

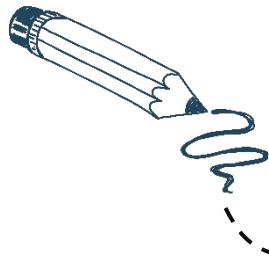
# PoRep

---- Filecoin中的复制证明

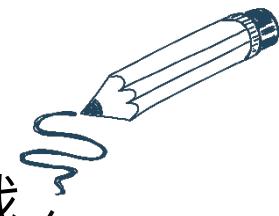


# 目 录

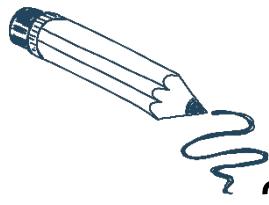
## CONTENTS



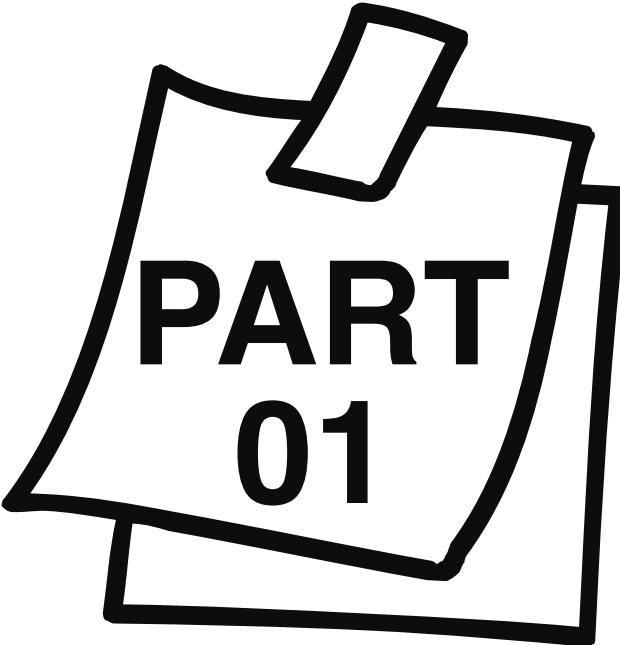
1、Filecoin是什么？



2、实现存储服务遇到的挑战

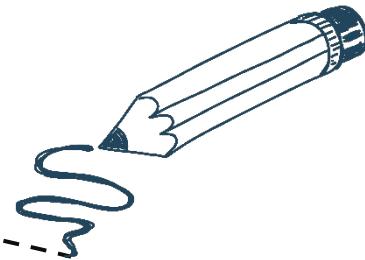


3、如何解决挑战？PoRep 和 PoSt！

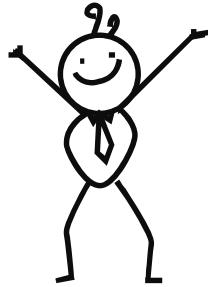
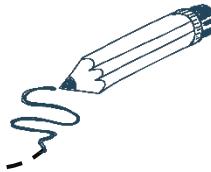


# PART 01

Filecoin是什么？



# Filecoin是什么?



首先，我们知道Filecoin是一个  
分布式的存储系统！

为了鼓励大家参与并促使该系统发  
展，系统中使用一种代币，也叫  
filecoin，符号为FIL。



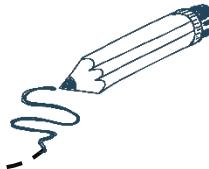
# Filecoin是什么？



构造一个存储和检索的交易市场，  
即数据存储服务的买卖市场。

提出复制证明（PoRep）和时空证  
明（PoSt）作为共识算法，保证系  
统安全和公平。

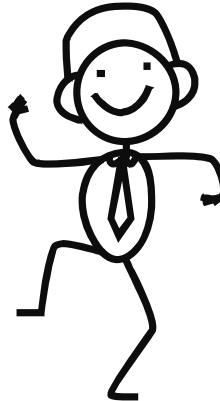
# Filecoin是什么?



## 两大特点

作为区块链项目，Filecoin具备两大特点：

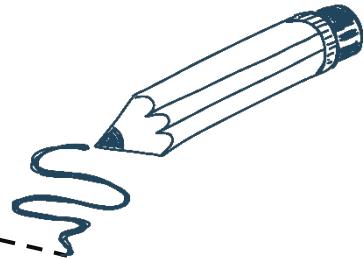
- 1 使用分布式记账本的同时，完成现实世界的需求（数据储存）。
- 2 使用PoS（硬盘空间）作为共识机制，相较于使用PoW（算力）更节约资源，也更分散。



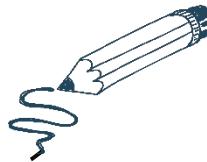
因此，与传统的存储服务相比，Filecoin更加安全、高效、低成本。所以，Filecoin项目被很多人看好，被认为是  
非常有前景的公链项目。

## PART 02

实现存储服务遇到的挑战



# 如何实现存储服务？



组成市场的角  
色有哪些？

举个例子

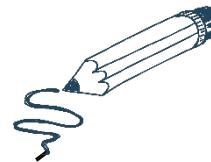
- 客户
- 存储矿工
- 检索矿工
- 网络



客户c有一个数据D想要在网络中存储5份，存储服务提供者（存储矿工）s1、s2、s3、s4、s5通过竞争，获得机会来存储D。整个网络根据每个矿工存储的空间按比例给予每个矿工获得记录下个区块的权利，记录的同时获得代币奖励。当客户需要取回数据时，检索矿工帮他取回。

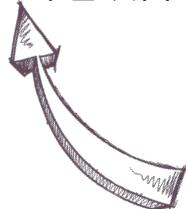
那么，会碰到哪些（意外）情况呢？

# 如何实现储存服务? -- 会遇到的问题



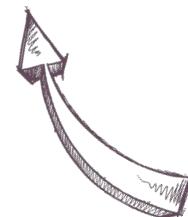
客户c的数据是否确实被5名  
储存矿工完整存储?  
也许s1存储了，但s3没有，  
当c向s3索取(确认)数据时，  
s3从s1获取，然后再交给  
c。

-- 外包攻击



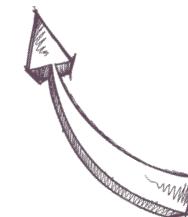
甚至，也许s1、s2、s3、  
s4、s5是同一自然人注册  
的5个账户，ta更倾向于只  
存储一份。

-- 女巫攻击



更甚者，也许客户c和存储矿  
工也是一起的，c的数据D由  
很短的D'生成。这样ta只要存  
储很小的D'即可。

-- 生成攻击





这下听懂了吧?  
-- 什么? 还是不明白!

呵呵, 没关系。  
让我们来做个有趣的类  
比!

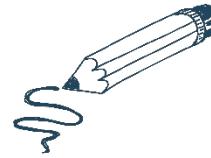
# 有趣的类比：课堂记笔记

某中学一位语文老师有个习惯，每节课他都会划一些重点，并要求所有同学记录在笔记本上，在学期期间的自习课上，随时拿出来复习。

假设我很讨厌这件事或者纸太贵，我该怎样应对？

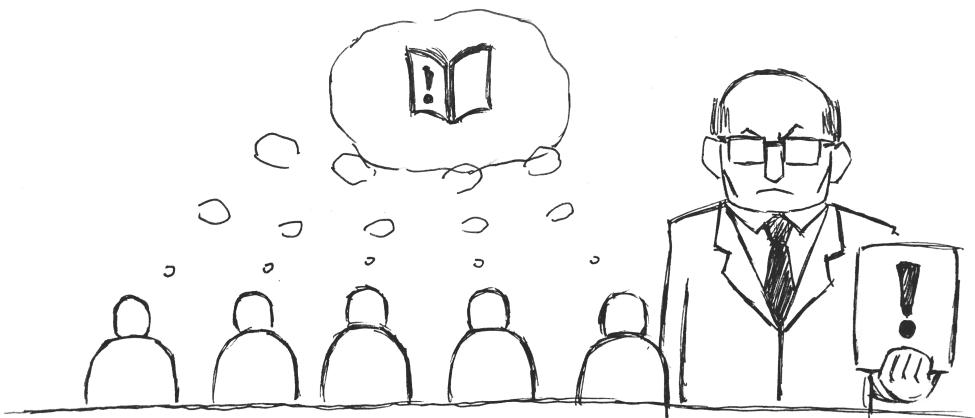
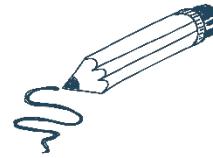


# 情况一



每次老师检查我的笔记时，  
我都会拿同桌的笔记来假  
装是我的。 (外包攻击)

## 情况二



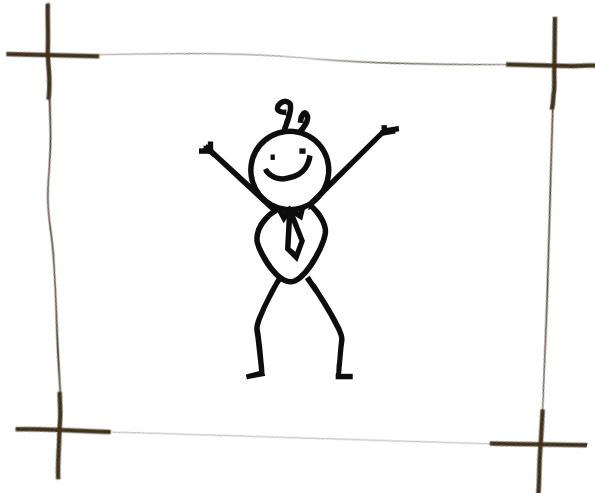
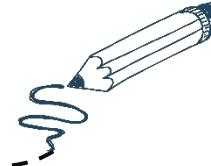
虽然声称每人都有笔记，事实上全班只有一份笔记，老师检查谁的笔记时，这份笔记就用作谁的。（女巫攻击）

## 情况三



老师每次说明天检查笔记  
时，我都会通宵写一份。  
(生成攻击)

# 不诚实行为会带来哪些影响？



## 出现的问题

### 【存储服务角度】

客户冗余备份数据期望达到的安全性得不到保障。

### 【区块链角度】

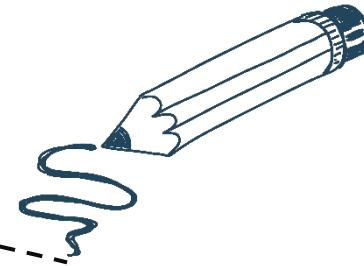
不确定存储矿工真实存储的数据量，从而无法形成公正的共识机制。

# PART 03

如何解决？

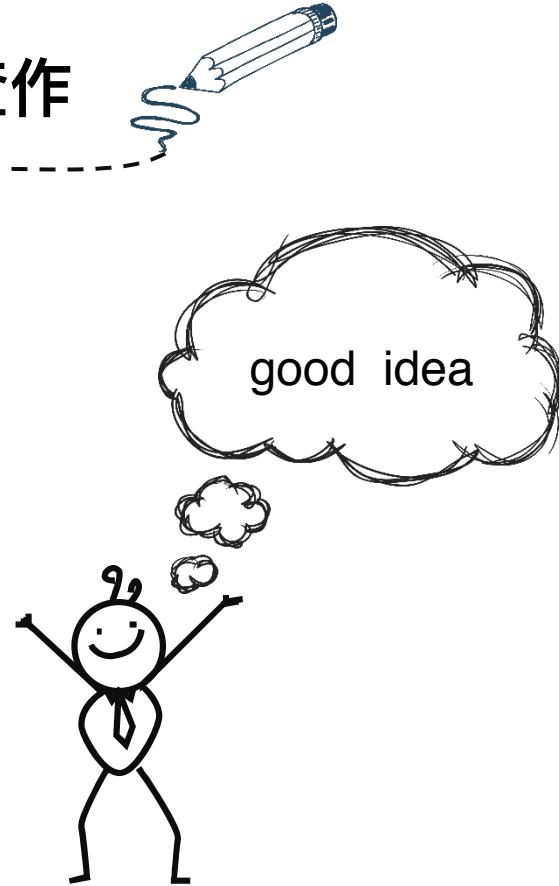
— PoRep 和

PoSt!



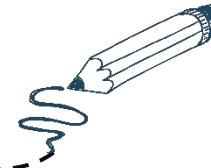
# 语文老师如何解决笔记检查作弊？

- ① 要认得每个人的笔迹，每个人的笔迹不同，所以一份笔记只能属于一个同学。
- ② 会在检查笔记之前一个小时才告诉同学们需要检查笔记，请他们做好准备。
- ③ 这时候，作为一个想要偷懒或者想节约笔记本空间的同学，你还有好的作弊方法吗？



# Filecoin的解决之道

## —— PoRep

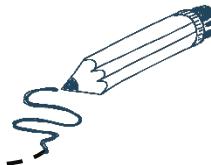


- ① 存储矿工不存储D，而是存储D的一个备份R（R是D一个长度相同的变换，且可逆）。
- ② 备份方法与数据编号和矿工编号绑定。
- ③ 让备份过程慢。
- ④ 备份后数据要满足以下要求：即便删除一部分之后，生成缺失的部分仍然很慢。

### 【这样做的结果】

每一个矿工确实各自存储了跟数据D长度（几乎）相同的一个备份R，且各个备份R各不相同

# 如何复制?



## 【使用的主要工具】

**Hash函数**: 使备份结果运算过程结果不可提前预测、接近随机

**VC (Vector comment)** : 用于承诺和揭示承诺

**DRG (Depth robust graph)** : 确保即便删除少量数据也要花费大量时间重新计算得到  
(同时也导致了备份慢)

**ZK-SNARK**: 保证证明、验证内容简洁且不透露真实数据的信息

## 【备选工具】

**VDE** (增加备份延迟)



# PoRep简略操作步骤



## 1 复制

根据filecoin给定的复制参数，数据编号，矿工编号和源数据，生成备份。

## 2 PoRep

根据filecoin给定的证明参数，给出复制证明。

# 接下来...



复制完并给出复制证明后，系统要求矿工持续存储备份，直到客户取回数据不再存储。



因此，系统每隔一段时间会要求矿工给出存储备份的证明——时空证明（PoSt）



矿工利用其存储数据使用的空间进行挖矿，成功概率与空间大小和（一定区间内）存储时间的乘积成正比。

感谢观看！

