ストックマーケット

@DG Lab Karl-Johan Alm

いきなりなんだけど

実際にPeg-inしよう！先ずは設定を入れよう。

```
$ cd bc2/
$ cp elements/conference/*.conf <PATH>
macOS: <PATH> =
~/Library/Application\ Support/Bitcoin
linux: <PATH> = ~/.bitcoin
```

いきなりなんだけど

それでdaemonをスタートする。

Bitcoinも設定が変わったので、再起動する。./bitcoindを入れたターミナルに

<Ctrl> Cを押す。そしてもう一度

```
$ ./bitcoind -printtoconsole
```

追加いきなり

$ cd bc2/bc2/src
$ ./bitcoin-cli getbalance

0.0の場合:

$ ./bitcoin-cli getnewaddress
1…
^をコピーしてSlackに出して下さい

いきなりなんだけど

Elements:

```
$ cd bc2/elements/src
$ ./elementsd -printtoconsole
```

いきなりなんだけど

他のノードと接続する。新しいターミナルに:

```
$ cd bc2/elements/src
$ ./elements-cli addnode
67.205.138.199:10042 onetry
$ ./elements-cli getblockcount
```

400以上じゃないといけない

いきなりなんだけど

便利だからアリアスを作る。

```
$ cd bc2/elements/src
$ alias e-cli="./elements-cli"
$ alias
b-cli="../../bc2/src/bitcoin-cli"
```

いきなりなんだけど

```
ADDRS=$(e-cli getpeginaddress)
MAINCHAIN=$(echo $ADDRS | python3
-c "import sys, json;
print(json.load(sys.stdin)['maincha
in_address'])")
```

いきなりなんだけど

特別なP2SHアドレスにBC2コインを送る

```
$ TXID=$(b-cli sendtoaddress
$MAINCHAIN 10)
```

15ブロックを待ってからクレーム出来る。

```
$ b-cli getblockcount
```

いきなりなんだけど

```
$ PROOF=$(b-cli gettxoutproof
'''["''''$TXID''''"]''')
$ RAW=$(b-cli getrawtransaction
$TXID)
$ CLAIMTXID=$(e-cli claimpegin $RAW
$PROOF)
```

いきなりなんだけど

これで(30秒以内)エレメンツチェーンの方に10個のMainCoinが増えているはず。

```
$ e-cli getbalance
{
  "maincoin": 10.0,
}
```

Agenda

- Elementsのアセットを発行
- Partial transaction（オファー）
- Complete transaction (マッチング)
- タスク（興味のある方）

Elementsのアセットを発行

Elementsのアセットを発行

自分のアセットを発行して、他のユーザーとやりとりをしよう。

発行するには一回、エントロピーが必要。Elements-bc2の中ではmaincoinというアセットを使わないといけない。

Elementsのアセットを発行

発行の注意：アセット名を付ける。

アセット名がユニークじゃないと空っぽになる！

① Slackに「**アセット名：なんちゃら**」を書いて下さい！

他の人が同じアセット名を言い出したらその人と話し合う

Elementsのアセットを発行

② アセット名が決まったら発行する:

```
$ cd bc2/elements/src
$ ./elements-cli issueasset 1000 10 true アセット名
```

（アセットを1000個発行する）

Elementsのアセットを発行

発行したらアセットIDが出る。コピペ、ノートに入れる。

```
{
  "txid": "897693a4bbcd0c5...",
  "entropy": "20cbe5ab9380f...",
  "asset": "4ac594f84e672...",   ← これ
  "token": "18ddb96199..."
}
```

Partial transaction (オファー)

Partial transaction (オファー)

① 特別なトランザクションを作る。５個の太郎コインを１０個のDGコインで買いたいとしよう。これを

IN: １０個のDGコイン（自分のウォレットから）
OUT: ５個の太郎コイン（自分のウォレットへ）

というトランザクションで表す。これをブラインドして（隠す）、サインして、適当に公に出す。

Partial transaction (オファー)

② ５個の太郎コインを払って、１０個のDGコインを買いたい人がこのトランザクションを見つけたら、

IN: ５個の太郎コイン（自分のウォレットから）
OUT: １０個のDGコイン（自分のウォレットへ）

を足して、ブラインドして、サインしたら、トランザクションをネットワークに送信出来る。

Partial transaction (オファー)

①と②を重ねると、

IN:　　１０個のDGコイン（①のウォレットから）
IN:　　５個の太郎コイン（自分のウォレットから）
OUT:　５個の太郎コイン（①のウォレットへ）
OUT:　１０個のDGコイン（自分のウォレットへ）

になる。INとOUTがmatchingしているので、送信出来る。

Partial transaction (オファー)

やってみよう！

先ずは相手を見つける。隣の人でいいので。

三人でもOK。

相手を見つけたら、Slackかどこかでダイレクトメッセージを始めて下さい。

Partial transaction (オファー)

オファーを作ってみる。相手が良いと言ってくれそうな割合で

```
$ ./elements-cli makeoffer SELLASSET
SELLAMOUNT BUYASSET BUYAMOUNT
```

（ヒント：maincoinは色々と重要なので、使わない方が良い）

Partial transaction (オファー)

一つ残念だが、両方が両方のアセットを持たないと出来ないので、先ずは0.0000001くらいを相手にあげちゃおう。

```
$  ./elements-cli getnewaddress
```

結果を相手に送る。

Partial transaction (オファー)

```
$ ./elements-cli sendtoaddress <貰ったアドレス>
0.0000001 "" "" false <自分のアセットのＩＤ>
```

これで取引が出来る様になる。

Partial transaction (オファー)

今度は、相手にオファーを送る。でっかいので

```
[+] > Code or text snippet
```

という（Slack内の）機能を使おう。

相手から貰ったものをダウンロードして、
`$HOME/bc2/offer.txt`にコピー。

Partial transaction (オファー)

トランザクションの中身を見ても何も分からないので、相手に自分が何個を入れて、何個を出しているか教える。

ここで質問：嘘をついたらどうなる？

① 送信する時に、嘘のままの取引が行われる
② 送信する時に、本当のままの取引が行われる
③ 送信出来ない

Partial transaction (オファー)

トランザクションの中身を見ても何も分からないので、相手に自分が何個を入れて、何個を出しているか教える。

ここで質問：嘘をついたらどうなる？

① 送信する時に、嘘のままの取引が行われる
② 送信する時に、本当のままの取引が行われる
③ 送信出来ない

Partial transaction (オファー)

嘘つきが①。10個のDGコインと言ったけど5個しか入れてない。

IN: 5個のDGコイン（①のウォレットから）
IN: 5個の太郎コイン（自分のウォレットから）
OUT: 5個の太郎コイン（①のウォレットへ）
OUT: 10個のDGコイン（自分のウォレットへ）

5個のDGコインオーバーなので送信したらエラー。

Partial transaction (マッチング)

Partial transaction (マッチング)

相手から貰った情報が良いと思うなら完成させる。

```
$ ./elements-cli matchoffer SELLASSET
SELLAMOUNT BUYASSET BUYAMOUNT $(cat
$HOME/bc2/offer.txt) > complete.txt
```

Partial transaction (マッチング)

出てくるHEXを送信すれば取引が実際に行われる。

```
$ ./elements-cli sendrawtransaction $(cat complete.txt)
```

Partial transaction (マッチング)

このトランザクションは以前言った様にちょっと特別。

普通の（現在の）bitcoinでは作れない。

・SIGHASH_SELECT*は使われている。
　サインした後に、普通は変えられない。
　IN/OUTを選択（SELECT）すると、他を変えてもOK。

Partial transaction (マッチング)

ブラインディング（隠す事）は、普通のElementsでは（今）INPUTが変わらない前提で動いているが、これはElements-bc2に直された。（重ねる事は許されるが、変える事は許されない）

タスク（興味あったら）

ストックマーケットを作ろう！

このセッションでやっていた事が結構面倒くさい。

アセット、txのhexを教えあったりする必要がある。

ウェブサイトにサービスを提供し、オファーのリストと作成（送信）とマッチングの機能を作ろう。

（やろうと思ったら是非相談下さい！Slack: kalle）

ストックマーケットを作ろう！

例えば:

売る		買う	
アセット	数	アセット	数
太郎コイン5		DGコイン	10
FooCoin	400	BarCoin	1

…


BC2
Blockchain Core Camp


DG
LAB