以太 & 智能合同

--实现分散的未来

lemmas 1 和2

计算电力需要电力，需要美元，如果矿工们正在实现收支平衡或盈利，则需要达到平衡

- lemma 1: 采矿奖励 = 采矿成本

如果你在投资的资本上大致实现了收支平衡，那么获得更多的哈希利率几乎没有什么边际成本。你只需要更多的资本就能达到51%

- lemma 2: 收购51% 的成本 < 采矿成本

lemma 3

拥有 > 51% 的大麻, 你能得到什么利润?

- 崩溃的货币？没关系。通过在交易所做空比特币来恢复价值 (然后是一些)

你可以有效地获得100% 的采矿奖励

- 只有我在你自己的街区
  - 可以防止其他人开采-你总是生产最长的 pow 链

这将如何影响价格取决于阈值

- q = 51% = > 49% 的块是孤立的
- q = 80% = > 20% 的块是孤立的
  - 平均比特币用户没有真正受到影响, 仍然能够进行交易

lemma 3: 51% 攻击值 > 采矿奖励

结合莱姆斯

外秤:

- lemma 1: 采矿奖励 = 采矿成本
- lemma 2: 收购51% 的成本 < 采矿成本
- lemma 3: 51% 攻击值 > 采矿奖励

因此, 51% 攻击的价值 > 获取51% 的成本

如果数学是正确的, 博弈论说51% 的攻击比特币是有利可图的

其他想法

保险合同

- 抵消采矿池成本的一种方法
- 更多的孤儿块是池战争的标志
- 基于孤立块数量的比特币利益相关者保险合同

漏洞的泛化

比特币开采为零和

- 一般来说, 要想增加收入, 就需要把别人排除在外

仅限会员的采矿

- 让哈希利率加入串通, 直到 8 0% 的网络在, 然后将其余的排除在外
- 没有不加入的动机
  - 攻击成功, 获得更多的奖励
  - 攻击不会失败: 以这样的方式进行攻击, 在达到门槛之前不会开始

天真的例子

- 3个泳池勾结, 拥有51% 以上
- 忽略另一个池的每一个第10个块
- 检测不到

比诚实的策略更有利可图

块后奖励比特币

假设: 平均比特币用户持有 100, 000 美元比特币, 愿意支付1000美元的费用

- (这是当比特币接近0块奖励)
- 基于交易费用的采矿是否可持续?
- 钱必须流动, 必须以交易费用支付, 以便矿工能够收取, 作为采矿奖励
- 依赖采矿奖励进入比特币的哈希功率量

因此

- (平均支付的费用)/(持有的 avg) = (攻击的费用)/(市场上限)
- 在我们的示例中, 攻击者只需支付比特币市场上限的1% 即可攻击

事后奖励比特币必须有高速的钱才能安全

以太 & 智能合同: 促成分散的未来


ethereum
HOMESTEAD RELEASE
BLOCKCHAIN APP PLATFORM

以太

术语概述

布洛克链

可靠的

减少的应用

智能合同

概述

什么是以太?

- ethereum 是一个分散的平台, 运行智能合同。
- 以太是一个基于帐户的区块链。
- ethereum 是一种分布式状态机, 它依赖于事务在状态之间移动。

分散: 没有单点控制/故障;审查抵抗

区块链: 状态是由一系列的块构建的, 这些块由事务组成, 由*帐户*, + 网络共识

智能合同: 更复杂的脚本, 允许 (几乎) 任意计算

基于交易: 状态转换发生在新的事务上, 新事务之间传递价值和信息*帐户*

基于会计: 状态是由*帐户*, 与每个*帐户*有一个地址, 以太的一些平衡, 并可选择一些合同代码和存储

比特币的一些区别

- 虽然**以太和比特币**具有类似的功能, 两者的计费方式完全不同:
  - 以太:**智能合同平台**
  - 比特币:**分散的资产**
- 基于帐户而不是基于 utxo
- 以太有一个图灵完整 比比特币脚本强大得多的脚本语言. 启用智能合同。
- 在某些方面, 以太资产是拥有一个激励一致的智能合同平台的副作用。
- 以太计划在不久的将来转移到利益证明公司。

其他执行细节:

块创建时间: (~ 12秒 vs ~ 10分钟)

工作证明: (ethash vs sha256)

乙二酸 (目前) 是抗 asic

汇率: (2016-10-19 00:37 太平洋标准时间)

美元: 12.45 美元

btc us:6355.38

在可计算性理论中, 数据操作规则系统 (如计算机的指令集、编程语言或细胞自动机) 被称为图灵完整或计算通用, 如果它可以用来模拟任何图灵机。

账户与 utxo

回忆: 比特币用户的可用余额是他们拥有输出地址私钥的未用交易输出的总和。

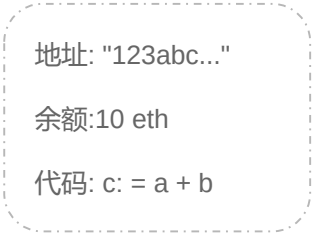
相反, 以太使用了一个不同的概念, 称为帐户。

比特币:

bob 拥有用于设置 utxo 的私钥

以太:

埃文拥有一个账户的私钥



外部拥有的账户 (eoa):

- 一般由某个外部实体拥有
- 由*地址*
- 保持一定的平衡*醚*(单位以太货币)
- 可以发送交易记录 (将以太转移到其他帐户, 触发合同代码)

合约帐目 (合约):

- 由*地址*
- 有一些*醚*平衡
- 具有关联的合同代码
- 代码执行由从其他合同接收到的事务或消息 (函数调用) 触发
- 合同具有持久存储

所有帐户 = = 网络状态

所有帐户的状态是 ethereum 网络的状态, 即*整个*网络在当前的平衡、存储状态、合同代码等方面是一致的。的***每个账户***.

网络状态将随每个块一起更新。

您可以将块视为状态转换函数;它采用以前的状态并产生一个新的网络状态, 每个节点都必须同意。

帐户通过交易记录与网络、其他帐户、其他合同和合同状态交互。

账户理由

节省空间: 只需更新每个帐户的余额, 而无需存储每个 utxo

最重要的是, 智能合同在有余额的账户之间进行转移时更直观地进行编程, 而不是不断更新 utxo 集, 以计算用户的可用余额。

帐户模型的一个弱点是, 为了防止重播攻击, 每个事务都必须有一个 "nonce"。

每个帐户都有一个附加的 nonce, 每次发送新事务时都会递增。如果交易记录的值为 1 + 帐户的 nonce, 则该交易记录有效。

这意味着, 即使是不再使用的账户, 也永远无法从账户状态中修剪!

重播攻击的示例:

1. 鲍勃从爱丽丝那里得到 5个 eth。
2. 鲍勃广播的正是同样的 txn.....。
3. 没有一个 nonce, 网络就会看到另一个完全有效的 txn!有了 nonce, bob 就不能修改新的 txn 的 nonce, 因为他没有访问 alice 的私钥。

智能合同-简介

con· trace

*(名词)*käntrakt/s

1. 书面或口头协议, 尤指关于就业、销售或租赁的协议, 旨在依法执行。

智能康-道

(名词)/smärt li käntrakte/

1. 便利、验证或强制谈判或执行数字合同的代码。

在 ethereum 的情况下, 智能合同只是一个带有代码的帐户。

以太中的合同就像生活在以太执行环境中的自主代理, 在交易或消息 "戳" 时总是执行特定的代码, 并直接控制自己的以太余额和自己的以太平衡永久状态。

智能合同在以太

ethereum 合同一般有四个目的:

- **存储和维护数据**, 表示对用户或其他合同有用的内容, 例如令牌货币或组织的成员资格。
- **管理合同**或多个通常不信任的用户之间的关系, 例如, 金融合同, 代管, 保险。
- **提供功能**作为软件库的其他合同。
- **作为外部拥有的帐户**有一个更复杂的访问策略, 也就是 "转发合同"。通常, 合同接收传入的消息, 并在满足某些条件的情况下将其转发到特定的目标, 例如, 只有在密钥持有人的 mof of 批准的情况下, 才转发消息的多签名合同。

或上述的一些组合!

```
合同菲利普令恩{
    /* 将帐户地址映射到令牌余额 */
映射 (地址=>uint256)公共平衡的; 平衡的
    /* 与初始供应开始合同
给合同创建者的令牌 */
    功能 菲利普令恩(uint256 初始供应)
{
        为创建者提供所有初始令牌
[msg. 发送方] 的平衡=初始供应;
}
    /* 将令牌发送到收件人地址 */
    功能 转移(地址到, uint256 值)
{
        如果([msg. 发送方] 的平衡<值)扔;检查发件人是否有足够的
        如果(平衡的 [到]+价值<平衡的 (到)扔;检查溢出
[msg. 发送方] 的平衡-=值;从发件人中减去
平衡的 [到]+=值;向收件人添加相同的
}
}
```

最小可驻令牌

菲利普令恩是数字令牌合约的剥离版本, 用 "稳定性" 编写。

您可以使用一些可以在不同帐户之间转移的令牌的初始供应来实例化它。

记得**合同是一个帐户!**它有自己的乙醚平衡, 地址, 存储等..。

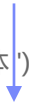
以太虚拟机

在每个节点上实际执行的 ethereum 协定代码是所谓的 evm 代码, 这是一种低级的、基于堆栈的字节代码语言。

每个 ethereum 节点都运行 evm, 作为其块验证过程的一部分。

evm 作为状态转换机制:

(块 _ 状态, 气体, 内存, 事务, 消息, 代码, 堆栈, pc)



(块 _ 状态 ', 气体 ')

其中块 _ 状态是包含所有帐户的全局状态, 包括余额和长期存储

evm 设计目标:

简单: 操作代码应尽可能处于低级。应尽量减少操作代码的数量。

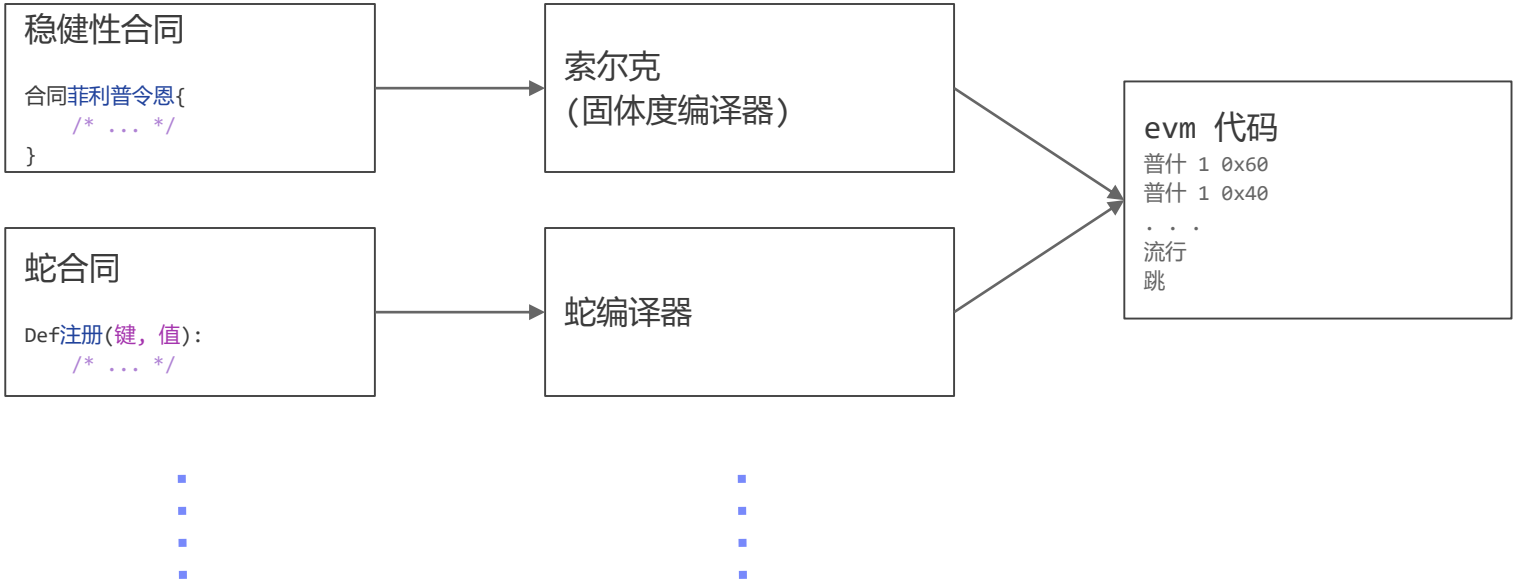
决定论: evm 代码的执行应该是确定性的;相同的输入状态应始终产生相同的输出状态。

空间效率: evm 组件应尽可能紧凑

专业化: 使用32字节值、自定义加密中使用的模块化算法、读取块和事务数据、与状态交互等, 轻松处理20字节地址和自定义加密

安全: 应该很容易想出一个气体成本模型来操作, 使 vm 不可利用

代码编译


稳健性合同
合同菲利普令恩{
/* ... */
}
蛇合同
Def注册(键，值):
/* ... */
索尔克
（固体度编译器）
蛇编译器
evm 代码
普什 1 0x60
普什 1 0x40
. . .
流行
跳

气体和费用

即时问题:

如果我们的合同有无限循环呢?

```
功能 Foo()
{
    而(真) {
        /* 永远循环!*/
}
}
```

网络上的每个节点都将永远被卡住执行循环!由于停止的问题, 它是不可能提前确定合同是否会终止拒绝服务攻击!

以太的解决方案:

每项合同都要求 "气体", "为合同的执行提供燃料"。

具体来说, 每个 evm 运算代码都需要一定数量的气体才能执行。

每个事务都指定斯塔特加斯, 或它愿意消耗的最大数量的气体, 和加斯价格, 或费用在以太它愿意支付单位气体。

气体和费用

在交易开始时,`startgas * 汽油价格`从发件人的帐户中减去以太。

如果合同成功执行, 使用的气体量低于预先指定的气体量, 剩余的气体将退还给发信人。

如果合同执行在完成之前就用完了气体, 那么执行就会恢复,`startgas * 汽油价格`不退款。

那无限循环呢?

以太仍然允许无限循环;然而, 无论是谁试图 dos, 网络都必须支付足够的以太来资助 dos。

把购买天然气看作是购买分布式计算能力。

以太智能合约

- 以太是***不***关于优化计算效率

- 它的并行处理是冗余并行的，为在不需要受信任的第三方的情况下就系统状态达成共识提供了一种有效的方法。

- 由于合同执行是冗余跨节点复制的，因此成本很高，这通常会产生一种激励，即不使用可以在链外完成的计算中使用区块链。

智能合同

用例和分析

令 牌

公共数据库

交叉

文件存储

市场

博客

自动共享经济

Daos

基本使用案例

令牌系统

- 很容易在以太中实现
- **具有一个操作的数据库**
  - 确保爱丽丝有足够的钱, 并确保她发起了交易
  - 从爱丽丝减去 x, 把 x 给鲍勃

示例 (来自以太白皮书):

```
def send(to, value):
    if self.storage[msg.sender] >= value:
        self.storage[msg.sender] = self.storage[msg.sender] - value
        self.storage[to] = self.storage[to] + value
```

公共登记处/公共数据库

示例: namecoin

- dns 系统
  - 将域名映射到 ip 地址
  - "maxfa. ng" = > "69.69.69.69"
- 变
- 易于在以太实现

示例 (来自以太白皮书):

```
def register(name, value):
    if !self.storage[name]:
        self.storage[name] = value
```

众筹与激励

简单示例: "以太上一根"

- 允许您在完成任意任务时提供赏金
- 贡献者将资金集中到一个智能合同中, 该合同支付给指定的接受者, 向贡献者投票, 认为任务确实已经完成

示例用例

- 一家公司正在污染当地的一条河流, 附近的居民承担着负面的外部性
  - 当地政府没有反应迟钝, 但居民愿意付钱
- 居民们集中资金, 激励公司清理 "

实施显性保证合同: 解决搭便车问题

高级用例

分散的文件存储

"分散的 dropbox 合同": 向个人支付少量以太, 以出租额外的硬盘空间

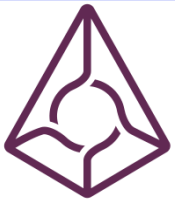
合同规范示例:

- 将猫的图片拆分为多个块, 加密每个块以保护隐私
- 从块创建 merkle 树, 在合同中保存 merkle 根
- 每 n 个街区, 合同将:
  - 使用前面的块标头 (随机性的来源), 在 merkle 树中选择一个随机块
  - 第一个实体提供块 (merkle 分公司) 的存储证明获得小 eth 奖励

恢复文件:

- 查询节点存储文件, 并支付少量费用 (通过微支付渠道) 检索文件
- > 解密数据, 获取毛茸茸的小猫
- > 利润

分散的预测市场



augur

预测市场利用群众的智慧预测未来

- 做市商创造活动
  - 例如: "谁将赢得2016年美国总统选举"
  - 事件必须是公开的, 并且易于验证, 并设定了截止日期。
- 参与者购买特朗普或希拉里的股票, 并支付少量费用
- 在选举日当天, 网络上随机的神谕会对谁获胜进行投票。
  - 与多数人投票的神谕收取一定的费用, 否则将受到处罚
- 正确投票的股东兑现了他们的赌注

每个市场的股价准确地代表了最好的
事件发生的预测概率

- 某人有额外的信息 = > 套利机会

GNOSIS

分散的预测市场

augur

使用案例

- **以经济高效的方式购买未来活动的信息**
  - 与其聘请专家和专家，不如为您的活动创造一个市场
  - "这部电影会失败吗"
  - 投注赞成和反对你的活动，以激励掌握这一活动信息的人 (在这种情况下，好莱坞内部人士)
- **套期保值和保险**
  - 火灾保险是一个赌注，你的房子会被烧毁
  - 创建市场 "我的房子会被烧毁吗？"
  - = > 如果你的房子被烧毁，请给予赔偿
  - 可以实施整个保险流动性池
  - 潜力极薄的利润，因为没有中央中介是必需的

GNOSIS

分散的预测市场

augur

使用案例

- **设置安全错误奖励**
  - "我的公司会被黑客攻击吗？"对它下注沉重，以创造一个财政激励
  - 发现漏洞的人会购买肯定股票，然后执行他们的黑客攻击
    - > 利润
  - 奥古尔以这种方式保护自己的代码
    - "有人能在这个预测市场上偷钱吗"
- **信号："把你的钱放在嘴边"**
  - 通过展示你对某事的承诺来展示你对事物的承诺
    如果你错过了你的承诺，你将遭受巨大的经济损失
  - 例如：踢球者运动;投资者担心你会推迟
    发射日期
    - "我的踢球者活动会准时启动吗"
    - 大量的赌注，你会推出你的产品的时间。

GNOSIS

分散的预测市场

augur

分散的好处

- 对市场创建没有限制
  - 但提出伦理问题
- 共享流动性池
  - 没有理由在多个国家存在相同的市场
  - 允许更先进的市场;
    例如组合预测市场
- 耐候性
- 自动、不可信任的付款

GNOSIS

分散的物联网

丝

- "基于区块链的分散物联网"
- "智能传感器的自组织网状网络"
- 适用于工业物联网应用

产品

- 10英里范围的传感器
- 电池寿命长达数年
- 无需互联网连接-使用网状网络

FILAMENT

使用的技术:

- tele哈希-端到端消息加密
- tmesh-自成无线网状网络
- 布洛克纳梅-专用设备发现
  - 使用比特币区块链 + 公证员来验证名称地址绑定的真实性
- 区块-智能合同和微交易

Exchange

Value can be exchanged between devices in the form of data, network access, currencies such as Bitcoin, compute cycles, contracts for ongoing service, trusted introductions to other devices, and more.

长丝是分散技术的一个很好的应用，特别是因为它强调弹性和可靠性。

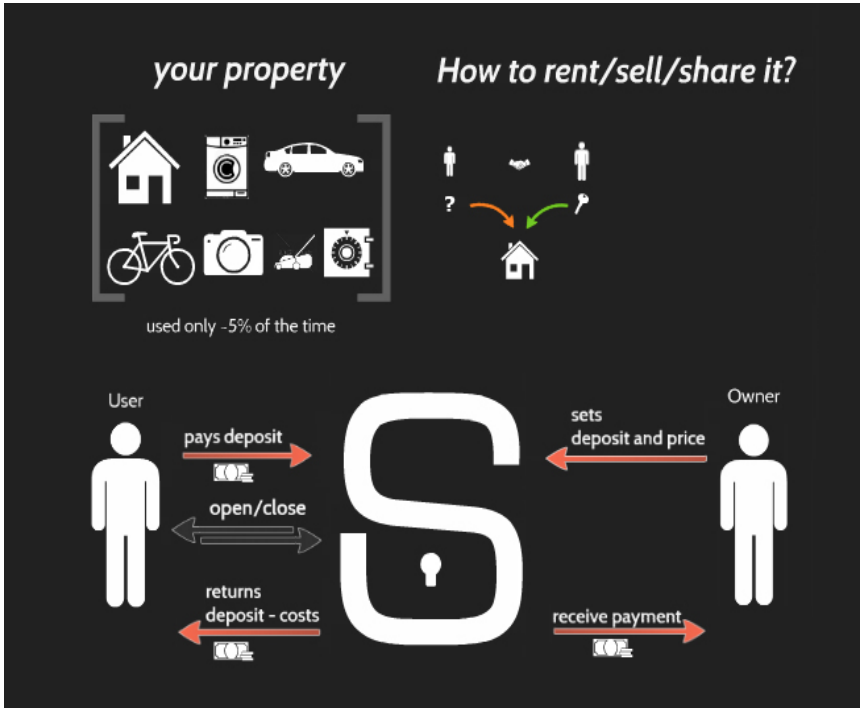
分散共享经济

连接器: 可以通过支付直接打开的锁

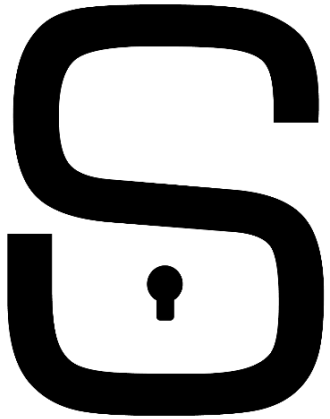
- 所有者设置存款 + 价格
- renter 支付存款 + 价格到锁连接到 ethereum 节点
- 锁定检测付款并解锁自身

用例 (s解锁. it):

- 全自动空中客车公司公寓
  - 不需要满足业主的关键
- 按需租用无线网络路由器
- 全自动商店
  - 通过将商品的价格发送给持有货物的锁购买商品
- 自动自行车租赁服务



分散共享经济



分散的好处 (s10:00. it):

- ...不信了吗?
  - 分散的声誉很难实施
  - 集中式解决方案可能更好
- 可编程货币
  - 仍然可编程的集中式解决方案
  - 公共区块链 (bitcincin/earum 等) 更容易实现技术整合
  - 进行交易不需要个人信息
- 物联网: 设备自主性
  - 设备可以独立于中央管理系统运行
  - 模块 化
- ???

分散的自治组织

分散的自治组织, 或**道**, 是完全由代码 (智能合同) 管理的组织

- 创建提案并对其进行表决
- 理论上, 企业可以完全存在于区块链上
  - 不确定的法律地位
- "代码就是法律"

问题:

- 部署后很难编辑管辖法律 (代码)
  - 可能的解决方案: 儿童 dao
- 不活跃的参与者: 没有足够的人对提案进行表决

破折号

- 以优先为中心的加密货币
- 采矿奖励的百分比

thdao

- 众筹基金
- 历史上最大的众筹项目
  - 在以太筹集 > 1.5亿美元
- 2016年6月被黑客攻击, 价值 6, 000万美元的以太被盗 (10% 的以太市场上限)

以太上的其他 dao:

- 斯特洛克。
- didix. io
  - 在以太区块链上存放黄金

概括

智能合同与区块链技术的局限性

没有不可信赖的方式访问外部数据

- 必须依靠神谕来提供来自封锁链外部的信息
  - 问题。。。神谕必须得到信任
- 潜在的解决方案: 经验证的执行 (不受信任的神谕)
  - 它有一个劣质的实现
    - tlsararary-修改 tls 协议, 以提供接收页的加密证明
- 潜在解决方案: oracle 网络对信息进行投票
  - 缺陷: 在协商一致议定书的基础上达成共识议定书
  - 难以协调激励/声誉

无法强制实施连锁支付

- 无法实施贷款和债券等金融产品
  - 资金必须存放在区块链上, 以确保付款
- 直觉: 部分由于违约风险, 我们支付贷款利息

合同无法操作机密数据

- 机密数据不能在其他人的计算机上组装
- 非常有限的访问控制功能
- 只能存储加密的数据并在本地解密
- 潜在解决方案: 同态加密

区块链杀手应用程序的基本属性

dapp 可以被认为是客户端软件--没有中央管理器

需要共识或协调的无信环境 (例如智能电网、能源市场)

以优先为中心的系统 (例如社交网络？)
- 虽然数据不应该存储在区块链本身

开放式集成的可编程货币
- 物联网、m2m 支付 (e. x. ibm adept)
- 易于发送和接收资金-无需个人信息
- 小额支付 (例如勇敢)

容错、有弹性的系统 (例如长丝)
- 自主网络和设备

创造激励措施的新闻方法 (如灵知)

新的治理模式 (例如 dao, 未来)

非中介, 耐审查

信任数学和代码, 而不是机构

与集中化的对比:

深度集成, 有凝聚力的用户体验
- 效率-区块链总体上是缓慢的
- 完全控制数据和读写权限

总是问: 为什么使用区块链比使用中央数据库更好?

社区、法规和争议

读数

- (维基)了解您的客户
  - https://www.wikiwand.com/en/Know_your_customer
- (维基)比特牌
  - https://www.wikiwand.com/en/BitLicense
- (文章)bit许可证2。0
  - http://www.coindesk.com/bitlicense-2-0-latest-revisions-mean-bitcoin-businesses/
- (文章)方块场辩论概述:
  - http://www.coindesk.com/making-sense-block-size-debate-bitcoin/

*选*阅读:

- 以太的当前用例列表 (中等文章)
  - https://medium.com//----------
- 消除共识计算机中的激励: 研究以太中的博弈论激励问题, 类似于我们上一次的讲座
  - https://eprint.iacr.org/2015/702.pdf

结束！

धन्यवाद
Hindi印地语

多謝
繁体中文

ขอบคุณ
泰语

Спасибо
俄语

谢谢
西班牙语

谢谢
英语

شكرا
阿拉伯语

奥布里加多
巴西葡萄牙语

多谢
简体中文

丹克
德语

格拉齐
意大利语

谢谢
法语

நன்றி
Tamil
泰米尔语

ありがとうございました
日语

감사합니다
朝鲜语

参考资料 (不完整)

- 以太白皮书
- 以太----------
  - https://github.com/phlip9/ether-on-a-stick
- 马丁·科佩尔曼的灵知使用案例介绍
  - http://www.slideshare.net/MartinKppelmann/gnosis-vision-and-crowdsale