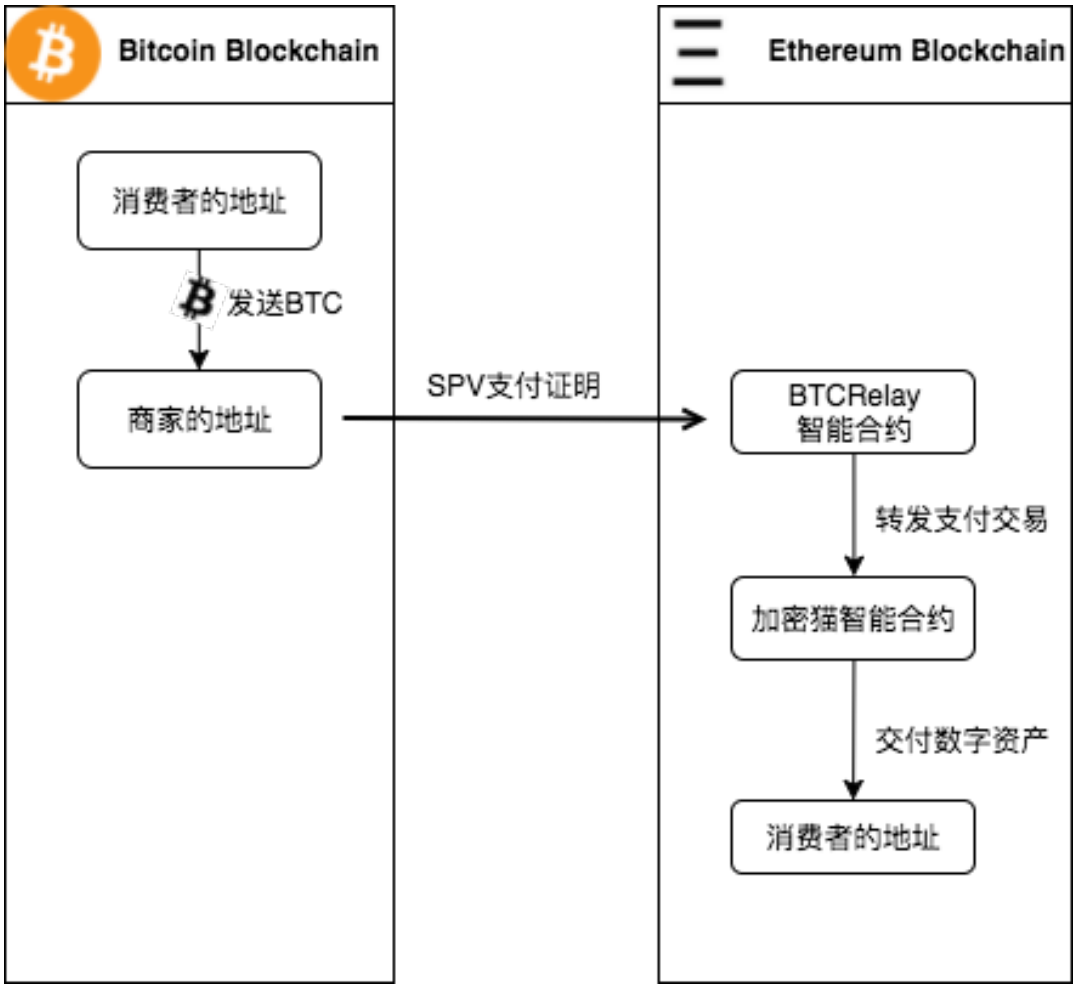


待售加密猫：0.1523ETH

通过BTCRelay，加密猫的商家也可以接受比特币支付。具体的业务流程如下：

1. 买家从Dapp网站上，选定加密猫，获取商家比特币地址，价格等信息。

2. 买家向商家的地址支付指定的比特币，并且在转账交易中附加额外的信息：买家的以太坊地址，商品ID等信息。
3. 系统监控商家的比特币地址，收到比特币支付消息之后，生成相应的SPV证明，与交易信息合并发送给BTCRelay智能合约。
4. BTCRelay 验证此交易是否被打包，如果验证成功转发给加密猫智能合约。
5. 加密猫智能合约从支付消息中提取转账额、加密猫ID、和买家的以太坊地址等数据，确认商品状态信息之后，将数字资产转移到买家地址上。



流程示意图：使用比特币购买加密猫

2. BTCRelay 设计结构

链间交易的验证的本质是建立预言机(Oracle)机制

从本质上讲，验证链外交易是要建立一个预言机机制，从链外第三方获取相关信息，使得智能合约能够验证链外交易。一般来讲，有两种设计方案：

1. 让链外第三方验证人组负责验证交易，每一个验证人独立的验证交易，如果通过就附加上自己的签名发给智能合约。智能合约通过收集验证签名，判断交易是否获取了足够多验证人的认可。如果是，那么就认为此交易是正确的。

这个方案的优点是简单、通用，适用于很多场景。但是缺点是验证人作弊的成本比较低。所以验证人组的人数不能太少，否则验证人就会比较容易串通；但是也不能太大，否则管理成本和运算成本就会过高。

2. 让链外第三方验证人输入比特币的每一个区块头数据，智能合约利用SPV原理验证每一笔比特币交易。

   这个方案验证人的作弊成本比较高，因为每个区块头包含POW，验证人如果要篡改一个区块头，必须要组织算力重新挖矿，才能提交符合难度系数的区块头；而诚实的验证人不需要任何算力去挖矿，只需要通过p2p网络同步最新的区块即可。但是这个方案也有缺点：智能合约需要管理和存储所有区块头数据。

对于比特币来讲，存储的成本大约是每年新增4.2M字节（每个比特币区块头80字节，平均每10分钟一个区块）。存储成本还是可以接受的，所以 BTCRelay 选择了第二种解决方案。

BTCRelay 的组织架构

BTCRelay 有4个核心组件构成，他们彼此之间的关系如下图所示：

- Relayer

  Relayer 由社区自愿者构成，运行一个软件从比特币网络获取区块头数据，然后传输给向BTC Relay智能合约。它在系统中承担着预言机的角色和任务，而且这也是 BTCRelay 项目名称的由来。

- Header Management

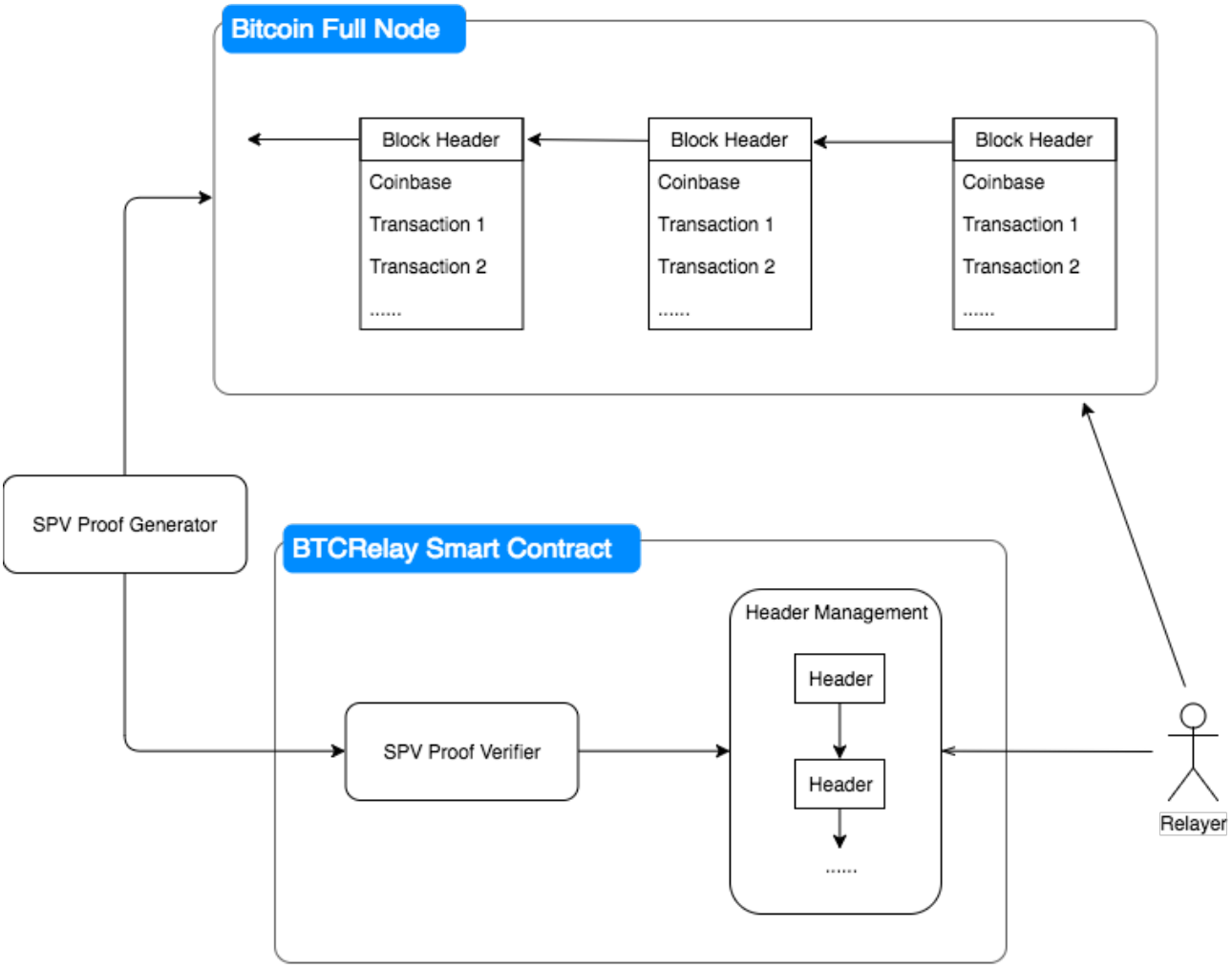
  这部分负责验证、存储和管理比特币的区块头列表。每次接收到来自 Relayer 的最新区块头数据，要验证其hash是否满足难度要求，然后验证是否为孤块，验证通过后保存到区块头数组中，并且根据区块高度和hash值建索引。

- SPV Proof Generator

  SPV Proof Generator 负责为指定的交易生成 SPV Proof。BTCRelay 本身并没有提供 SPV Proof Generator 的实现。但是可以使用第三方提供的开源工具，比如 bitcoin-proof。

- SPV Proof Verifier

  SPV Proof Verifier 根据 SPV 原理验证比特币交易，此外为保证交易的最终确定性，还要验证交易的确认数是否大于等于6。所有验证条件满足后，会把交易转发给预先指定的其它智能合约处理。



BTCRelay 架构示意图

此系统提现了信任最小化的设计原理，比特币和以太坊的智能合约是经过实际检验的去中心化分布式账本系统；SPV Proof Generator / Verifier 的安全性来自于 Merkle 树的抗强碰撞性；Relayer 的安全性来自于比特币的POW难度系数，如果要提供一个假的区块头，必须组织庞大算力才能挖出假的区块。系统中的每一个组件都很难作弊，所以 BTCRelay 整体上成为一个去信任的分布式系统。

3. 详细技术分析，源代码分析

从技术的角度来讲，BTCRelay 的主要功能包括：

1. 验证一个已经被确认的比特币转账交易
2. (可选)将已经验证的比特币交易转发给其它智能合约
3. 存储比特币的区块头
4. 检查智能合约内部最新的Bitcoin区块信息

BTCRelay 智能合约的详细的ABI接口信息请参考：http://btcrelay.surge.sh/BitcoinRelayABI.js。

这里只介绍关键的几个接口函数:

- verifyTx(rawTransaction, transactionIndex, merkleSibling, blockHash)

验证指定的比特币交易是否被主链上的区块确认，而且确认数至少是6.

输入参数为比特币交易，及其SPV Proof相关数据

- rawTransaction - 比特币交易的二进制字节数组表示
- blockHash - 此交易所属区块的hash值
- transactionIndex - 此交易在所属区块的索引值
- merkleSibling - hash数组，用于重新计算Merkle Root

返回值：

- 如果交易验证失败，则返回 0；
- 如果交易验证成功，则返回交易的hash值

- relayTx(rawTransaction, transactionIndex, merkleSibling, blockHash, contractAddress)

  通过调用 verifyTx() 验证比特币交易的正确性，如果验证通过，把交易转发给指定的目标合约地址。目标合约必须实现 processTransaction(bytes rawTransaction, uint256 transactionHash) returns (int256) 函数。

  输入参数为比特币交易，SPV Proof相关数据，以及目标智能合约的地址。

  - rawTransaction - 比特币交易的二进制字节数组表示
  - blockHash - 此交易所属区块的hash值
  - transactionIndex - 此交易在所属区块的索引值
  - merkleSibling - hash数组，用于重新计算Merkle Root

  返回值：

  - 如果交易验证失败，则返回错误码 ERR*RELAY*VERIFY；
  - 如果交易验证成功，则返回目标智能合约 processTransaction() 函数的返回值。

- storeBlockHeader(blockHeader)

  验证并且保存新的比特币区块头, 验证条件包括：是否符合难度系数，前一个区块是否已经存在.

  输入参数：

  - blockHeader - 比特币区块头的二进制字节数组表示.

  返回值：

  - 如果交易验证失败，则返回 0；
  - 如果交易验证成功，则返回此区块的高度。

- getBlockHash(blockHeight)

根据指定的区块高度，返回区块头hash

- getBlockHeader(blockHash)

  根据指定的区块头hash，返回区块头数据

- getBlockchainHead(), getLastBlockHeight(), others

总结

BTCRelay 通过内嵌一个“小型的比特币区块头数据库”，结合 SPV 原理实现了从比特币到以太坊的、去信任的、单向跨链通信。虽然没有实现资产的双向锚定，并不是一个完整的侧链项目，但是其设计简单而精巧，对于侧链跨链协议的实现有很重要的参考价值，这个方案也可以拓展到其它跨链通信的场景中。

但是从官网和 github 上看，BTCRelay 有比较长的时间没有更新了。它自身也存在以下的问题：

1. 手续费太高：提交区块头、提交SPV Proof 都需要手续费，这部分成本由Relayer，和SPV Proof Generator 承担。
2. 社区成员Relayers活跃度太低，从其mainnet来看，现在只有单一的Relayers
3. 延时太高： 由于比特币交易至少需要一个小时(6个确认)才能达到一般最终确定性的要求，所以从交易被确认到 BTCRelay 验证通过至少有1个小时的延时。

开放式思考

1. 如何建立一种信任机制，让比特币节点验证以太坊的交易，或者智能合约的状态更新?
2. 如果要建立一种信任机制，让以太坊识别EOS的交易，还能用 BTCRelay 这种方案吗?
3. 如果要建立一种信任机制，让以太坊识别传统金融机构的交易(比如银行转账，股票交易等)，有什么解决方案?
4. 假如你是一个程序员，请尝试写一个程序生成指定比特币交易的SPV Proof，并且验证SPV Proof的正确性。
5. 尝试写一个使用比特币众筹的Dapp。

参考资料列表

- BTCRelay 官网: http://btcrelay.org/
- BTCRelay 源码：https://github.com/ethereum/btcrelay
- BTCRelay 智能合约ABI接口: http://btcrelay.surge.sh/BitcoinRelayABI.js
- BTCRelay 主网：https://etherscan.io/address/0x41f274c0023f83391de4e0733c609df5a124c3d4
- BTC Relay Is Live
- Enabling Blockchain Innovations with Pegged Sidechains
- Trust pegging of BTC in Ethereum - 双向挂钩