シーズン1＆2

@DG Lab Karl-Johan Alm

Agenda

- 事前準備の確認
- シーズン1の復習
- シーズン2の紹介

事前準備の確認

事前準備の確認

- Slackに登録、サインイン
- BC2ネット
- Elements-bc2ネット

https://bc-2.jp/sde.html

シーズン1の復習

Bitcoin Core Contributorになろう

テーマはビットコインの細かい部分を理解してもらい、日本のコントリビューターを増やす事だった。

・UTXOモデル、トランザクションの構成・作り方

・コンセンサスとOrphan Blocks

・BIPプロセス

事前に知っておいた方が良い事

前回は何も知らない前提で計画を立てたが、今回は知っているだろうと想定できる事がいくつかある。

・UTXOモデル

・トランザクションの構成

・CLIの使い方

シーズン2の紹介

BC2ネット

事前準備に書いている通り、今回は二つのネットがある：

① BC2 (bitcoinの弄ったネットワーク、マイニング）

② Elements-bc2 (elementsの弄ったネットワーク、フェデレーション）

BC2ネット

Proof of WorkをSHA256dではなく、Cuckoo-Cycleに。

Smooth difficulty adjustment (マイニングの難度の調整をもっとスムーズに[1])

[1] https://scalingbitcoin.org/milan2016/presentations/Scaling%20Bitcoin%203%20-%20Mark%20Friedenbach.pdf

「MainCoin」

BC2ネットのBC2コイン→Elements-bc2のMainCoin (peg-in).

BC2のコインをマイニングする必要がある。

```
$ cd bc2/bc2/src
$ ./bitcoin-cli generate 1
```

スマホアプリ

Multi-Asset Wallet付きのスマホアプリ（iPhone / Android）が配られている。

Android user: http://bc-2.jp/tools/bc2app.apk

iPhone user: https://appsto.re/i6dY7PL

スマホアプリ

Faucetで自分のウォレットにチャージして下さい。

One-time codeを貰っているのでそれを使ってやって下さい。（アプリの「コピー」機能でアドレスを入力）

・Lunch Tokenで明日（Day 2）に昼ご飯を買う

・自分のアセットを作ってみる（今日の最後）

Elements-bc2ネット

新しいopcode：OP_CHECKMERKLEBRANCH: Merkle branchの検証

新しいscript機能：tail eval (P2SHっぽく)

新しいサイン方法：SIGHASH_SELECTIN|OUTPUTS

新しいRPCコマンド：makeoffer, matchoffer

OP_CHECKMERKLEBRANCH

スマートコントラクトのセッションで詳細を説明する

```
if (cond1) {
  exec1();
  if (cond2)
    exec2();
  else
    exec3();
} else …
```

OP_CHECKMERKLEBRANCH

スマートコントラクトのセッションで詳細を説明する

```
if (cond1) {
  exec1();
  if (cond2)
    exec2();
  else
    exec3();
} else …
```

① cond1
exec1
cond2
exec2

OP_CHECKMERKLEBRANCH

スマートコントラクトのセッションで詳細を説明する

```
if (cond1) {
  exec1();
  if (cond2)
    exec2();
  else
    exec3();
} else …
```

①	②
cond1	cond1
exec1	exec1
cond2	!cond2
exec2	exec3

OP_CHECKMERKLEBRANCH

スマートコントラクトのセッションで詳細を説明する

```
if (cond1) {
  exec1();
  if (cond2)
    exec2();
  else
    exec3();
} else …
```

① cond1 exec1 cond2 exec2

② cond1 exec1 !cond2 exec3

③ !cond1 …

OP_CHECKMERKLEBRANCH

スマートコントラクトのセッションで詳細を説明する

```
if (cond1) {
  exec1();
  if (cond2)
    exec2();
  else
    exec3();
} else …
```

① cond1 / exec1 / cond2 / exec2 / HASH

② cond1 / exec1 / !cond2 / exec3 / HASH

③ !cond1 / … / HASH

-> MerkleTree

Tail-eval (P2SHっぽく)

スマートコントラクトのセッションで詳細を説明する

スクリプトの終わりにスタックに残っているデータがあれば、それを実行。

これと、OP_CHECKMERKLEBRANCHがあれば、MAST (Merkelized Abstract Syntax Trees) が出来る。

(https://github.com/jl2012/bips/blob/mast/bip-mast.mediawiki)

SIGHASH_SELECTIN|OUTPUTS

サインする時、SIGHASH_SELECTINPUTS や OUTPUTSというフラグを使えば、何を証明するかを選択出来る。

ストックマーケットセッションで使用例を紹介する。

（別々に作成されたトランザクションを重ねる事が出来る…！）

RPC: makeoffer, matchoffer

今日の最後のセッションに詳細を説明する。

makeoffer → 「人参3本をポテト2個に変えたい」

matchoffer → 「ポテト2個を人参3本に合わせて変える」


BC2
Blockchain Core Camp


DG
LAB