

Koumei Cottage

**孔明屋**

白  
皮  
书

Draw on the wisdom of the masses

汇聚群众的智慧

## 前 言

区块链技术被认为是继大型机、个人电脑、互联网之后计算模式的颠覆式创新，很可能在全球范围引起一场新的技术革新和产业变革。

Koumei Cottage，简称 KMC，中文名孔明屋，是基于阿希平台开发的去中心化的预测市场。它属于 type 1 型 dapp，按照阿希白皮书<sup>[1]</sup>中对 dapp 的定义，它具有开源、自治、分布式存储、公开透明、代币激励等特性，与以太坊上的 dapp 的区别是，它的数据存储在独立的区块链中，而不是与其他应用共享一个链或一套账本，不附属于其他区块链系统。

我们在开发 KMC 的过程中，大量参考了 Augur、Delphy 白皮书中的内容，包括预测市场的概念、原理、算法等，但每一行代码都是独立、自主完成的，可以理解为，KMC 是我们对 Augur、Delphy<sup>[2]</sup> 白皮书的一个开源实现，基于阿希平台提供的应用 SDK 的一个实现。

为了避免重复性劳动，这份白皮书不会写太多前人已经写过的东西，关于预测市场的细节问题，我们建议用户直接阅读 Augur、Delphy 的白皮书。

但我们也会引用 Augur、Delphy 白皮书中的一些精要文字，以方便 KMC 的用户迅速理解预测市场的一些基础概念。

## 背景概述

随着区块链技术的发展，公有链成为一个公正的分布式信任中介。Koumei Cottage 就是为了建立一个可信、自治、可扩展的预测市场。区块链信息透明、不可篡改等特性与 Koumei Cottage 完美结合，加上区块链天然的特性让 Koumei Cottage 的运行近乎零成本，智能合约的实现更是让区块链拥有可编程动态决定结果的计算能力，在这样的背景下 Koumei Cottage 应运而生。

## 什么是预测市场

现代的科学预测方法主要有两种，一种是利用统计和数学模型进行预测，另一种是利用机器学习和数据挖掘进行预测。本质上，这两种方法主要利用历史数据和软件系统来产生预测。

近年来，利用“社会化分析（social analysis）”的预测市场作为新的第三种方法呈现异军突起之势。预测市场利用市场的激励机制，使得大众都能贡献出自己的经验和智慧，汇集市场信息帮助人们做决策，让参与者比任何单独的个体、专家更具有智慧。

Delphy 利用市场的激励机制，帮助市场的参与者透明地、实时地表达自己对未来事件的发生结果的信心和判断，从而实现有效地预测未来。Delphy 内生的分布式的机制保证了预测结果的不可操纵，也为群体智慧所依赖的信息的多元化、决策的独立性和组织的分布式提供了有效的基础设施保障。

Delphy 是一个预测即服务(Prediction as a Service, PaaS) 的移动应用平台和生态链。用户一方面可以随时随地参与预测市场的交易，另一方面也可以利用 Delphy API & SDK 实现各种定制，开设各个垂直领域的预测市场。Delphy 预测市场的应用相当广泛，包括但不限于金融、保险、国防、医疗卫生、公共管理、体育、娱乐，甚至企业内部的预测市场等。

我们可以看到，预测市场是一种科学的预测和分析技术，利用群体智慧帮助人们做决策，与机器学习、数据挖掘技术可以相提并论，目的相同，只是手段不同。预测市场与其他技术的优势是，通过经济杠杆带动群体智慧，实时的对事件作出判断。

从系统角度来说，KMC 这个预测市场与 Delphy 类似，它包含一套应用交互界面和开放的 API，既可以为普通用户提供直接服务，也可以供开发者进行二次开发，以实现定制化的垂直领域的预测市场。

本质上，预测市场是基于市场原则来收集整理交易各方对同一事件的信心和判断，从而产生对事件的未来结果的预测。如果说，股票市场是在为股票未来的预期收益定价一样，预测市场就是在为未来事件的预期结果进行定价。

预测市场可以理解为股票、证券、数字资产交易所的抽象和升华，而股票、证券、数字资产交易所可以看做是预测市场在某一事件下的特化。

## 预测市场准确性的案例

### 糖豆案例 1：

在 2007 年，哥伦比亚商学院教授 Michael Mauboussin 让他的 73 位学生聚集瓶子中糖豆的数量。学生所估计的数量在 250-4100 之间。其实瓶子中有 1116 个糖豆。学生们估计值与真实值 1116 平均偏离 700。也就是 62% 的错误率。然而，尽管学生的估计很不准确，但是他们估计的平均值是 1151，与真实数值 1116 只有 3% 的误差。

这一研究以各种形式被重复过多次，结果都与上面相同。我们正在将这种群体智慧应用到每一个学科中，从政治学到气候学，并用利益得失来强迫群体说真话。

### 潜艇案例 2：

在 1968 年 5 月，美国的一艘名为 Scorpion 的潜艇，在被大西洋完成执勤任务返回纽波特纽斯港口的途中消失了。虽然海军知道潜艇最后的报告位置，但是不知道 Scorpion 发生了些什么事情，只知道自最后一次联系后，潜艇大概又前行到哪里。最后他们将搜索范围确定在方圆 20 英里，几千英尺深的区域。这是一个希望渺茫的搜索。人们能够想到的唯一可能的解决方案是，召集三、四位潜艇和洋流的顶级专家，问他们认为潜艇在哪里。但是，根据 Sherry Sontag 和 Christopher Drew 在《Blind Mans Bluff》的记录，一位名叫 John Craven 的海军军官有一个不同的计划。

首先，Craven 设想一系列可以解释 Scorpion 可能发生的事故的情景。然后，他召集了一组具有不同背景的人，包括数学家、潜艇专家和搜寻人员。Craven 让他们猜测那种情景的可能性最大，而不是让他们彼此商量得出答案。为了让猜测更加有趣，Craven 采用了下注的模式，奖品是 Chivas Regal 酒。参与的成员就潜艇为什么出事故、下沉的速度、倾斜的

角度等问题进行打赌。

没有一段信息碎片能够告诉 Craven 潜艇在哪里。但是 ,Craven 相信 , 如果他将小组成员提出的所有答案汇集在一起 , 针对潜艇沉没做一个完整描述 , 他就能够知道潜艇在哪里。这就是 Craven 所做的事情。他利用了所有的猜测 , 使用被称为贝叶斯理论的公式 , 判断潜艇的最后位置。( 贝叶斯理论是计算事件的新信息如何改变你对此事件原有预期的方式。) 做完这些事情 , Craven 获得了团队关于潜艇位置的集体估计 ( collective estimate )。

Craven 得出的位置并不是团队任何单个成员所猜测的位置。换句话说 , 团队中每个成员的猜测与 Craven 使用汇集起来的所有信息得出的位置一致。最后的判断是一个由团队整体做出的集体判断 , 而不是代表团队中最聪明人的个人判断。它也是一个绝妙的判断。

Scorpion 潜艇失踪五个月以后 , 一艘海军船发现了它。潜艇所发现的位置与 Craven 团队猜测的位置相差 220 码 ( 译者注 : 1 码等于 0.9144 米 )。

这个实例的惊人之之处在于 , 这个团队所依靠的证据几乎没有 , 只是一些数据碎片。没有人知道为什么潜艇沉没 , 没人知道潜艇下沉的速度和倾斜角度。虽然团队中没人摘掉这些信息 , 但是作为一个整体的团队却知道这些信息。

## 为什么要做 KMC

与 CCTime 一样，KMC 也是阿希平台的一个示范型应用，其目的，一方面是为了检验和证明阿希平台的能力，另一方面是为了给阿希社区的开发者一个参考范例。

另外，传统预测市场存在信任问题，非常适合用区块链的技术和去中心的思想去解决，而目前市场上存在的基于区块链技术的预测市场系统远非完美，KMC 在借鉴其他同类产品的同时，也有其自身的特色，是对这个行业的一个有效的补充。

## 特 色

KMC 是基于阿希侧链技术的去中心化应用，与生俱来的带有阿希平台应用的特色，包括独立账本、环境隔离、资产跨链、易升级、低成本等特点，下面我们分别解释。

### 独立账本与环境隔离

KMC 不像以太坊上的应用，数据附属在以太坊的账本中，并且与其他应用共享资源。KMC 是一个独立的区块链系统，拥有独立的账本数据、见证人和共识网络。

独立账本带来的好处是没有数据臃肿问题，用户不会为不相关的应用数据买单，也不会受到宿主区块链的影响和限制。同时，也不会与平台上的其他应用竞争资源，不同应用中的错误也不会彼此影响。

### 多资产复用

阿希主链上注册的资产可以自由转入任意一个 dapp，因此 KMC 系统天生就可以运行多种资产。

也就是说 KMC 系统中的不同市场可以直接支持不同的资产或币种，大家可以用各



自拥有的币种直接参与事件的预测，而不是必须兑换成 KMC 的原生代币，KMC 的原生代币主要用于系统的手续费支付，也是一种可以代表系统决策权的权益代币。

阿希团队目前正在积极研发与比特币和以太坊系统的跨链，将来比特币和以太币融入阿希生态之后，KMC 也可以受益。

## 易于升级

由于该系统是一个相对独立的系统，源码本身并没有作为 chaincode 存储在链上，升级相对容易。

## 成本

合约的执行只消耗 KMC 系统原生代币，不受制于 XAS，XAS 只会在注册 KMC 代币时才会消耗，也就是说对只对 KMC 系统本身感兴趣的用户，无需持有 XAS，有效降低了用户的成本。

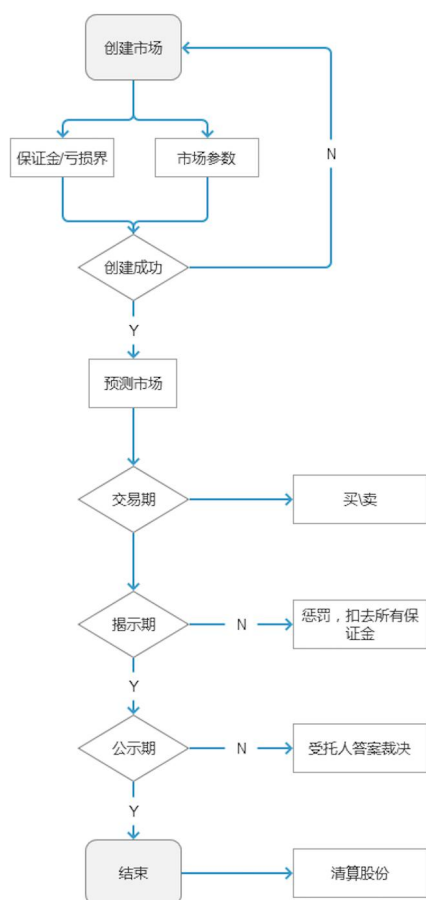
## 设计与实现

### 角色与用例

系统用户按角色可分为三种：

- 1、市场发起人，负责市场的创建，事件和结果的揭示。
- 2、普通用户，主要参与的功能是对事件各种可能性的预测、投票，对市场的评论，对结果的投诉。
- 3、受托人，主要负责对有争议事件结果的裁决，以防止市场发起人作弊。

### 业务流程



创建市场 -> 预测期(交易期) -> 交易结束 ->  
揭示期 -> 公示期 -> 结束

首先，任何账户都可以创建市场，创建市场必须抵押一定的资产，KMC 系统中叫做保证金，在其他系统中也有的叫亏损界，它决定了市场创建者可能亏损的资产的上限。

除了保证金，还需要创建者提供这个市场的其他参数，比如标题、图标、描述、选项、结束时间、所支持的币种、总股份数等。

市场被创建后，进入预测期，或者交易期，任何持有该市场支持所支持币种的用户可以参与预测，从这个市场所描述的事件的可能发生的若干个结果中选择一个就叫预测，有的系统叫做投票(ballot)，预测的

时候还需要冻结一定数额的资产，然后你就可以得到相应的股份，你可以在任何时间点卖掉你的股份。

经过一段时间后，市场交易期结束，进入揭示期，该期间持续 24 小时，创建者必须在此期间揭示符合事实的正确答案，否则会受到惩罚，丢掉所有保证金。

答案被揭示后，进入公示期，公示期持续 24 小时，在公示期间，受托人可以对答案进行裁决，如果市场创建者故意揭示了错误的答案，受托人可以通过联合签名的方式进行修正，并没收市场创建者的保证金。

公示期结束后，预测成功的参与者可以进行清算流程，将自己所拥有的股份按比例兑换成相应资产。

## 状态模型

### 市场

- id 市场编号
- tid 事务编号
- initiator 发起人
- timestamp 创建时间
- title 标题
- image 图标 url
- desc 市场描述
- endHeight 交易截止高度
- revealHeight 揭示高度
- state 状态
- currency 支持币种

- margin 保证金
- share 初始股份
- total 市场总额
- revealChoice 揭示选项
- verdictChoice 仲裁选项

## 事件结果

- mid 市场编号
- choice 事件选项
- desc 事件选项内容
- share 该选项所投入的总股份

## 交易记录

- mid 市场编号
- tid 事务编号
- trader 交易者地址
- choice 事件选项
- share 交易股份
- amount 交易金额

## 结算记录

- mid
- Tid

- address
- amount
- Share

#### **股份余额**

- mid
- address
- choice
- Share

#### **评论**

- mid
- tid
- authorId
- Content

#### **投诉**

- mid
- tid
- appealer
- content
- Amount

### 揭示

- mid
- tid
- Choice
- Height

### 裁决

- mid
- tid
- choice
- publicKeys
- signatures

## 激励与惩罚机制

根据 LMSR(logarithmic Market scoring rule)法则，市场发起人必须发布难以预测的事件才有利可图，普通用户预测成功一个可能性较低的事件时，能获得较高回报，预测成功一个可能性较高的事件时，获得的回报较低，如果预测失败则损失所有参与预测的资金。

市场发起人在创建市场时需要交纳保证金，如果市场发起人创建了不符合规则或常理的市场，或在揭示答案时进行作弊，可能会损失所有保证金。

## 合约调用

### 创建市场

```
1 . let secret = 'account secret'
2 . let title = 'market title'
3 . let image = 'http://xxxxx/logo.png'
4 . let desc = 'market description'
5 . let results = 'first choice,second choice'
6 . let currency = 'XAS'
7 . let margin = '20000'
8 . let share = 100
9 . let endHeight = 10000
10 . let trs = AschJS.dapp.createInnerTransaction({
11 .   fee: '100000000',
12 .   type: 1000,
13 .   args: [
14 .     title, image, desc, results, currency, margin, share, endHeight
15 .   ]
16 . }, secret)
```

### 预测交易（买卖）

```
1 . let secret = 'account secret'
2 . let mid = '1'
```

```
3 . let share = 20
4 . let choice = 0
5 . let trs = AschJS.dapp.createInnerTransaction({
6 .   fee: '10000000',
7 .   type: 1001,
8 .   args: [
9 .     mid,share,choice
10 .   ]
11 . }, secret)
```

## 结算

```
1 . let secret = 'account secret'
2 . let mid = '1'
3 . let trs = AschJS.dapp.createInnerTransaction({
4 .   fee: '10000000',
5 .   type: 1002,
6 .   args: [
7 .     mid
8 .   ]
9 . }, secret)
```

## 揭示

```
1 . let secret = 'account secret'
```



```
2 . let mid = '1'
3 . let choice = 0
4 . let trs = AschJS.dapp.createInnerTransaction({
5 .   fee: '10000000',
6 .   type: 1003,
7 .   args: [
8 .     mid, choice
9 .   ]
10 . }, secret)
```

## 申诉

```
1 . let mid = '1'
2 . let content = 'appeal content'
3 . let amount = '100000'
4 . let trs = AschJS.dapp.createInnerTransaction({
5 .   fee: '10000000',
6 .   type: 1004,
7 .   args: [
8 .     mid, content, amount
9 .   ]
10 . }, secret)
```

## 仲裁

```
1 . let mid = '1'
2 . let choice = 1
3 . let signatures = 'sig1,sig2,sig3'
4 . let trs = AschJS.dapp.createInnerTransaction({
5 .   fee: '10000000',
6 .   type: 1005,
7 .   args: [
8 .     mid,choice,signatures
9 .   ]
10 . }, secret)
```

## 查询接口

### 获取所有市场

#### 接口

- get /markets

#### 参数

currency 获取指定币种的市场，比如 XAS

initiator 获取指定创建者的市场，格式为账户地址，比如：

APqTgWgMZqrsWgK1J3FbESHoiYYNeCm5Hz

state 获取指定状态的市场，整数，比如 1

tid 使用交易号获取指定单个市场，交易号，比如：

8065a105c785a08757727fded3a06f8f312e73ad40f1f3502e0232ea42e67efd

limit 获取的最大数量，整数，如 20，默认 50

offset 偏移位置，整数，如 10，默认 0

#### 示例

1. `http://localhost:4096/api/dapps/826571fb9dd13627377c4e86e46908aaecdc97e6`

`fc97ccaf3e68e318867229b6/markets`

2 .

3 . {

4 .     "markets": [

5 .         {

6 .             "id": "1",

7 .             "tid":

```
"0ea46e653d02174c5963dff5beabc4d869e8e39da9b2998f6ad96f4113dbef86",
```

```
8 .      "initiator": "ABoicmy1Lh134YyTYjvbHto75JpU5MYjov",
9 .      "timestamp": 41427905,
10 .     "title": "first market title",
11 .     "image": "http://asch.so/logo.png",
12 .     "desc": "first market desc",
13 .     "results": 26,
14 .     "endHeight": 12000,
15 .     "revealHeight": 0,
16 .     "state": 0,
17 .     "currency": "XAS",
18 .     "margin": "100000000000",
19 .     "share": 100,
20 .     "total": "0",
21 .     "t_timestamp": 41427905,
22 .     "t_type": 1000,
23 .     "t_height": 4
24 .   }
25 . ],
26 .   "count": 1,
27 .   "success": true
28 . }
```

## 获取单个市场

### 接口

- get /markets/:id

### 示例

```
1 . http://localhost:4096/api/dapps/826571fb9dd13627377c4e86e46908aaecdc97e6
fc97ccaf3e68e318867229b6/markets/1
2 . {
3 .   "market": {
4 .     "id": "1",
5 .     "tid":
"0ea46e653d02174c5963dff5beabc4d869e8e39da9b2998f6ad96f4113dbef86",
6 .     "initiator": "ABoicmy1Lh134YyTYjvbHto75JpU5MYjov",
7 .     "timestamp": 41427905,
8 .     "title": "first market title",
9 .     "image": "http://asch.so/logo.png",
10 .    "desc": "first market desc",
11 .    "results": 26,
12 .    "endHeight": 12000,
13 .    "revealHeight": 0,
14 .    "state": 0,
15 .    "currency": "XAS",
16 .    "margin": "10000000000",
17 .    "share": 100,
```

```
18 .     "total": "0",
19 .     "t_timestamp": 41427905,
20 .     "t_type": 1000,
21 .     "t_height": 4
22 . },
23 .     "success": true
24 . }
```

## 获取单个市场的选项

### 接口

- get /markets/:id/results

### 参数

probability 加上此选项可以计算每个选项的获胜概率

### 示例

```
1 . http://localhost:4096/api/dapps/826571fb9dd13627377c4e86e46908aaecdc97e6
fc97ccaf3e68e318867229b6/markets/2/results
2 .
3 . {
4 .     "results": [
5 .         {
6 .             "mid": "2",
7 .             "choice": 0,
8 .             "desc": "first choice",
9 .             "share": 0
```

10 .        },

11 .        {

12 .        "mid": "2",

13 .        "choice": 1,

14 .        "desc": "second choice",

15 .        "share": 0

16 .        }

17 .        ],

18 .        "success": true

19 .        }

20 .        http://localhost:4096/api/dapps/826571fb9dd13627377c4e86e46908aaecdc97e6

fc97ccaf3e68e318867229b6/markets/1/results?probability=1

21 .        {

22 .        "results": [

23 .        {

24 .        "mid": "1",

25 .        "choice": 0,

26 .        "desc": "first choice",

27 .        "share": 20,

28 .        "probability": 0.549833997312478

29 .        },

30 .        {

31 .        "mid": "1",

32 .        "choice": 1,

```
33 .     "desc": "second choice",
34 .     "share": 0,
35 .     "probability": 0.45016600268752216
36 .     }
37 . ],
38 .     "success": true
39 . }
```

## 获取单个市场的历史交易

### 接口

- get /markets/:id/trades

### 示例

```
1 . http://localhost:4096/api/dapps/826571fb9dd13627377c4e86e46908aaecdc97e6
fc97ccaf3e68e318867229b6/markets/2/trades
2 .
3 . {
4 .     "trades": [
5 .         {
6 .             "mid": "13",
7 .             "tid":
"a7ab83c5c8ad8cd90534a05d00ecef7e61aa61a9b5353fa40a95c66a79207525",
8 .             "trader": "AAEfqC2LHCiJnQFwDrapeFALBc98ugoxkK",
9 .             "choice": 1,
10 .            "share": 10,
```



```
11 . "amount": "462577900",
12 . "t_timestamp": 42304225,
13 . "t_type": 1001,
14 . "t_height": 1237
15 . },
16 . {
17 . "mid": "13",
18 . "tid":
```

```
"8ef01d7a959a624c11be23010d0f13930bf97e64faf1dede19d8a7bc1c071d12",
```

```
19 . "trader": "AHufgbJaQJkd39EbRVvzjB5AGsyLb6etv",
20 . "choice": 1,
21 . "share": 30,
22 . "amount": "1537422100",
23 . "t_timestamp": 42304225,
24 . "t_type": 1001,
25 . "t_height": 1237
26 . },
27 . {
28 . "mid": "13",
29 . "tid":
```

```
"dc74c202045da57a0069ddee5b7a091008b099af9a06b05211011a81669a24d1",
```

```
30 . "trader": "AClhQcQ8gqUUZGMVqaNH833THXA9PPb5H5",
31 . "choice": 0,
32 . "share": 20,
```

```
33 .      "amount": "1049916900",
34 .      "t_timestamp": 42304215,
35 .      "t_type": 1001,
36 .      "t_height": 1236
37 .    }
38 .  ],
39 .  "count": 3,
40 .  "success": true
41 . }
```

## 获取单个市场的清算信息

### 接口

- get /markets/:id/settles

### 示例

```
1 . http://localhost:4096/api/dapps/826571fb9dd13627377c4e86e46908aaecdc97e6
fc97ccaf3e68e318867229b6/markets/13/settles
2 .
3 . {
4 .   "settles": [
5 .     {
6 .       "mid": "13",
7 .       "tid":
"ee0922b2882adbf6c6250f17d8330533bf592c6f658b8568a804dfcd3c57c2ee",
8 .       "address": "A5jHjQcfd78WffvaPDLd4aBZkseaiae2L4",
```

```
9 .      "amount": "7981388700",
10 .     "share": 0,
11 .     "t_timestamp": 42304365,
12 .     "t_type": 1002,
13 .     "t_height": 1251
14 .     },
15 .     {
16 .         "mid": "13",
17 .         "tid":
"aa68cae1f771d4af49ac82ba63dd0f38f4be229e128b526df620e618d653b38a",
18 .         "address": "ACLhQcQ8gqUUZGMVqaNH833THXA9PPb5H5",
19 .         "amount": "2000000000",
20 .         "share": 20,
21 .         "t_timestamp": 42304365,
22 .         "t_type": 1002,
23 .         "t_height": 1251
24 .     }
25 .     ],
26 .     "count": 2,
27 .     "success": true
28 . }
```

## 获取单个市场的所有评论

### 接口

- get /markets/:id/comments

### 示例

```
1 . http://localhost:4096/api/dapps/826571fb9dd13627377c4e86e46908aaecdc97e6
fc97ccaf3e68e318867229b6/markets/3/comments
2 . {
3 .   "comments": [
4 .     {
5 .       "mid": "3",
6 .       "tid":
"48262c32308cfa58f53533633a71c11ead5c837d0e4ca184b67189c3755ba2b",
7 .       "authorId": "APqTgWgMZqrswgK1J3FbESHoiYYNeCm5Hz",
8 .       "content": "first comment",
9 .       "t_timestamp": 41537245,
10 .      "t_type": 1006,
11 .      "t_height": 1047
12 .    }
13 .  ],
14 .  "count": 1,
15 .  "success": true
16 . }
```

## 获取单个市场的揭示信息

### 接口

- get /markets/:id/reveal

### 示例

```
1 . http://localhost:4096/api/dapps/826571fb9dd13627377c4e86e46908aaecdc97e6
fc97ccaf3e68e318867229b6/markets/13/reveal
2 .
3 . {
4 .   "reveal": {
5 .     "mid": "13",
6 .     "tid":
"c653b497ff3500f48942fa1fd29bdb9a90331c0e9ed8e9357b68b7a81fcd7378",
7 .     "choice": 0,
8 .     "height": 1241,
9 .     "t_timestamp": 42304275,
10 .    "t_type": 1003,
11 .    "t_height": 1242
12 .  },
13 .  "success": true
14 . }
```

## 获取单个市场的仲裁信息

### 接口

- get /markets/:id/verdict

### 示例

```
1 . http://localhost:4096/api/dapps/826571fb9dd13627377c4e86e46908aaecdc97e6
fc97ccaf3e68e318867229b6/markets/42/verdict
2 .
3 . {
4 .   "verdict": {
5 .     "mid": "42",
6 .     "tid":
"8e31dd741dfdaf64b5d104fa203a7b66518c04363dd6a80e394d2e67b21ec644",
7 .     "choice": 0,
8 .     "signatures":
"8b1c24a0b9ba9b9ccf5e35d0c848d582a2a22cca54d42de8ac7b2412e7dc63d4adbf962c11
cca7a68e4c1898e68b28d40d640c0c8acaf2e59e6074a17841b21fcf8e00fe201acf89843ffe6
07bf7289d40b5b943301c5a91f3e2f4fca7f5070d,aa7dcc3afd151a549e826753b0547c90e6
1b022adb26938177904a73fc4fee36424df2baf542c431f7310444b64404606716b39ce4a92
471cb5aa28aea35b274b608945208791853ab531988a5c546662f211ce7b118676ec812b51
ee820570c,e29c75979ac834b871ce58dc52a6f604f8f565dea2b8925705883b8c001fe8ceb9
ddcba63c2907c490c8577a1f08a406fbfa0515272613e38299d6d30bb1803d9bb0f559df4c7
4ae511e13a4b4510d9b782559d0ae40cbd960902b987b95f405",
```

```
9 .      "t_timestamp": 43278675,  
10 .     "t_type": 1005,  
11 .     "t_height": 18293  
12 .     },  
13 .     "success": true  
14 . }
```

## 获取单个市场下某个用户的所有股份

### 接口

- get /markets/:id/shares/:address

### 示例

```
1 . http://localhost:4096/api/dapps/826571fb9dd13627377c4e86e46908aaecdc97e6  
fc97ccaf3e68e318867229b6/markets/7/shares/APqTgWgMZqrsWGK1J3FbESHoiYYNeCm5  
Hz
```

```
2 .  
3 . {  
4 .   "shares": [  
5 .     {  
6 .       "mid": "7",  
7 .       "address": "APqTgWgMZqrsWGK1J3FbESHoiYYNeCm5Hz",  
8 .       "choice": 0,  
9 .       "share": 2  
10 .    }  
11 .  ],
```

```
12 .   "count": 1,  
13 .   "success": true  
14 . }
```

## 获取用户的所有股份

### 接口

- get /shares/:address

### 示例

```
1 . http://localhost:4096/api/dapps/826571fb9dd13627377c4e86e46908aaecdc97e6  
fc97ccaf3e68e318867229b6/shares/APqTgWgMZqrswgK1J3FbESHoiYYNeCm5Hz  
2 .  
3 . {  
4 .   "count": 2,  
5 .   "shares": [  
6 .     {  
7 .       "mid": "7",  
8 .       "address": "APqTgWgMZqrswgK1J3FbESHoiYYNeCm5Hz",  
9 .       "choice": 0,  
10 .      "share": 2  
11 .    },  
12 .    {  
13 .      "mid": "14",  
14 .      "address": "APqTgWgMZqrswgK1J3FbESHoiYYNeCm5Hz",  
15 .      "choice": 0,
```



```
16 .      "share": 15
17 .      }
18 .    ],
19 .      "success": true
20 .    }
```

## 计算购买股份所需要的金额

### 接口

- get /markets/:id/calc

### 参数

choice 购买的选项

share 购买的股份数

### 示例

```
1 . http://localhost:4096/api/dapps/826571fb9dd13627377c4e86e46908aaecdc97e6
fc97ccaf3e68e318867229b6/markets/1/calc?choice=0&share=20
```

```
2 .
3 . {
4 .   "mid": "1",
5 .   "choice": 0,
6 .   "share": 20,
7 .   "amount": "1379678200",
8 .   "success": true
9 . }
```

## 结 语

人类对于未知的渴望，使得预测未发生事情的原动力深藏于每一个普通人的心中，从《乾坤万年歌》《马前课》《藏头诗》《推背图》《禅师诗》等一系列关于预言的古书都展示出古人对于预测未来的兴趣。但受限于当时获取信息的渠道、认知能力、科技水平的限制，很难进行基于群体智慧的对未知的事物进行可靠的预测。

科技的高速发展，使得获取信息的成本不断降低，每天都可以第一时间了解全球各地所正在发生以及将要发生的事情，信息互联网为我们提供了海量的信息，使得我们可以根据这些海量的信息对未来将要发生的事件进行预测。根据群体力量所涌现出的高级智慧进行预测往往具备很高的准确性，这样无论是对于个人还是企业、政府都可以进行决策前的指导，它是可以将决策的失误率大幅度的降低，大大提升整个社会资源的合理配置程度。

Koumei Cottage 提供了一个平台，它可以把大众的知识（Knowledge）和经验（Experience）转化为智慧（Wisdom）。

### 更多关注：

官方网址：[www.kmwisdom.com](http://www.kmwisdom.com)

QQ 群：669183944

论坛：[bbs.asch.io](http://bbs.asch.io)

微博：[AschPlatform](https://weibo.com/AschPlatform)

推特：[@AschPlatform](https://twitter.com/AschPlatform)

邮箱：[service@asch.io](mailto:service@asch.io)

【1】 <http://docs.asch.mobi/docs/>

【2】 天算白皮书： [https://delphy.org/papers/Delphy\\_Whitepaper\\_CH.pdf](https://delphy.org/papers/Delphy_Whitepaper_CH.pdf)

Augur 白皮书： <http://8btc.com/thread-19088-1-20.html>