

# 什么是区块链？

区块链到底是什么？相信很多人还是一知半解，今天我们就通过几个例子一起来了解一下到底什么是区块链。

区块链本质上是一个去中心化的分布式账本数据库。其本身是一串使用密码学相关联所产生的数据块，每一个数据块中包含了多次比特币网络交易有效确认的信息。

这是区块链的定义，但要逐步了解区块链，我们还需要一步步了解如下东西。

## 二

**区块链主要是为了解决什么问题？**

我们首先要思考一个问题，区块链这个概念当初是怎么提出的呢？

这就要从传统的中心化交易来谈起。

借用 PG Two 买夹克的例子，我们来看看中心化和去中心化体系的区别。



网购夹克的整个流程依托于支付宝展开, 因此, 这个买卖过程是中心化的。

其实支付宝最初开发的目的, 是为了解决交易双方的信任风险问题, 无论是 PG TWO 还是卖家君, 在这点上只能完全信任支付宝和它背后的马云。

往大了说, 中心化系统由资金雄厚和技术实力强大的机构、企业做信任背书。

中心化体系具备管理高效的优势, 但它的不足也比较明显。

仍以支付宝为例, 全部交易记录和账本都存储在支付宝服务器上, 但是假设某天所有相关的服务器不幸被坏蛋捣毁, 那么 PG TWO 付的款

（或卖家君还没有到手的夹克钱），还有其他买家、卖家的资金，甚至你我存在余额宝的钱，都会消失在这个互联网世界里。

该找谁说理去？

就算我气汹汹地找到支付宝对质：“我还有 5 万在余额宝里面！”可中心账本已经被彻底破坏，谁又能证明我的话为真呢？

若得不到有效证明，最终我也只能忍着泪跟这沓钱 say goodbye 了。

这个问题就是区块链要着重解决的问题：如何去中心化——打破像支付宝这种中心化的互联网产品或者平台的单方面道德风险。

这时，去中心化系统的优势就凸显出来了。

设想一下，如果全网络存在许多的记账节点，能够共同记录支付宝上每一笔交易、转账和提现等，也就是说支付宝所有账本在全世界有很多备份；就算服务器被黑客攻击，相同的账本副本也都好好地保存在其他节点上。

在区块链的世界里，不需要大企业做信用背书。

### 三

区块链技术是怎么做到的呢？

区块链技术的原理在于：给每一个参与交易的成员一个平等的记账权利。

还拿刚才的支付宝的例子，来谈一下在区块链应用的条件下，交易是怎么完成的。

（1）PG Two 买夹克，把款项支付给支付宝，记录下付款记录，同时把付款记录广播出去，让大家都记录下来；

（2）卖家收到支付宝发来的买家付款信息后，安排发货，记录发货信息，同时把发货信息广播出去，让大家都记录下来；

（3）PG Two 收到夹克，确认收到货物，记录下自己的收货记录，并把这条记录广播出去，让大家都记录下来；

（4）支付宝收到确认收货信息，把款项打给卖家，记录这条转款记录，并把这条记录广播出去，让大家把这条记录记下来。

利用区块链技术，我们发现，之前担心的支付宝单方面利用中心化优势而可能产生的道德风险很好的被约束了。

大家记录，是为了防止欠债人要赖等损害信任的情况出现。因为如果支付宝想要私吞款项，交易中的其他各方都有交易记录，可以共同为该交易的存在作证。

所以，区块链给一个市场中的交易各方都提供了一个平等的记录权利，任何交易都会被全体成员记录下来作为公开信息而存在，当任何

一方想要违背协议发生道德风险时，其他成员可以共同作证，制止这种行为。

这就是区块链的精髓所在，它让每一个参与其中的人都有知情权和决策权，自己的权利不会被中心化的一方剥夺。

## 四

区块链上的区块。

由于系统的初始设定，不同区块链产生区块的速度不一样。

比如比特币区块链大概每 10 分钟挖出一个区块，而以太坊区块链的出块时间约 14 秒。

每个区块包含这段时间内产生的所有交易记录，如一个新挖出的比特币区块就包含了前 10 分钟内的交易信息。

除此之外，还包含时间戳和前一个区块的哈希值，等等（关于区块的组成，我们在后续文章再详细了解，这里先把握基础知识）。

这是很聪明的设计：每诞生一个新的区块，就会被盖上相应的时间戳，新产生的区块按照区块挖出的时间顺序连接到链条上去。

这样，区块链无限延长，账本数据库也能无限扩大、容纳无穷尽的交易信息。

更令人拍案叫绝的是，新区块里的每一笔交易数据，都由相应的交易发起方进行数字签名，链上所有人都可以用交易发起方的公钥（公钥是公开的）验证该数字签名的真实性。

此外，之前挖出的区块里的交易数据将永久记录在区块链上，几乎无法篡改。

另一方面，篡改数据难于上青天，成本极高。

有意者必须足够财大气粗，并且拥有至少全网 51% 的算力。而要达到这样的计算能力，有相关人士表示：矿机成本+电费+其他，成本约 5.5 亿 RMB。

综上， 在这样一个充满不信任感、危机四伏的网络环境里，区块链或许是当前解决信任问题的最佳方案。