

Factoid 应用程序接口(API)

这份是关于Factoids币和Factoid钱包的应用程序接口(API)的摘要。此处，钱包(Wallet)是一个命令行驱动程序，旨在演示API而非商业钱包解决方案。

第一步需要安装Factom客户端和Factom钱包帮手。参见[如何安装]指引，设定你的运行环境(<http://factom.org/howto>) 并需要运行Factomd和Fctwallet。请注意在今后的几个月内，可能需要对Factomd不定期的进行更新，以持续与网络对接。具体通知更新，请参见我们的[技术博客](#)

关于这两个应用程序的接口，**Factomd**是加入Factom网络的客户端，另一个**fctwallet**提供通用的钱包功能，并维护私钥所存放的地址簿。此外，第三个程序**walletapp**提供线下钱包(cold wallet)功能，生成线下交易并其上传Factom网络。

Fatcom客户端**Factomd**提供表征性状态传输(RESTful)接口,默认存放地址<http://localhost:8088> 任何向Factomd发出的请求都不会产生安全性问题，因此Factomd无需位于同一位置进行程序创建和提交事务。

Factom钱包**Fctwallet**亦提供表征性状态传输(RESTful)接口,默认存放地址<http://localhost:8089> 针对钱包中的Factoid地址，向Fctwallet发出请求会产生并建立交易，因此需要保证安全的访问接口。

Factom钱包**walletapp**是替代方案，仅支持Factoid功能，并只与Factomd进行通信。*walletapp*用于冷存储，创建线下交易，并将线下交易传输到Factom网络。我们很快会基于*walletapp*发布图形用户界面(GUI),向Factoid用户提供支持。

Factom钱包命令行**factom-cli**用于支持Factoid的交易，并接入Factom通用协议。*factom-cli*通过连接到*factomd*和*fctwallet*的接口，执行相应的功能。这个程序的主要用来证明Factom API的使用，也可以作为Factom脚本交易处理。

请注意：如果安装了curl，以下API调用的案例可以在命令行中直接运行，并根据具体情况，API的调用以curl -X POST 或者 curl -X GET作为前缀。

factomd

这份摘要，有关Factoids交易的factomd应用程序接口。我们在今后将会添加更多的细节。

- Post <http://localhost:8088/v1/commit-chain/>

生成链，创建一个新链的第一步
- Post <http://localhost:8088/v1/reveal-chain/>

显示链中的第一个记录，需要完成一个新链的构建
- Post <http://localhost:8088/v1/commit-entry/>

生成记录，写入链中记录的第一步
- Post <http://localhost:8088/v1/reveal-entry/>

显示新的条目，需要完成写入链中的记录
- Post <http://localhost:8088/v1/factoid-submit/>

提交事务。需要对交易进行编码，作为请求的一部分。例如，生成一个从xxx账户向yyy账户转10个Factoids币的一项交易，可以编码为

```
curl -X GET http://localhost:8088/v1/factoid-submit/http02015023e2886901010083ddb4b3006302ac3d
a1a1e5eac31af88cddb886f34470cc0415d1968d8637814cfac482f283dceb940025edb8b25808b6e6d
48ad5ba67d0843eaf962c40f63c9b4df91b8fe7364ae872014b776d236585f2ed658ec9d24a4a65e08e
f6074573f570b8b25a9d424b1d955d2caaa4d2cfe30eb8217844f8b28b8a47ce6dc3e5eecd03f30954c
a3f0b64a63e0687f667bc3300bb33a0638953d442db2cdfb4d27045318ec09463542c66305
```

这看上去是很复杂的数据结构。很多用户会使用fctwallet建立这项请求。

- Get <http://localhost:8088/v1/directory-block-head/>

返回目录区块头目的哈希值。无需任何参数，返回一个JSON形式的字符串。

```
{"KeyMR": "f7eb0456b30b1a4b50867a5307532e92dde7279ffc955ce1284cd142f94d642"}
```

- Get <http://localhost:8088/v1/directory-block-height/>

```
curl -X GET http://localhost:8088/v1/directory-block-height/
```

```
{"Height":4585}
```

- ```
curl -X GET http://localhost:8088/v1/get-raw-data/f7eb0456b30b1a4b50867a5307532e92ddee7279ffc955ce1284cd142f94d642
```

```

{"Data": "00fa92e5a291592f5f78c547560edceb8bc5ef142f20e9689fcd587557a2f3d18406d6e5ece9eacaa1c31d1371af60d6a9d5ea65654d1ff5698f7fb181d0ae4bc8582c093186dd2a14e83bbf53bb7cab230b1d0e2ce"}

```

```
curl -X GET http://localhost:8088/v1/directory-block-by-keymr/f7eb0456b30b1a4b50867a5307532e92ddee7279ffc955ce1284cd142f94d642
```

```
{ "Header": {
 "PrevBlockKeyMR": "e9eacaa1c31d1371af60d6a9d5ea65654d1ff5698f7fb181d0ae4bc8582c0931",
 "SequenceNumber": 4289,
 "Timestamp": 1443711000},
 "EntryBlockList": [
 { "ChainID": "000a",
 "KeyMR": "98f7817976ed8ff9aa306834d98c145d7c0334d7057f89dd2f035df1b37946ae" },
 { "ChainID": "000c",
 "KeyMR": "9432448e6c7f56450804b42ed9c1653182efb6f48a5d8da2c22d1789e7dbff44" },
 { "ChainID": "000f",
 "KeyMR": "b642daa292af42dda109bc87cddd31647da6fef9f3f25129c3740ef4d72761a0" },
 { "ChainID": "df3ade9eec4b08d5379cc64270c30ea7315d8a8a1a69efe2b98a60ecdd69e604",
 "KeyMR": "789b0103e5f8358d7f8402264837986a2b29ac59be8a796dbbe75eecf6a853d9" }
]
}
```

```
curl -X GET http://localhost:8088/v1/entry-block-by-keymr/789b0103e5f8358d7f8402264837986a2b29ac59be8a796dbbe75eecf6a853d9
```

```
"Header":{
 "BlockSequenceNumber":2479,
 "ChainID":"df3ade9eec4b08d5379cc64270c30ea7315d8a8a1a69efe2b98a60ecdd69e604",
 "PrevKeyMR":"63833701a61a846ebe8d38a1c6ede6bf5d5516990c34372c7f7936812ec09bde",
 "Timestamp":1443711000
},
"EntryList":[
 { "EntryHash":"c8f4936962836cda0d8bf712653d97f8d8b5cbe675e495b6dfab6b2395c8b80a",
 "Timestamp":1443711360
 }
]
```

这是一个记录区块的结构，分拆成JSON结构

- Get [http://localhost:8088/v1/entry-by-hash/\(\[^\s\]+\)](http://localhost:8088/v1/entry-by-hash/([^\s]+)), **handleEntry**)

返回一项分拆成JSON的记录：

```
curl -X GET http://localhost:8088/v1/entry-by-hash/c8f4936962836cda0d8bf712653d97f8d8b5cbe675e495b6dfab6b2395c8b80a
```

返回:

```
{"ChainID":"df3ade9eec4b08d5379cc64270c30ea7315d8a8a1a69efe2b98a60ecdd69e604",
"Content": "7b22416e63686f725265636f7264566572223a312c224442486569676874223a343238382c224b65794d52223a2265396561636161316333316431333731616636306436613964356561363536353464316666353
```

返回一项特定记录，并分拆成JSON格式

- Get [http://localhost:8088/v1/chain-head/\(\[^\s\]+\)](http://localhost:8088/v1/chain-head/([^\s]+))

返回条目链中第一项记录的梅克尔根(KeyMR)。请求：

```
curl -X GET http://localhost:8088/v1/chain-head/df3ade9eec4b08d5379cc64270c30ea7315d8a8a1a69efe2b98a60ecdd69e604
```

返回：

```
{"ChainHead":"bfd814a3b9a4356e04c816fe4ce1a53198953ab321912d60dacba766950e5591"}
```

- Get [http://localhost:8088/v1/entry-credit-balance/\(\[^\s\]+\)](http://localhost:8088/v1/entry-credit-balance/([^\s]+))

返回指定数据条目信用地址的余额。例如，请求：

```
curl -X GET http://localhost:8088/v1/entry-credit-balance/748be8327d20fee4365e6b5a3dca7df1e59da47e9ebd99129ba84d58d4d0726b
```

可能返回（取决于该地址该时刻的余额）：

```
{"Response": "4000", "Success": true}
```

这表明该数据条目信用地址 (EC2eUoDPupuQXm5gxs1sCBCv3bbZBCYFDtJafQ6iRaAKfyXNqjEJ) 解码为: 748be8327d20fee4365e6b5a3dca7df1e59da47e9ebd99129ba84d58d4d0726b 并有4000的条目信用积分。

- Get [http://localhost:8088/v1/factoid-balance/\(\[^\s\]+\)](http://localhost:8088/v1/factoid-balance/([^\s]+))

返回指定地址的Factoid余额。例如，该项请求：

```
curl -X GET http://localhost:8088/v1/factoid-balance/f6e117ea838cb652e9cfc3b29552d5887800a7ba614df0bd8c13e171eddc5897
```

```
{"Response": "1210268000", "Success": true}
```

Get <http://localhost:8088/v1/factoid-get-fee/>

```
curl -X GET http://localhost:8088/v1/factoid-get-fee/
```

```
{"Fee":100000}
```

- Get `http://localhost:8088/v1/properties/"`, `handleProperties`)

```
curl -X GET http://localhost:8088/v1/properties/
```

```
{
 "Protocol_Version":1005,
 "Factomd_Version":3002,
 "Fctwallet_Version":0
}
```

- Get [http://localhost:8089/v1/factoid-balance/\(\[^\s/\]+\)/](http://localhost:8089/v1/factoid-balance/([^\s/]+)/)

```
curl -X GET "http://localhost:8089/v1/factoid-balance/FAxxx"
curl -X GET "http://localhost:8089/v1/factoid-balance/9e72fa1dbdc30b557c857a1dcdca04b4ae748e52dc492e1f85f6af6f29f6534"
curl -X GET "http://localhost:8089/v1/factoid-balance/FactomAddress01"
```

```
{"Response": "1210680000", "Success": true}
```

- Get [`http://localhost:8089/v1/entry-credit-balance/\(\[^\s\]+\)`](http://localhost:8089/v1/entry-credit-balance/([^\s]+))

<http://epiceditor.com/>

```
curl -X GET "http://localhost:8089/v1/entry-credit-balance/ECxxx"
curl -X GET "http://localhost:8089/v1/entry-credit-balance/748be8327d20fee4365e6b5a3dca7df1e59da47e9ebd99129ba84d58d4d0726b"
curl -X GET "http://localhost:8090/EntryCreditAddress001"
```

将返回

```
{"Response": "4000", "Success": true}
```

假定你的钱包有一项条目记录 EntryCreditAddress001 对应所给定的公共地址的私钥。

- Get [`http://localhost:8089/v1/factoid-generate-address/\(\[^\^\]+\)`](http://localhost:8089/v1/factoid-generate-address/([^\^]+))

生成地址，并在钱包中创建一个条目保存该地址。地址是从确定的散列中创建。因此，当你备份钱包时，即使有些地址是备份后被创建，但是你的钱包也能够有效恢复。

调用:

```
curl -X GET http://localhost:8089/v1/factoid-generate-address/fctAddress0001
```

会生成一个新的地址fctAddress0001, 并为其分配一个新的私钥。

- Get [`http://localhost:8089/v1/factoid-generate-ec-address/\(\[^\^\]+\)`](http://localhost:8089/v1/factoid-generate-ec-address/([^\^]+))

生成信用条目地址，并在钱包中创建一个条目保存该地址。地址是从确定的散列中创建。因此，当你备份钱包时，即使有些地址是备份后被创建，但是你的钱包也能够有效恢复。

调用:

```
curl -X GET http://localhost:8089/v1/factoid-generate-ec-address/ECAddress0001
```

会生成一个新的地址 ECAddress0001, 并为其分配一个新的私钥。

- Get [http://localhost:8089/v1/factoid-generate-address-from-private-key/\(.\\*\)](http://localhost:8089/v1/factoid-generate-address-from-private-key/(.*))

这项请求是从另一个源导入十六进制的factoid私钥，提供了一个私钥和名字。例如：

```
curl -X GET http://localhost:8089/v1/factoid-generate-address-from-private-key/?name=addr01&privateKey=85d6755c286c6f139b1696ca74b0c14da473beadc37b2ec6273f2a92ce8d7c88
```

会导入给定的私钥，保存在钱包地址addr001下，并返回一个公钥。需要注意的是，以这种形式导入的私钥需要立即备份，以保证安全性。

- Get [http://localhost:8089/v1/factoid-generate-ec-address-from-private-key/\(.\\*\)](http://localhost:8089/v1/factoid-generate-ec-address-from-private-key/(.*))

这项请求是从另一个源导入十六进制的信用条目私钥，提供了一个私钥和名字。例如：

```
curl -X GET http://localhost:8089/v1/factoid-generate-ec-address-from-private-key/?name=addr001&privateKey=3ffa892f2445286a06c0dc591d7fa557d16701e444ec1cbee2930f7d7dfb62d57
```

会导入给定的私钥，保存在钱包地址addr001下，并返回一个公钥。需要注意的是，以这种形式导入的私钥需要立即备份，以保证安全性。

- Get [http://localhost:8089/v1/factoid-generate-address-from-human-readable-private-key/\(.\\*\)](http://localhost:8089/v1/factoid-generate-address-from-human-readable-private-key/(.*))

这项请求是从另一个源导入可读的factoid私钥，提供了一个私钥和名字。例如：

[illegible]

会导入给定的私钥，保存在钱包地址addr001下，并返回一个公钥。需要注意的是，以这种形式导入的私钥需要立即备份，以保证安全性。

- Get [http://localhost:8089/v1/factoid-generate-ec-address-from-human-readable-private-key/\(.\\*\)](http://localhost:8089/v1/factoid-generate-ec-address-from-human-readable-private-key/(.*))

这项请求是从另一个源导入可读的信用条目私钥，提供了一个私钥和名字。例如：

[illegible]

会导入给定的私钥，保存在钱包地址addr001下，并返回一个公钥。需要注意的是，以这种形式导入的私钥需要立即备份，以保证安全性。

- Get [http://localhost:8089/v1/factoid-generate-address-from-token-sale/\(.\\*\)](http://localhost:8089/v1/factoid-generate-address-from-token-sale/(.*))

在销售过程中接受Koinfy的信息，并在钱包中生成相应的记录，例如：

```
curl -X GET http://localhost:8089/v1/factoid-generate-address-from-token-sale/?name=koinifyAddr&mnemonic=<12 words separated by %20>
```

返回公钥

- Post [`http://localhost:8089/v1/factoid-new-transaction/\(\[^\s\]+\)`](http://localhost:8089/v1/factoid-new-transaction/([^\s]+))

创建一项新的交易以及对应的钥匙，该钥匙将在其他操作过程中用于添加输入或输出，添加信用条目输出，支付费用，签署交易，以及传递。例如：

```
curl -X POST http://localhost:8089/v1/factoid-new-transaction/trans
```

返回

```
{"Response":"Success building a transaction","Success":true}
```

生成了一项新交易，交易名为'trans'我们将在接下来的命令中使用该项交易。

- Post [`http://localhost:8089/v1/factoid-delete-transaction/\(\[^\s\]+\)`](http://localhost:8089/v1/factoid-delete-transaction/([^\s]+))

## 根据名字删除特定的交易

```
curl -X POST http://localhost:8089/v1/factoid-delete-transaction/trans
```

返回

```
{"Response":"Success deleting transaction","Success":true}
```

移除'trans'交易后。若要继续以trans交易名进行交易，需要重新创造交易'trans'。

- Post [http://localhost:8089/v1/factoid-add-fee/\(.\\*\)](http://localhost:8089/v1/factoid-add-fee/(.*))

对于指定交易增添费用。这项请求计算需要支付的费用，并把它添入特定的输入。输入和输出必须完全相等，因为这项调用不会处理不等的交易，由于校对金额一致会相当的棘手。

```
curl -X POST http://localhost:8089/v1/factoid-add-fee/ -d "key=trans&name=FA3EPZYqodgyEGXNMbiZKE5TS2x2J9wF8J9MvPZb52iGR78xMgCb"
```

返回

```
{"Response": "Added 0.153318 to FA3EPZYqodgyEGXNMbiZKE5TS2x2J9wF8J9MvPZb52iGR78xMgCb", "Success": true}
```

假定Factoid地址是该项交易的输入，那么所需要的费用将会被添加到该地址

- Post [http://localhost:8089/v1/factoid-add-input/\(.\\*\)](http://localhost:8089/v1/factoid-add-input/(.*))

添加给定的输入信息到特定的交易

```
curl -X POST http://localhost:8089/v1/factoid-add-input/ -d "key=trans&name=FA3EPZYqodgyEGXNMbiZKE5TS2x2J9wF8J9MvPZb52iGR78xMgCb&amount=10000000"
```

返回

```
{"Response": "Success adding Input", "Success": true}
```

添加某项给定地址的输入信息到“trans”交易， Factoids数量 (12)将会以定点表示法列示 (1200000000)

- Post [http://localhost:8089/v1/factoid-add-output/\(.\\*\)](http://localhost:8089/v1/factoid-add-output/(.*))

将给定的输出信息添加到特定的交易

```
curl -X POST http://localhost:8089/v1/factoid-add-output/ -d "key=trans&name=FA3SXWH3x3HJCjNd3LGrvZnZKJhmdSFKEYd1BgjeHeFPiTvwfw8N&amount=10000000"
```

返回

```
{"Response": "Success adding output", "Success": true}
```

将一项给定地址的输出信息添加到trans交易。 Factoids数量 (13)将会以定点表示法列示 (1300000000)

- Post [http://localhost:8089/v1/factoid-add-ecoutput/\(.\\*\)](http://localhost:8089/v1/factoid-add-ecoutput/(.*))

添加条目信用输出信息到特定的交易。需要注意的是，条目信用信息的固定输出形式是Factoids。具体分配的条目信用的数量取决于交易发生时factoid转换为条目信用的折换率。

```
curl -X POST http://localhost:8089/v1/factoid-add-ecoutput/ -d "key=trans&name=EC2ENydo4tjz5rMiZVDiM1k315m3ZanSm6LFDYcQyn5edBXNnrva&amount=10000000"
```

返回

```
{"Response": "Success adding Entry Credit Output", "Success": true}
```

添加某个给定条目信用地址的输出信息到trans交易。假定factoid与条目信用的转换率.001。那么1000条条目信用将转换为1个factoid作为输出信息1000， factoids以固定形式列示(100000000 == 1 factoid)

- Post [http://localhost:8089/v1/factoid-sign-transaction/\(.\\*\)](http://localhost:8089/v1/factoid-sign-transaction/(.*))

对特定交易签名。

```
http://localhost:8089/v1/factoid-sign-transaction/trans
```

返回

```
{"Response": "Success signing transaction", "Success": true}
```

对trans交易签名。

- Post [http://localhost:8089/v1/compose-submit-chain/\(\[^\s\]+\)](http://localhost:8089/v1/compose-submit-chain/([^\s]+))

创建一个JSON对象，用于通过Factomd请求，生成链和显示链

```
$ curl -X POST -H 'Content-Type: application/json' -d '{"ExtIDs":["foo", "bar"], "Content":["Hello Factom!"]}' localhost:8089/v1/compose-chain-submit/app
```

返回

```
{"ChainID": "92475004e70f41b94750f4a77bf7b430551113b25d3d57169eadca5692bb043d", "ChainCommit":
{"CommitChainMsg": "0001521deb5c7891ac03adffe815c64088dc98ef281de1891c0f99a63c55369c1727dc73580cbcc309ee55fa780ce406722b7a074138c994c859e2eda619bbad59b41775b51176464cb77fc08b6ef6767dcc315b4729a871071053cfe4af5a6
{"Entry": "0092475004e70f41b94750f4a77bf7b430551113b25d3d57169eadca5692bb043d000a0003666f6f000362617248656c6c6f20466163746f6d21"}}}
```

- Post [http://localhost:8089/v1/compose-submit-entry/\(\[^\s\]+\)](http://localhost:8089/v1/compose-submit-entry/([^\s]+))

创建一个JSON对象，用于通过Factomd请求，生成记录和显示记录

```
$ curl -i -X POST -H 'Content-Type: application/json' -d '{"ChainID": "5c337e9010600c415d2cd259ed0bf904e35666483277664d869a98189b35ca81", "ExtIDs":["foo", "bar"], "Content":["Hello Factom!"]}' localhost:8089/v1/compose-entry-submit/app
```

[返回](#)

```
{"EntryCommit":
{"CommitEntryMsg":"","0001521dc2d47d32cbdd3fc21889e22cc408ae0b0c120662c0873331cc5ce8ebdc1b6722968ce20179a1ad273d890287e5d4f16d2669c06c523b9e48673de1bfde3ea2fda309ac92f4f4b4d52cc6b228b9b621b1b1969ab46bfa4f80379e14d"
{"Entry":"","005c337e9010600c415d2cd259ed0bf904e35666483277664d869a98189b35ca81000a0003666f6f000362617248656c6c6f20466163746f6d21"}}}
```

- Post [http://localhost:8089/v1/commit-chain/\(\[^\s\]+\)](http://localhost:8089/v1/commit-chain/([^\s]+))

对一个特定的条目信用钥匙以二进制形式签署，并递交至Factomd服务器

- Post [http://localhost:8089/v1/commit-entry/\(\[^\s\]+\)](http://localhost:8089/v1/commit-entry/([^\s]+))

在记录链中提交记录

- Post [http://localhost:8089/v1/factoid-submit/\(.\\*\)](http://localhost:8089/v1/factoid-submit/(.*))

向Factom递交交易。该请求需要一个JSON参数。例如：递交一个名为trans的交易：

```
curl -X POST http://localhost:8089/v1/factoid-submit/{"Transaction":"trans"}
```

[返回](#)

```
{"Response":"Success Submitting transaction","Success":true}
```

- Get [http://localhost:8089/v1/factoid-validate/\(.\\*\)](http://localhost:8089/v1/factoid-validate/(.*))

目前未实现

- Get [http://localhost:8089/v1/factoid-get-fee/\(.\\*\)](http://localhost:8089/v1/factoid-get-fee/(.*))

获取每条信用条目对应的Factoid的换算率 例如：

```
curl -X GET http://localhost:8089/v1/factoid-get-fee/
```

[返回](#)

```
{"Response":"0.006666","Success":true}
```

- Get <http://localhost:8089/v1/properties/>

获取Factom客户端，fctwallet，factomd和协议的版本号 例如：

```
http://localhost:8089/v1/properties/
```

[返回](#)

```
{"Response":"Protocol Version: 0.1.5\nfactomd Version: 0.3.4\nfctwallet Version: 0.1.4\n","Success":true}
```

- Get <http://localhost:8089/v1/factoid-get-addresses/>

获取钱包的地址列表 例如：

```
curl -X GET http://localhost:8089/v1/factoid-get-addresses/
```

- Get <http://localhost:8089/v1/factoid-get-transactions/>

获取在建过程中的交易以及对应的钥匙 例如：



```
curl -X GET http://localhost:8089/v1/factoid-get-transactions/
```

- Post [http://localhost:8089/v1/factoid-get-processed-transactions/\(.\\*\)](http://localhost:8089/v1/factoid-get-processed-transactions/(.*))

如果键入'all'，会返回所有交易。如果键入一个地址，会返回所有使用该地址作为输入信息，输出信息，或作为信用条目输出的交易，并以文本形式返回。例如：

```
curl -X POST http://localhost:8089/v1/factoid-get-processed-transactions/ -d "address=FA2opZ5tRQET3LNRPfXFR2dWDRD1Sgc1aEYNStTXtkPWQtEvoAiY"
curl -X POST http://localhost:8089/v1/factoid-get-processed-transactions/ -d "address=<addrname>"
curl -X POST http://localhost:8089/v1/factoid-get-processed-transactions/ -d "cmd=all"
```

- Post [http://localhost:8089/v1/factoid-get-processed-transactionsj/\(.\\*\)](http://localhost:8089/v1/factoid-get-processed-transactionsj/(.*))

如果键入'all'，会返回所有交易。如果键入一个地址，会返回所有使用该地址作为输入信息，输出信息，或作为信用条目输出的交易，并以一串JSON对象的数组返回。例如：

```
curl -X POST http://localhost:8089/v1/factoid-get-processed-transactionsj/ -d "address=FA3RrKWJLQeDuzC9YzxcSwenU1qDzzwjR1uHMpp1SQbs8wH9Qbbr"
curl -X POST http://localhost:8089/v1/factoid-get-processed-transactionsj/ -d "address=<addrname>"
curl -X POST http://localhost:8089/v1/factoid-get-processed-transactionsj/ -d "cmd=all"
```