**Mutual Chain**

**A High Performance Blockchain Framework for Mutual Financial Service**



**White Paper**

**June 2017**

|  |  |
| --- | --- |
| Author(s): |  |
| Version: | 0.7 |

AMENDMENT HISTORY

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Editor** | **Changes** |
| V0.1 | Jun. 25, 2017 | W. QIN | 1. Initial version |
| V0.2 | July 6, 2017 | Gavin Zheng  X. Tang | 1. Add Chapter9 about smart contract 2. Mutual Insurance logic diagram added |
| V0.3 | July 8, 2017 | Z. He | 1. Add blockchain values for insurance to section 2.1 2. 3 Insurance use cases |
| V0.4 | July 15, 2017 | W. QIN | 1. MChain Value Network, |
| V0.5 | July 22, 2017 | W. QIN  Gavin Zheng | 1. MChain Middleware |
| V0.6 | July 29, 2017 | W. QIN  Sean Tang | 1. Fintech Engine 2. Insurance Business Logic |
| V0.7 | July 30, 2017 | David Cao | Content Update |

* **CONTENTS**

1 概述 – 基于加密数字货币的互助金融解决方案 5

1.1 背景介绍和互助链的使命 5

1.2 互助区块链的构成与特征 5

1.3 互助区块链的范围和局限 7

2 Mutual Chain系统构架 8

2.1 Mutual Chain链式结构 8

2.2 Mutual Chain系统实现 – 数字资产计算理论设计原理 9

3 应用场景分析及流程架构 10

3.1 应用场景角色概述 10

10

3.1.1 Blockchain’s Value for Insurance 12

3.1.2 Blockchain Network Participants 12

3.2 应用场景一（互助保险） 12

3.3 应用场景3 14

3.4 应用场景4 14

4 互助链区块链浏览器与钱包 16

5 互助链区块链中间件构架 16

6 互助链跨链价值网关 17

7 互助链Fintech引擎 17

8 互助链智能合约逻辑 17

9 互助链数字资产与交易 17

10 互助链的智能合约平台 19

11 互助（Hand in hand）保险系统 20

11.1 为什么选择以太坊 20

11.2 智能合约平台 20

11.3 系统框架 21

11.4 互助币 22

11.5 人工智能模块 22

11.6 风险防范 23

12 互助链做为一个房屋按揭平台 25

12.1 Mutual Chain Mortgage Origination: Processes are timely, costly and can result in search errors, which generate transactional risk 25

25

12.2 Mutual Chain Mortgage Origination: Processes are timely, costly and can result in search errors, which generate transactional risk 25

13 互助链做为一个基于区块链技术的融资平台 26

14 互助链Muton币ICO及ICO之后的项目情况 26

14.1 互助币Muton（代币符号:MUC）的发售 26

14.2 互助币的锁定机制 26

14.3 互助币最低筹集资金和资金使用情况 26

15 互助链开发计划与预算 26

16 互助链团队 26

17 附录 27

17.1 项目组织结构 27

17.1.1 核心团队 27

17.1.2 角色与责任 27

17.2 主要里程碑Key Milestone 27

# 概述 – 基于加密数字货币的互助金融解决方案

互助链（Mutual Chain）是一个基于加密数字货币的互助金融技术构架和完整的（金融）产品及服务的解决方案， 是全球独创作为分布式数据存储、点对点传输、共识机制、加密算法等技术

的集成金融应用，旨在打造一个全球化分布式网络自主，共赢，共享的社会化运转机制。虽然区块链技术还处于发展的初期，但其更加透明、开放的理念和机制已经展现出无比的生命力，同时以比特币、以太坊为代表的数字货币的出现正在以燎原之势震动着传统的货币体系和金融体系。互助链正是这一基础上，建立一个分布式的未来个人金融中心，同时逐步发展成为一个分布式的数字资产金融基础设施网络。

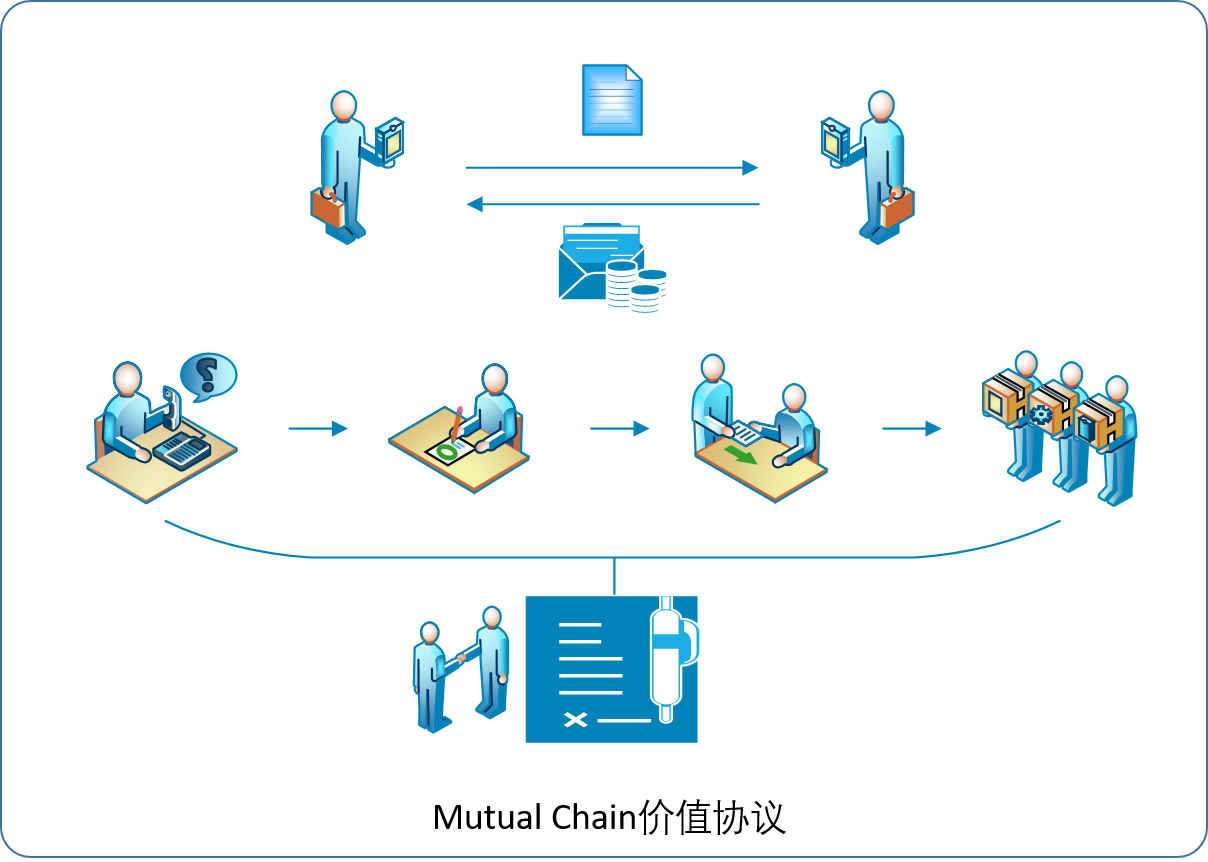
## 背景介绍和互助链的使命

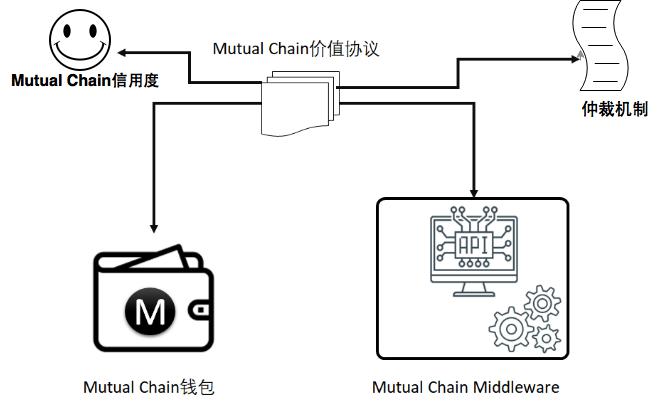
互助链（Mutual Chain）的使命是是每一个人都能轻松的启动和支持创建全新的基于数字货币的新金融服务和产品的项目。通过互助链，任何机构和个人，都可以在未来互助链中开设自己的业务窗口，拥有自己的数字钱包和自己的互助链信用体系，同时互助链提供特有的仲裁机制，一旦产生争议，仲裁机制将对交易内容和历史进行分析，提供仲裁依据和帮助。通过互助链提供的基于区块链的商业工具，现代社会中的商业模式和金融服务可以更加快捷高效的融入互助链引擎中，并且建立各自的金融服务区，使得更多人能够享受更加丰富的基于数字资产的金融服务， 比如保险，个人信贷等等。更加准确的描述，互助链是一个基于区块链的分布式超级金融产品及服务市场。

## 互助区块链的构成与特征

互助链（Mutual Chain）由互助链价值协议与互助链价值网络组成.

互助链价值协议包括钱包，信用度和仲裁机制。互助链钱包提供给个人或者机构存贷、兑换、支付、结算等服务，使得互助链链中的金融产品可以通过钱包进行交易和服务。 同时互助链信用度通过交易和链中行为积累，维持整个互助链交易信用体系。仲裁机制监督所有交易和执行，并对交易历史和数据进行分析，产生仲裁依据。





互助链价值网络由金融服务区和商业逻辑组成，通过区块链底层协议将互助链参与的一切交易数据真实源记录通过验证数据分享的开源机制写入到各个节点中,让真实数据通过智能合约变得更加简单而可靠,让每个人在商业活动中都能成为自己的智能合约执行者和数据分享者,并通过共同数据验证机制保证交易的安全和可靠性, 我们通过区块链的共享和写入机制,形成一个平行于现实世界的可调用,可验证的真实区块链金融网络.



在技术实现上，区块链的底层协议将保证数据的公正，安全和可靠。

狭义来讲, 区块链是一种按照时间顺序将数据区块以顺序相连的方式组合成的一

种链式数据结构, 并以密码学方式保证的不可篡改和不可伪造的分布式账

本。广义来讲，区块链技术是利用块链式数据结构来验证与存储数据、利

用分布式节点共识算法来生成和更新数据、利用密码学的方式保证数据传

输和访问的安全、利用由自动化脚本代码组成的智能合约来编程和操作数

据的一种全新的分布式基础架构与计算范式。它具有以下特性，

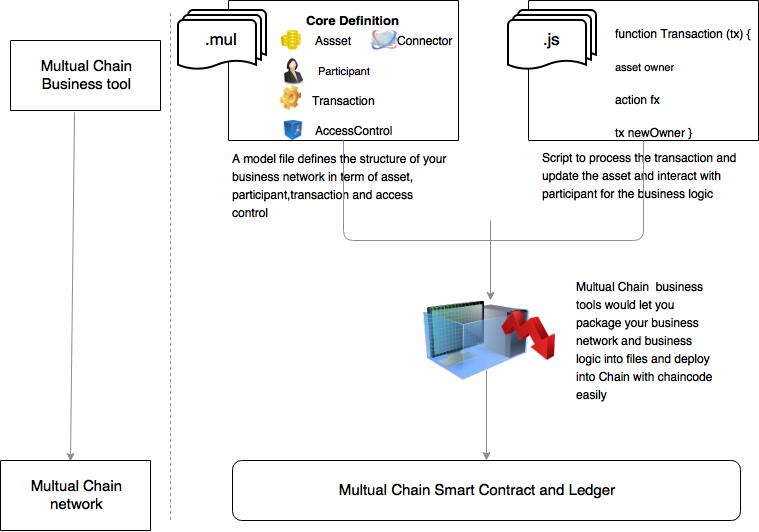
1. Broadcast，以区块为单位的链状数据块结构：区块链系统各节点通过一定的共识机制选取具有打包交易权限的区块节点，该节点需要将新区块的前一个区块的哈希值、当前时间戳、一段时间内发生的有效交易及其梅克尔树根值等内容打包成一个区块，向全网广播
2. 全网共享账本（SharedLedger）：在典型的区块链网络中，每一个节点都能够存储全网发生的历史交易记录的完整、一致账本，即对个别节点的账本数据的篡改、攻击不会影响全网总账的安全性。此外，由于全网的节点是通过点对点的方式连接起来的，没有单一的中心化服务器，因此不存在单一的攻击入口。同时，全网共享账本这个特性也使得防止双重支付成为现实。
3. 源代码开源：区块链网络中设定的共识机制、规则等都可以通过一致的、开源的源代码进行验证。
4. 非对称加密：典型的区块链网络中，账户体系由非对称加密算法下的公钥和私钥组成，若没有私钥则无法使用对应公钥中的资产。
5. 智能合约 （SmartContract）：区块链系统中的应用，是已编码的、可自动运行的业务逻辑，通常有自己的代币和专用开发语言。
6. DAPP：包含用户界面的应用，包括但不限于各种加密货币，如以太坊钱包。互助连将发行自己的代币用于链中的金融交易。
7. 虚拟机：用于执行智能合约编译后的代码。虚拟机是图灵完备的。

# Mutual Chain系统构架

## Mutual Chain的商业工具

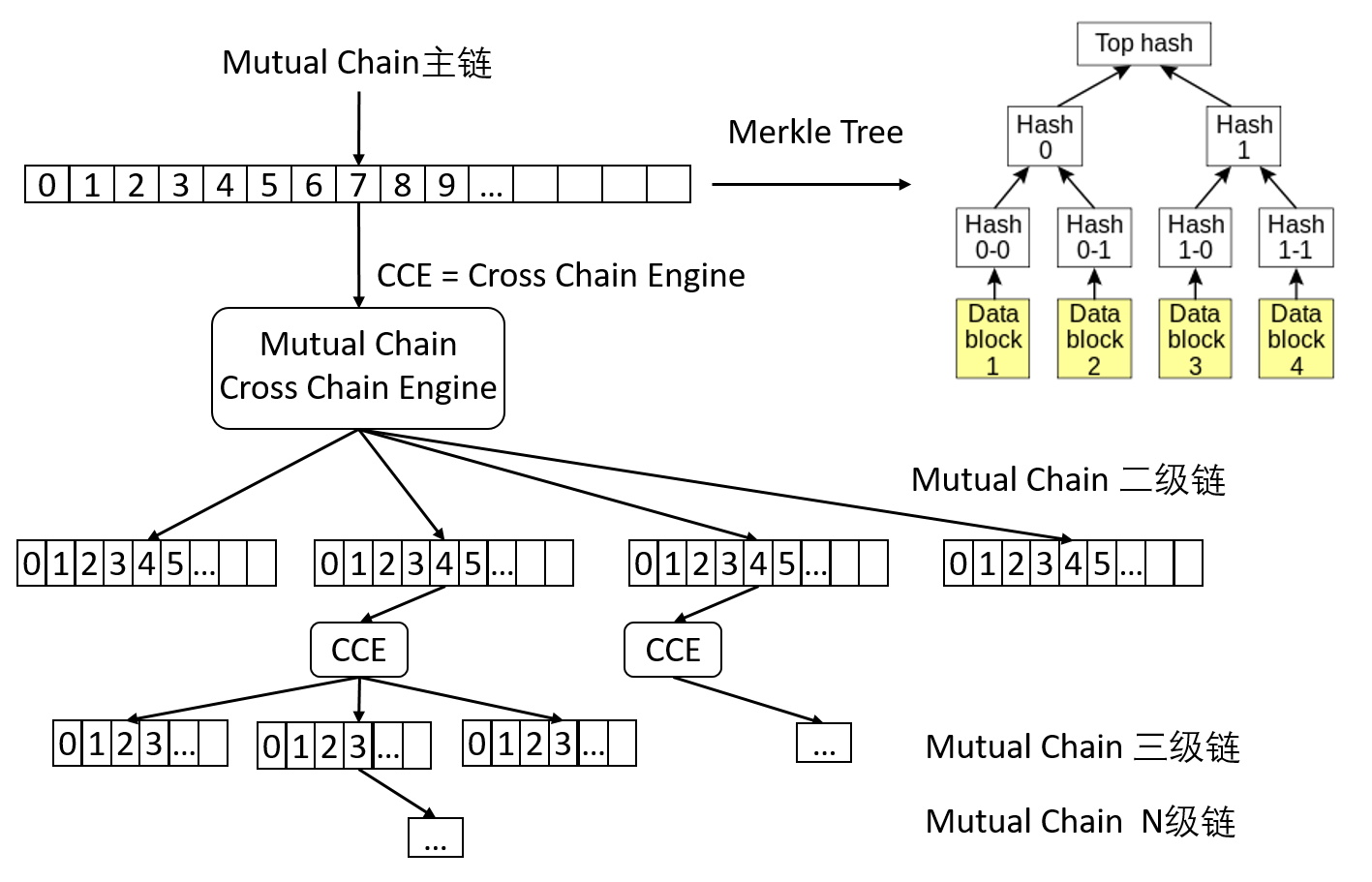
互助链将开发特有的商业工具, 帮助现代商业模式和金融应用融入到链中建立自己的商业金融区。工具包括：

1. Asset定义： 资产定义包括互助链流通资产。对于保险业来说，保单，保额以及理赔都是链中流通资产。
2. Participant定义：包括金融交易中的参与方，对于保险业来说，可以定义投保人，理赔，合作伙伴，以及基金管理和联保单位的各自属性。
3. Access Control: 定义参与方的权限和访问内容。
4. Transaction: 包括智能合约和Asset流动交易的逻辑， 这是各个商业金融网络在互助连中的核心交易流程以及保证，以智能合约的形式在互助链中执行。
5. Connector：这是各个金融商业区的企业级整合工具，通过connector可以实现链数据和现有中心商业网络的互联和整合，建立一个区块链和中心金融服务区的Hybird应用。



## Mutual Chain链式结构

* Mutual Chain是一个松紧耦合可配置的复杂多级链式结构，
* 用户和系统可根据业务场景需要再部署钱进行调配(configurable)
* 多级链式结构间通过跨链引擎来协调与复制/同步
* 链间连接通过加密HASH函数保障安全性，可扩展性和可访问性

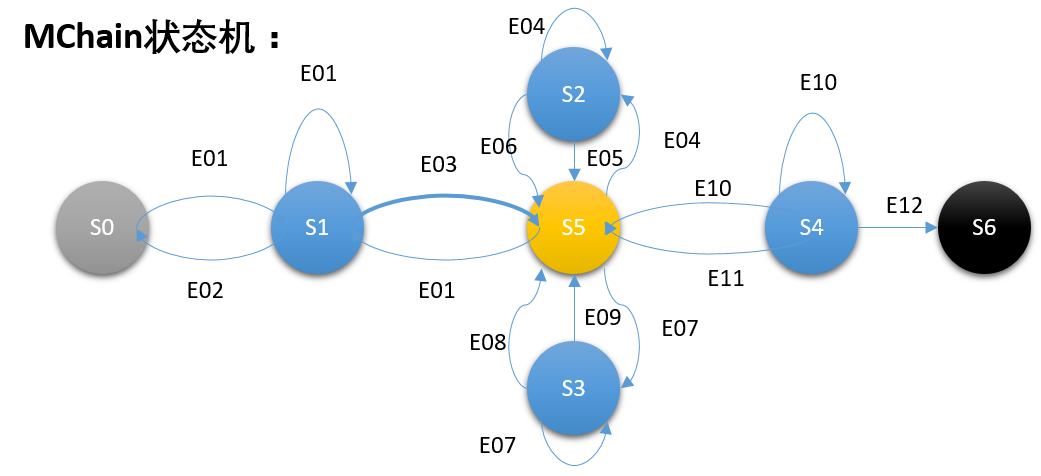


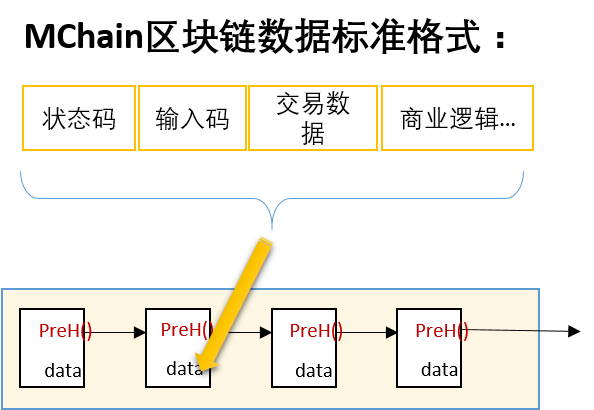
## Mutual Chain系统实现 – 数字资产计算理论设计原理

（Mutual Chain状态机）确定型有限状态机原理理论描述：

Mutual Chain上链信息是一个确定型有限状态机**M=(Σ,S,s0,δ,F)** - Deterministic Finite State Machine(DFSM)

* Σ是输入字母表 (有限，非空） - 动作
* S是有限非空状态 - 状态(目前暂定5个状态）
* s0是初始状态 S的一个子集
* δ是状态转换函数： δ：S x Σ -> S (δ必须返回一个属于S的状态s)
* F是一组最终状态集，是（可为空）S的一个子集



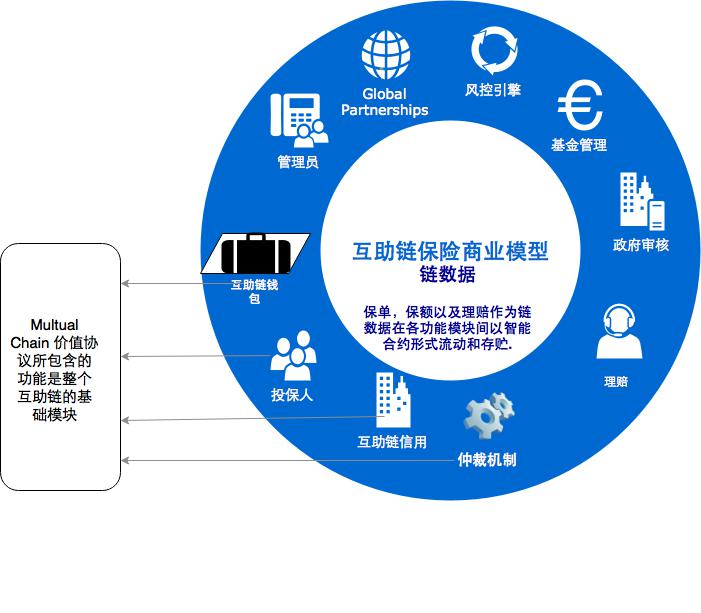
****

**MChain状态转换函数（区块链上链数据结构和流程逻辑数据标准基础）：**



# 应用场景分析及流程架构

## Insurance应用场景角色概述



Blockchain’s value for Insurance

External Reporting

Claims

Post Bind

Accounting & Settlement

Placing

## 

***Blockchain has the potential to drive simplicity and efficiency through the establishment of new financial services infrastructure and processes across all of the key aspects of the insurance value chain***.

Operational simplification: Reduces manual efforts to reconcile and resolve disputes;

Counterparty risk reduction: Agreements codified and executed in a shared, immutable environment;

Liquidity and capital improvement: Can reduce locked-in capital and provides increased transparency;

Regulatory efficiency improvement: Enables real-time monitoring;

Clearing and settlement time reduction: May reduce some third parties supporting transaction verification / validation and accelerates settlement;

Fraud minimization: Helps with asset provenance and transaction history within a single source;

**Insurance Administration:**

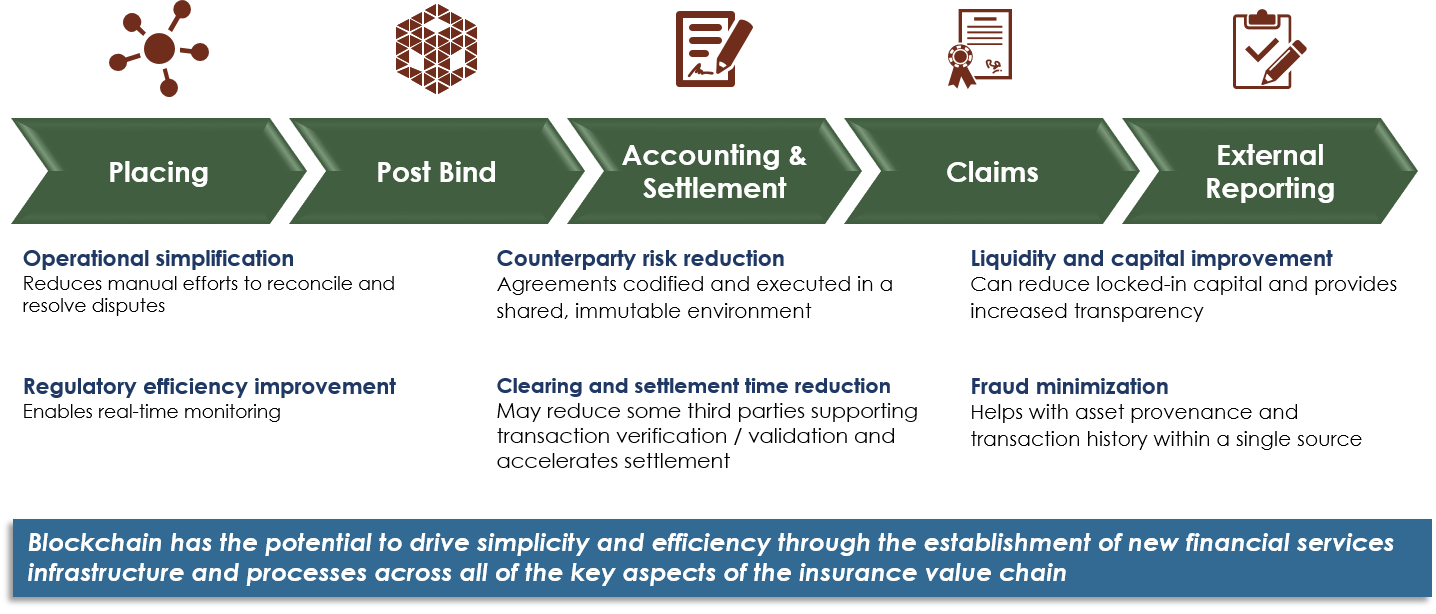
Today:

1. Manual processes between multiple parties (Insurance provider, benefits administrator, employer, employee, service provider network, etc)
2. Protracted coordination of business transactions between business network participants
3. Different versions of the same data in multiple systems of record, leading to added costs and lengthy disputes
4. Error prone verification processes for eligibility and benefits

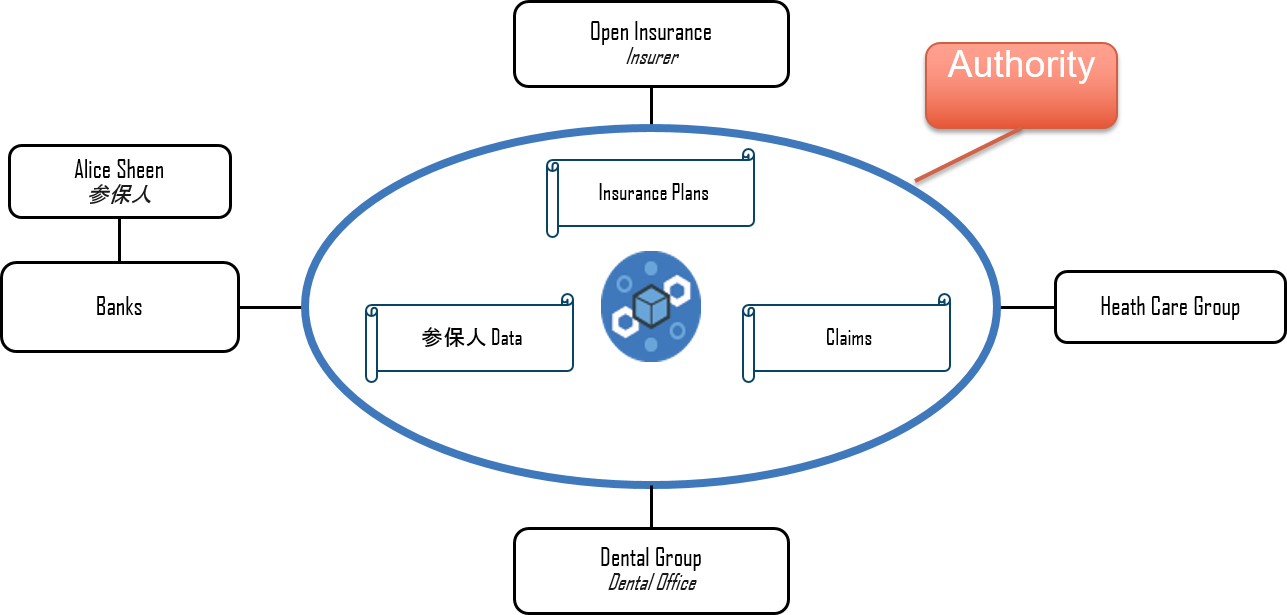
With Blockchain:

1. Manual process are automated by sharing plan participant, entitlement and claim data
2. Smart contracts computationally encode participant registration and verification rules
3. Shared data and smart contract computations reduce downstream disputes
4. Benefits verification based on multiple data sources readily available to all service providers

### Blockchain’s Value for Insurance



### Blockchain Network Participants



## 应用场景一（互助保险）

Client registers on mutual insurance exchange and create own wallet

* verify personal information
* link bank account to exchange
* setup private key and wallet for policyholder and beneficiary
* System maintenance cost and blockchain gas
* Investments and reinsurance

Client buys ETH or BTC coins from exchange

* Exchange prepares quote based on client request on coverage
* Smart Contract is initialized between each policyholders to mutually cover each other

Client shops on life insurance coverage on exchange and quote premium in mutual insurance coin

* Smart contract distributes new coverage coins equally to each current policyholders
* Each current policyholder gets new coverage coins portion and agrees to lock it until coverage expires or payout as part of death claims

Client buys mutual insurance coin (smart contract) to engage with current mutual insurance policyholders

Smart contract executes based on survivorship certificate and expires coverage between policyholders and unlock mutual insurance coverage

Blockchain broadcast event

Smart contract executes based on death certificate and mutual policyholders pays mutual insurance coin to beneficiary wallet

Plan Participant Onboarding ( Insurance Sales): Interactions between Plan Sponsor and Plan Administrator to on-board Plan Participants is often cumbersome and error prone

Role of Blockchain:

* Records required plan participant information directly from Plan Sponsor’s HR systems
* Makes data available to all blockchain participants on a real-time basis reduces possibility of dispute
* Improves client satisfaction and reduces costs

## 应用场景3

Coverage Verification:

Lack of ready availability can:

- Delay provision of coverage upfront

- Involve cumbersome manual discussions

- Require upfront payment despite coverage

Role of Blockchain:

* Provides immediate access to coverage information and current plan participant status
* Smart contracts verify plan participant coverage immediately, and avoid creation of payment disputes after the appointment occurs

## 应用场景4

Claim Submission:

Initial submission and filing process can:

- Allow erroneous submissions

- Result in dispute resolution

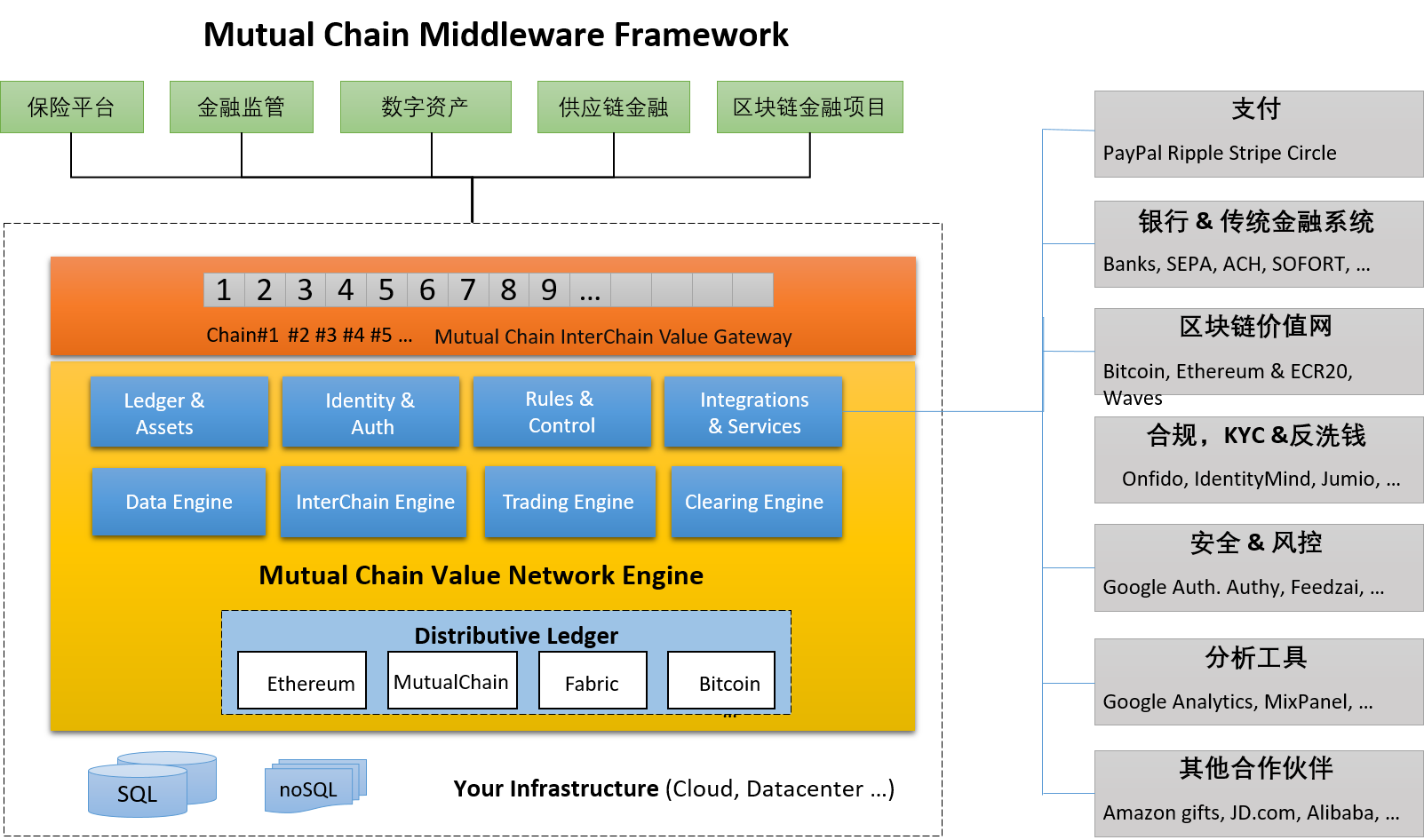
- Delay payment times

Role of Blockchain:

* Provides smart contracts that capture key contractual elements of the underlying policy and flag claim errors before submission
* Gives all permissioned parties access to claim data
* Provides immediate access to more relevant data and reduces the time required to make business decisions

# 互助链区块链浏览器与钱包

# 互助链区块链中间件构架



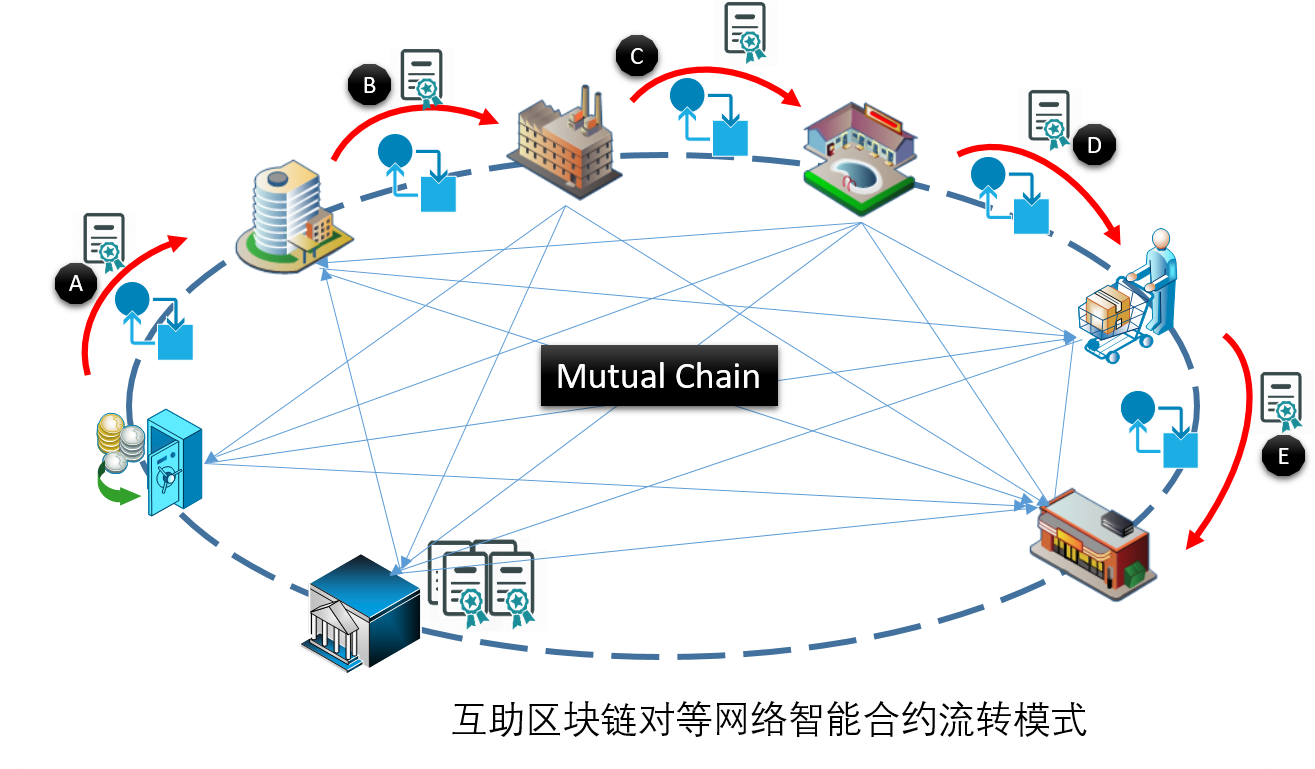
# 互助链跨链价值网关

**互助链跨链价值网关** - 是一个可以跟多个不同技术架构的异构价值网络进行通讯并能将资产数据进行自由转换的，可扩展，开放的软件接口。

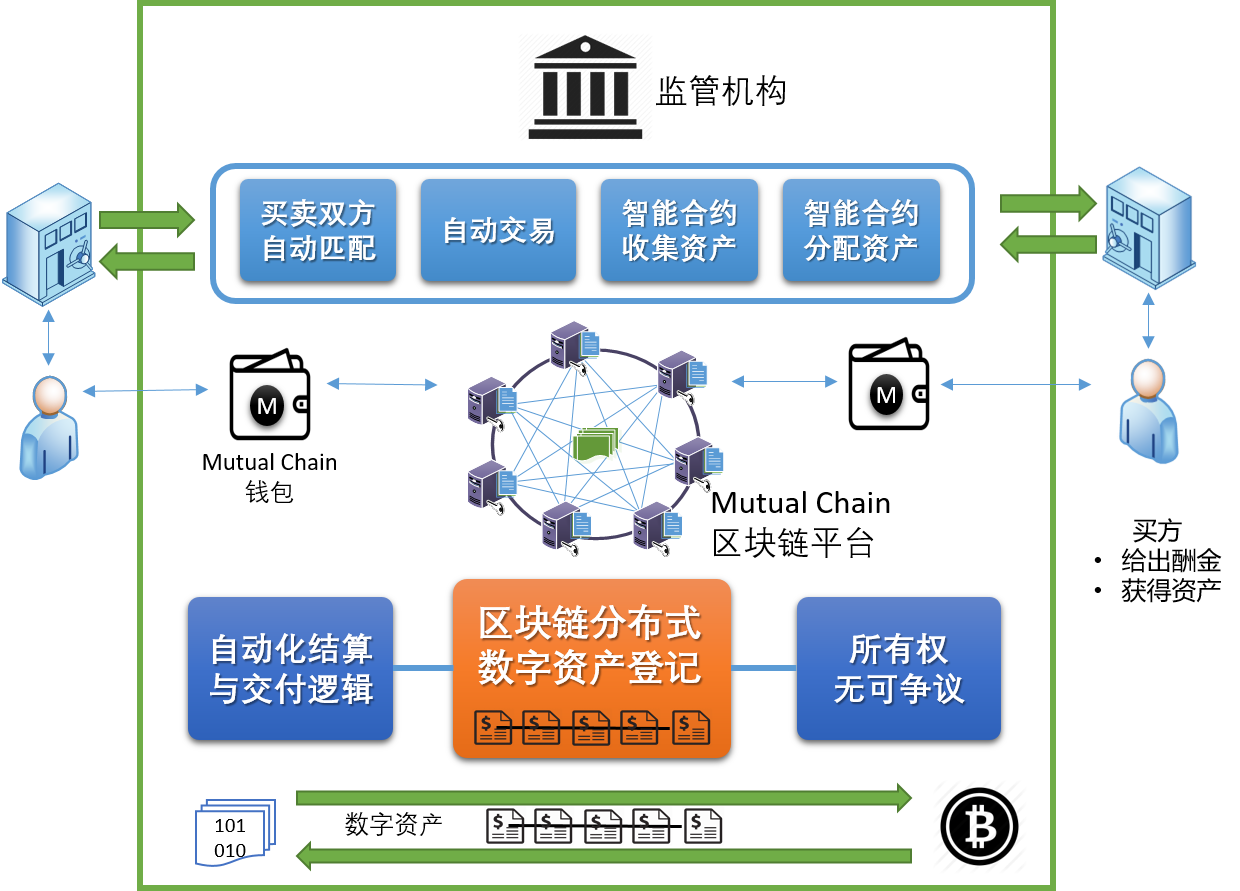
# 互助链Fintech引擎

**互助链金融科技引擎** - 包括未来可能的区块链数据读写交换引擎，资产交易，供应链金融逻辑，资产结算和清算引擎等。

# 互助链智能合约逻辑



# 互助链数字资产与交易



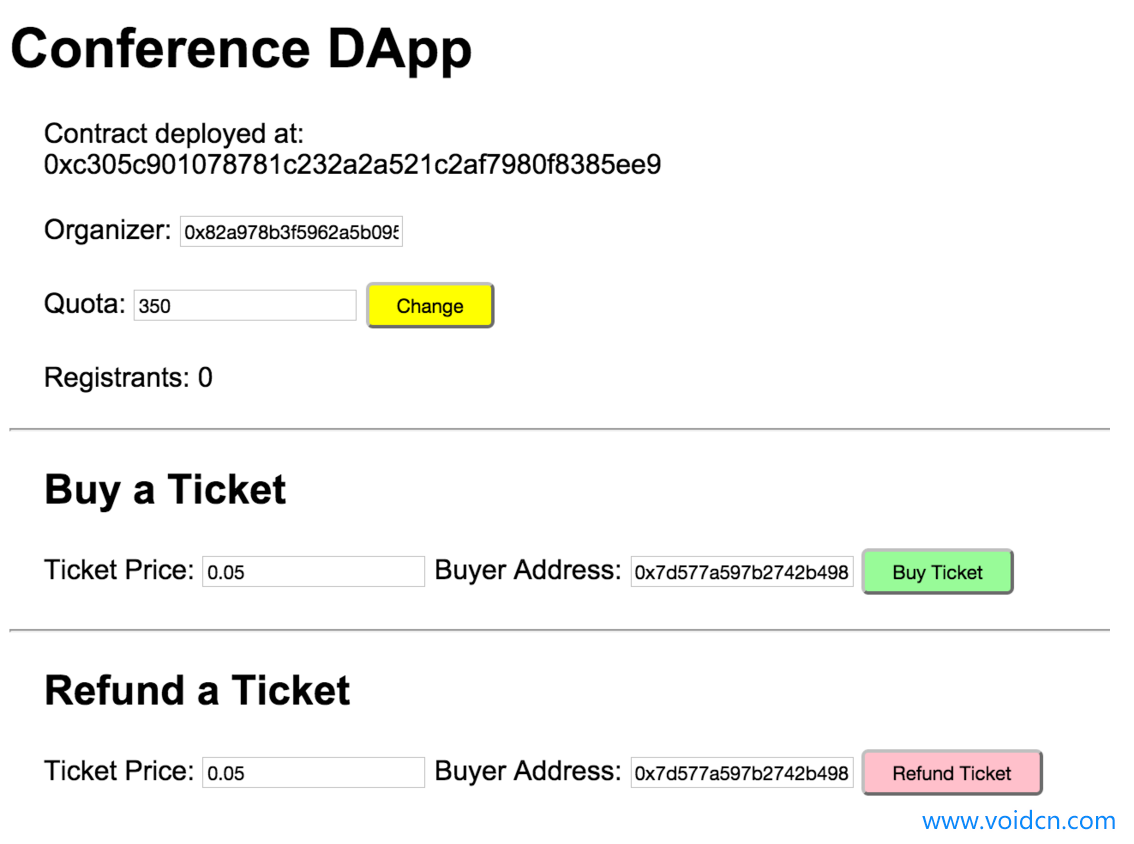
* 分布式交易账本
  + 第一次将资产完全数字化
* 数据更安全更具公信力
  + 完全透明公开的交易数据和交易历史，不可篡改，任何人（包括交易所平台的监管和运营）都不可以挑战数据的
* 交易根据智能合约自动匹配，更公平公正高效
* 交易资产上链存证，产权和所有权管理更方便
* 结算清算交割自动化与智能合约化，降低交付成本甚至消除大量繁琐的交易前中后台工作
* 消除交易双方对手风控
* 担保保险融资和政府监管更透明高效

# 互助链的智能合约平台

互助链（Mutual Chain）是类似于以太坊的区块链结构，用户的担保，承保，赔付，收益都以智能合约的形式写入底层区块链中。这样做的好处在于：

1. 去除中间环节，降低成本
2. 合约强制执行，不需要人工干预
3. 资金的使用，流向公开，透明

以太坊上要写智能合约有好几种语言可选：有点类似Javascript的Solidity, 文件扩展名是.sol. 和Python接近的Serpent, 文件名以.se结尾。还有类似Lisp的LLL。我们为互助链选择了Solidity。DApps的打包工具我们选择Truffle。



# 互助（Hand in hand）保险系统

## 为什么选择以太坊

也许出于安全性的考虑，比特币所提供的脚本语言并不是图灵完备的，能力有限，并不能完成现实世界里的大多数计算问题。同时比特币网络本质上是一个状态的转移系统，在比特币系统里，有一个包含现在所有已存在的比特币的持有者的状态，并且有一个“状态转移函数”可以使用一个状态和一个交易来产生一个新的状态。且比特币系统只有UTXO概念，对某个地址的查询效率低下。以太坊的概念是在2013至2014年间由Vitalik Buterin提出，其目的是研发出 “下一代加密货币与去中心化应用平台”，并在2014年通过ICO而得以开始发展。实现整个以太坊系统的是开源代码，其核心是基于专有加密货币的以太币（Ether）的公共区块链平台。在这个平台之上，以太坊提供了去中心化的以太坊虚拟机，方便其他第三方研发人员来实现基于以太坊的智能合约。以太坊虚拟机使得第三方研发人员能够简单快捷地重复利用以太坊所定义的区快链核心系统，同时能够把自身的研发重心放在区块链的应用上。

相较于大多数其他区块链技术，以太坊平台提供能账户，虚拟机和智能合约。大大提高了开发DApp的效率。

## 智能合约平台

支持智能合约体系的关键部分是以太坊虚拟机（Ethereum Virtual Machine ），它提供了一套类似汇编语言的图灵完备语言，可以执行任意复杂算法的编码，用来桥接数据处理和应用处理。互助链是类似于以太坊的区块链结构，用户的担保，承保，赔付，收益都以智能合约的形式写入底层区块链中。这样做的好处在于：

* 去除中间环节，降低成本
* 合约强制执行，不需要人工干预
* 资金的使用，流向公开，透明

以太坊上要写智能合约有好几种语言可选：有点类似Javascript的Solidity, 文件扩展名是.sol. 和Python接近的Serpent, 文件名以.se结尾。还有类似Lisp的LLL。我们为互助链选择了Solidity。DApps的打包工具我们选择Truffle。基于以太坊虚拟机以及Solidity编程语言，我们就可以开发出以太坊智能合约。简单来说，每一个合约都是一个基于以太坊区块链系统的分布式应用程序。当研发人员将一个开发出的合约在以太坊平台上上线后，这个合约就类似于一个活在以太坊平台里的应用代理人，它拥有自己的以太币地址，可以用来进行以太币交易、可以和创建者以及其他使用者交互信息、也可以用来执行具有复杂逻辑性的应用，譬如建立一套基于以太坊的代币系统、发起一次ICO活动以及建立一个去中心化自治组等等

## 系统框架

由于区块链具有完全公开、高可靠性、即时交割、去信任等诸多优点，我们设计了一个基于区块链的互助保险系统。它主要由4层架构：网页/移动客户端，基于区块链的业务逻辑管理模块以及数据存储模块。

**移动端**

**Web 端**

**客户端**

**业务端**

**AI**

**身份认证**

**资金池管理**

**精算**

**智能合约n**

**区块链**

**智能合约1**

**智能合约k**

**担保数据库**

**数据端**

**承保数据库**

**保险合同**

## 互助币

互助币是基于以太坊ERC20协议开发出的代币，这样可以使得互助币在ICO结束后很方便地上线全球重要数字货币交易平台。同时也支持转账和查询余额**。**互助币被用来构建保险系统的资金池，用户可以充值兑换成互助币，利用互助币来支付保险费用，各种手续费用或者赔付费用。在用户选择离开互助保险平台时，用户可以提取与其互助币价值相当的真实货币。随着互助保险生态的丰富，互助币应该会承担越来越丰富的角色：互助币不仅能支付，理赔，而且应该能在重要的数字货币平台上和其他数字货币进行交易，具有投资属性。另外，互助币也会在ICO和众筹中扮演重要角色。

## 人工智能模块

传统的保险行业沿用的是工业时代的规则，先圈定目标人群，然后为不同的目标人群设计不同的保险品种，注重规模化效应，所有投保人都被置于同一风险水平之上，而完全忽视了人群中的个性化差异。进入互联网时代以后，个性化差异问题越来越突出。功能的定制化和精准营销成为潮流。保险行业也从名目繁多的保险种类，发展到针对个人需求定制保险以及自主发现新的保险种类。以前，由于计算机技术手段的限制，扑捉个性化差异并推出相应的保险产品，人类有些力不从心。而随着大数据时代的到来以及计算机算力的提高，近年来人工智能取得了极大的突破。这就为保险行业的颠覆性改变提供了契机。

互助保险系统的人工智能模块通过运用人工智能最新的研究成果，实现精确的风控模型和定价策略，通过对个体的基于大数据的智能分析，实现：

* 个性化保险设计
* 精准化推销
* 自主发现新的保险品种
* 精确地理赔

## 风险防范

2017年7月20号的以太坊钱包漏洞让以太币价格大跌，让数字货币持有人心惊胆战，也让投资人的信心动摇。但是正如Vitalik针对此次漏洞的回应，这次事件不是以太坊的漏洞，它只是一个编码漏洞而已，所以也不需要硬分叉。此次事件是由于智能合约编程那个语言Solidity的问题。在互助币的发行和运营过程中，我们可以采用以下措施，尽量提高安全性，以保护用户的投资以及互助保险系统的平稳运行：

* 由于互助币是建立在联盟链的基础上，我们可以将智能合约的代码不公开。仅仅对于经过资格审核的伙伴公开
* 互相做严格的code review
* 采用安全的开发流程

单元测试

私有链测试

测试链测试

* 功能解耦

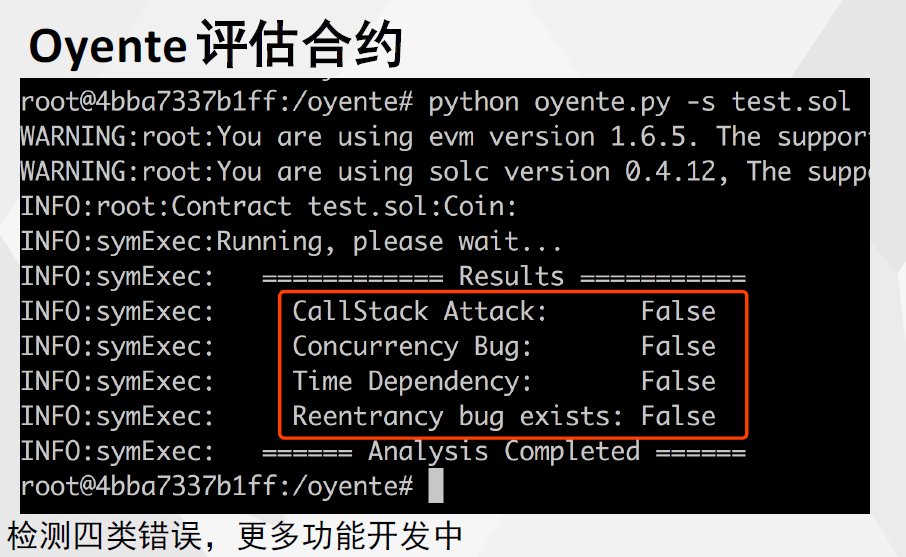
正交分解

逻辑简单

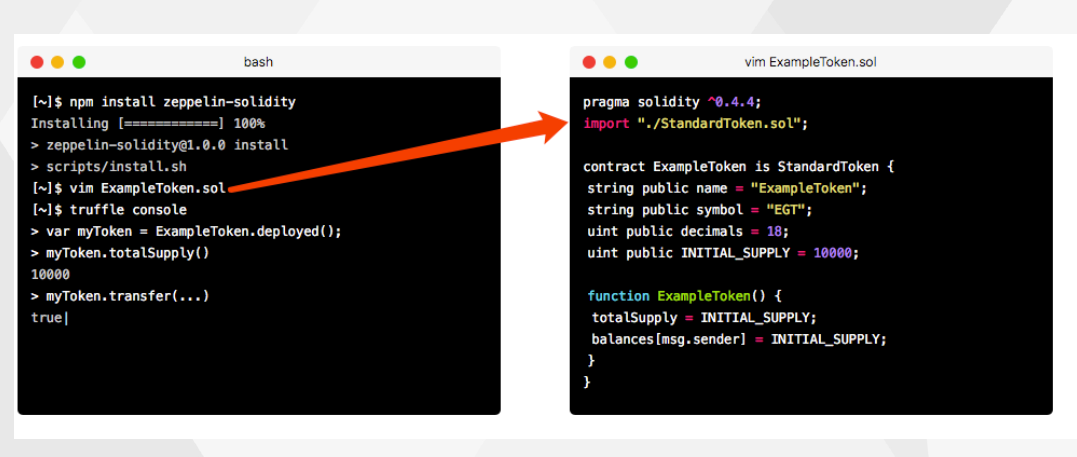
只对最核心的部分去中心化

* 安全相关的工具

Oyente



Zeppelin



Zeppeline提供了更安全的组件：

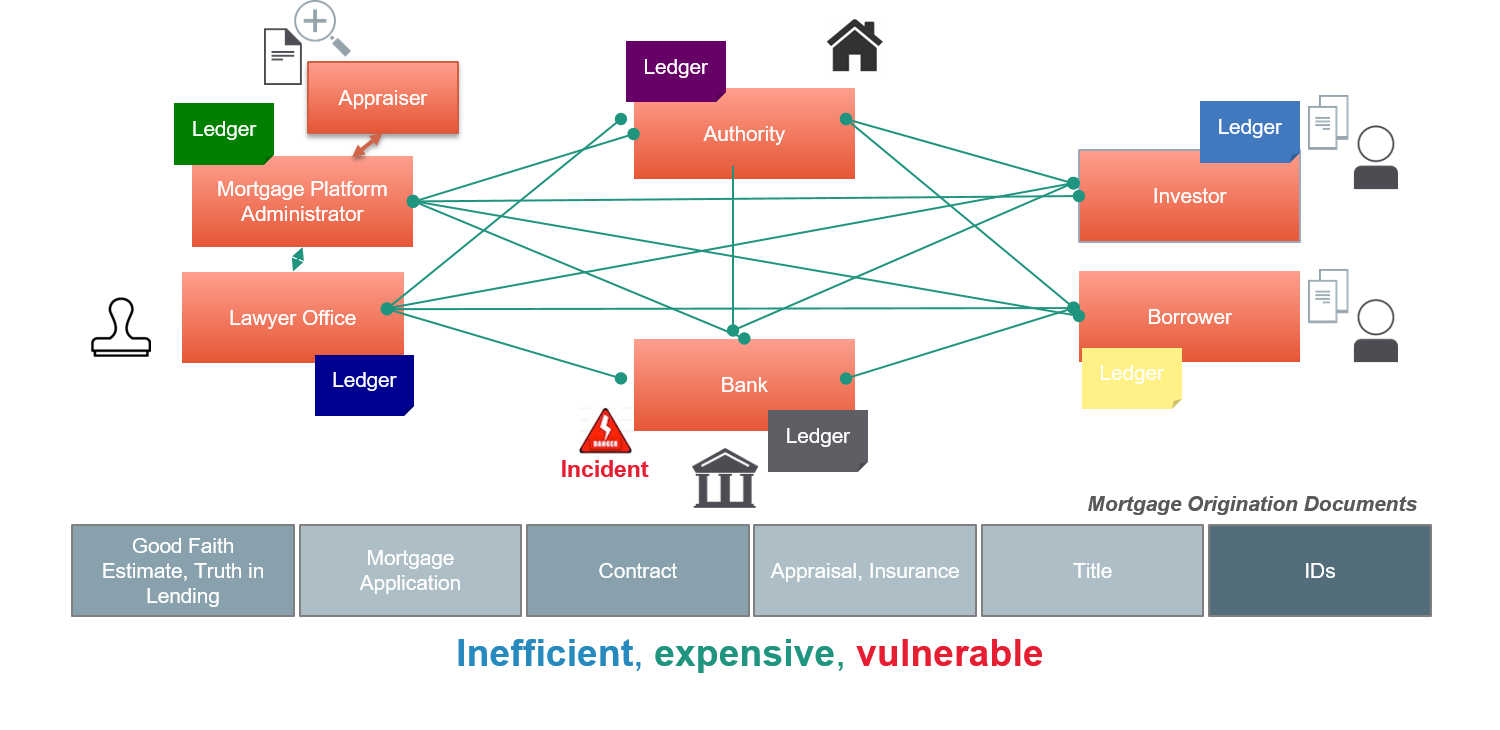
* + 安全数值运算，防止溢出
  + 所有权管理
  + 支付，余额以及限制每日交易数量
  + 多重签名
  + Token抽象类和安全实践

# 互助链做为一个房屋按揭平台

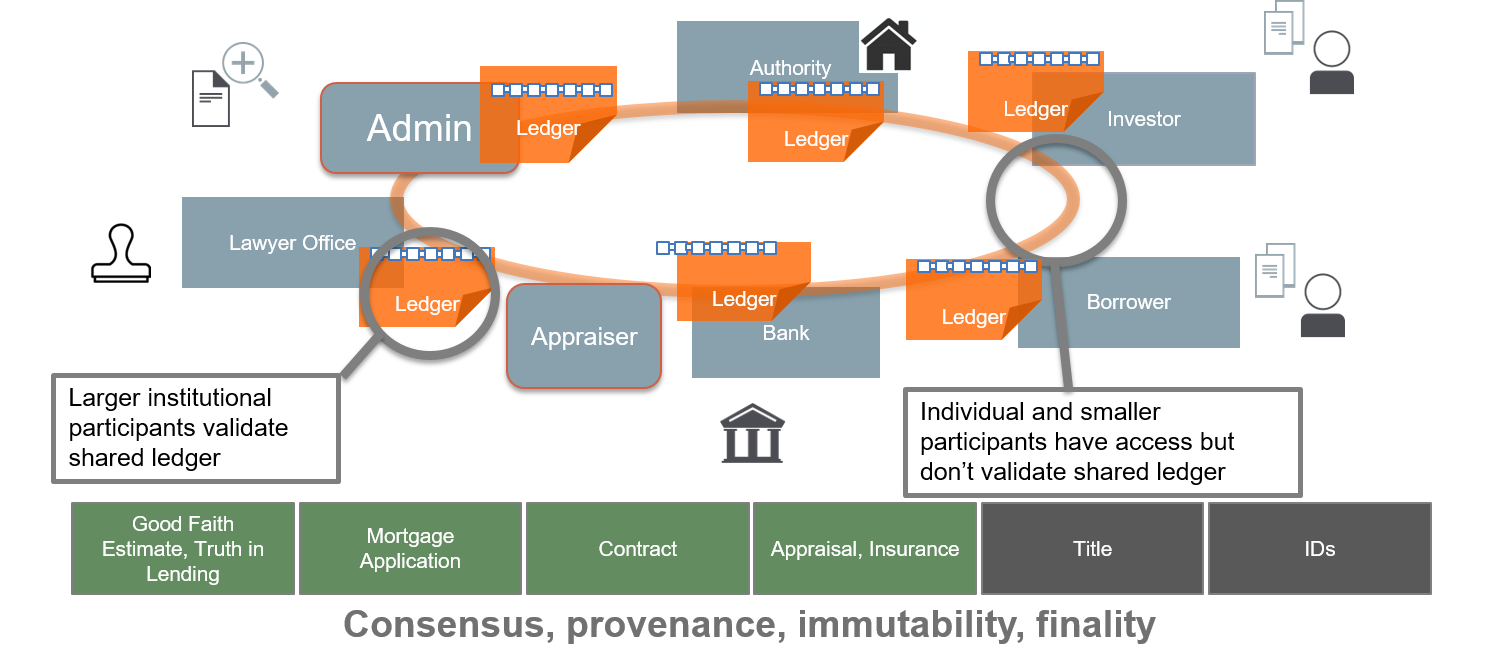
## Mutual Chain Mortgage Origination: Processes are timely, costly and can result in search errors, which generate transactional risk

# 

## Mutual Chain Mortgage Origination: Processes are timely, costly and can result in search errors, which generate transactional risk



## Mutual Chain A shared ledger between involved parties can increase trust, speed of execution, auditability and cost



# 互助链做为一个基于区块链技术的融资平台

# 互助链Muton币ICO及ICO之后的项目情况

## 互助币Muton（代币符号:MUC）的发售

## 互助币的锁定机制

## 互助币最低筹集资金和资金使用情况

# 互助链开发计划与预算

# 互助链团队

（见附件）

# 附录

## 项目组织结构

### 核心团队

第一阶段，我们预计需要建立7-11人的国际团队，具体包括：

* 项目总监委员会
* 执行总监(1)
* 区块链技术主管/架构师 (1~2)
* 区块链研发工程师 (3~5)
* 前端开发设计 (1)
* 保险逻辑和产品开发 (1~3)

### 角色与责任

|  |  |  |  |
| --- | --- | --- | --- |
| **团队** | **角色** | **姓名** | **负责的具体工作** |
| 项目总监委员会 | 委员会成员 |  |  |
| 执行与运营 | 执行总监 |  |  |
| 架构师 | 区块链架构师 |  |  |
| 区块链技术主管 |  |  |  |
| 区块链研发工程师 |  |  |  |
| 前端开发设计 |  |  |  |
| 保险逻辑和产品开发 |  |  |  |

## 主要里程碑Key Milestone

|  |  |  |
| --- | --- | --- |
| **#** | **里程碑** | **日期** |
|  | Phrase I |  |
|  | Phrase II |  |

**参考文献**

|  |  |
| --- | --- |
| [1] | Satoshi. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008. [Online]. Available: <http://bitcoin.org/bitcoin.pdf>. [Accessed 10 09 2015]. |
| [2] | Vitelik. Buterin, “A Next-Generation Smart Contract and Decentralized Application Platform”, 2013. <https://github.com/ethereum/wiki/wiki/White-Paper> |
| [3] | Gavin. Wood, "Ethereum: A Secure Decentralized Generalised Transaction Ledger," 2014. <http://gavwood.com/paper.pdf> |
| [4] | ERC 20 Type Tokens. “ERC: Token Standard” q  <https://github.com/ethereum/EIPs/issues/20> |
| [5] | David. A. Johnston, “The General Theory of Decentralized Applications, Dapps”, 2014. <https://github.com/DavidJohnstonCEO/DecentralizedApplications> |