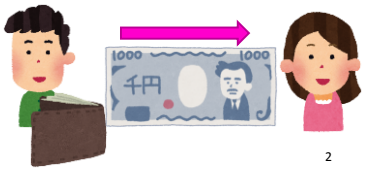
地域経済情報研究会

ビットコインの仕組みと問題点

経営学部　浮穴学慈

お金を支払うということ

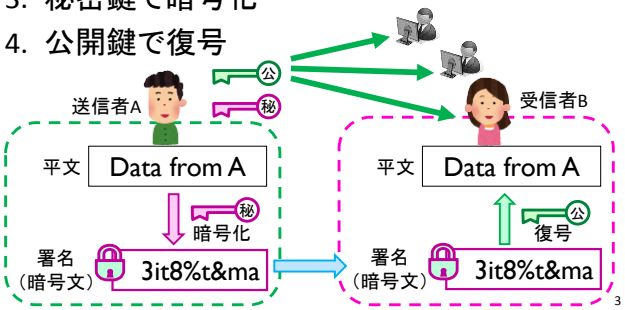
- 貨幣＝価値尺度、価値貯蔵、流通手段
- 所有権・価値の移転
  - 現金：銀行券等の直接の受渡しが、移転を表す
  - Suicaなど：発行主体が、移転情報を管理
- 集権的管理者なしで、移転を管理できるのか
  - 盗難（所有者が了承しない移転）への対策
  - 二重取引への対策
  - 通貨偽造への対策



2

本人の了承といえば、電子署名 ECDSA

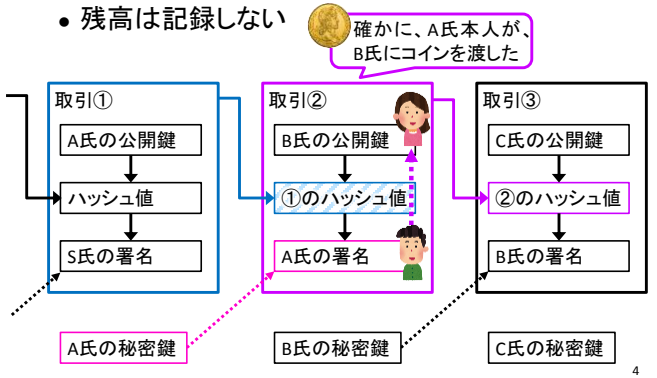
1. 送信者は、公開鍵と秘密鍵のペアを生成
2. 公開鍵（復号専用）を公開・配布
3. 秘密鍵で暗号化
4. 公開鍵で復号



3

移転を表すハッシュチェーン・データ構造

- 所有権の移転処理を取引（transaction）という
  - 残高は記録しない



4

ハッシュ値とは

- 暗号学的ハッシュ関数により計算される値
  - ハッシュ値を知っても、元のデータを計算不可能
  - 同じハッシュ値を生む、他のデータを計算不可能

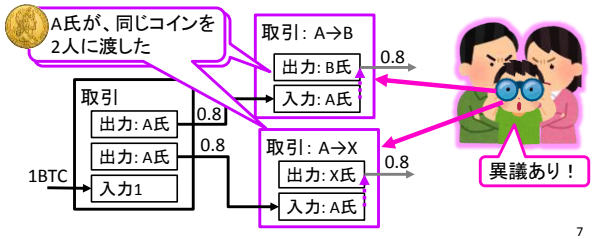
例： SHA-256

- ほぼ同じ例文だが、ピリオドを加えるだけで、ハッシュ値は大幅に変化する
  - ▸ The quick brown fox jumps over the lazy dog
    → D7A8FBB3 07D78094 69CA9ABC B0082E4F 8D5651E4 6D3CDB76 2D02D0BF 37C9E592 （256 bit（32 Byte））
  - ▸ The quick brown fox jumps over the lazy dog.
    → EF537F25 C895BFA7 82526529 A9B63D97 AA631564 D5D789C2 B765448C 8635FB6C

5

二重取引を防ぎたい

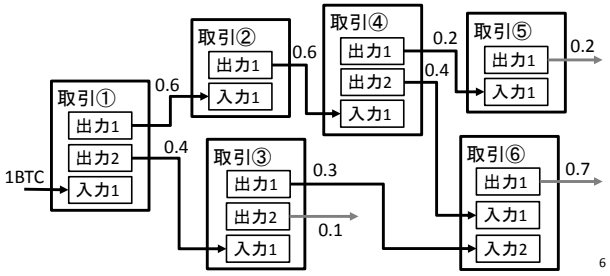
- 取引を公開し、矛盾が無いか皆で確認したい
  - P2Pネットワークでは、取引の先後を完全に特定することは不可能
  - 同じコインの多重取引は、どれか1つだけが、全体にとって一貫した「正しい取引」であれば良い



7

お釣りをどうするか

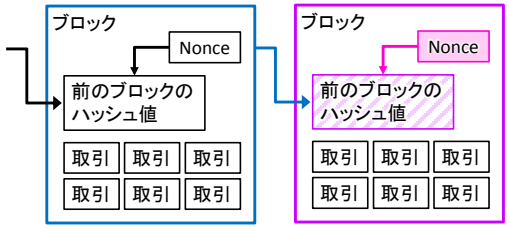
- 1個の取引の内に、複数の入力と複数の出力を設定できる
  - コインの分割や合算を可能にしている
  - コインを分割して自分にも渡せば、お釣りになる



6

二重取引を防ぐブロックチェーン

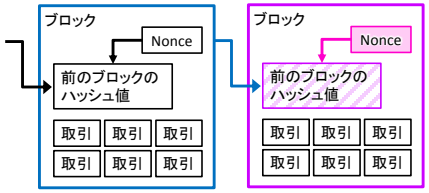
- 全取引履歴を公開帳簿として皆で共有
  - 「正しいブロック」に含まれる取引だけを「正しい取引」とし、確定させる（≒/≠コミット）
  - ブロック内の取引の正当性を皆が確認できる
  - 以前のブロックの取引改竄は、今のところ不可能



8

「正しいブロック」とマイニング

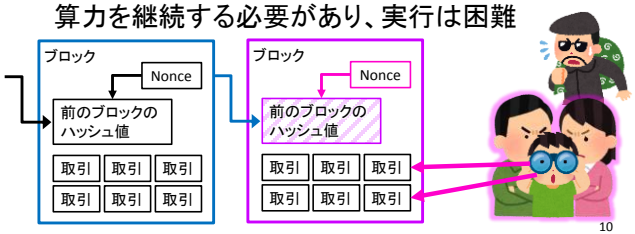
- 「正しいブロック」は、先頭に0が並ぶハッシュ値を持つ必要がある
  - 条件に合うNonce値を発見することでブロックが生成される。これをマイニングという
  - Nonce値の発見には、相当の計算力が必要で、約10分に1個作られるよう制御されている



9

「正しいブロック」の承認（PoW）

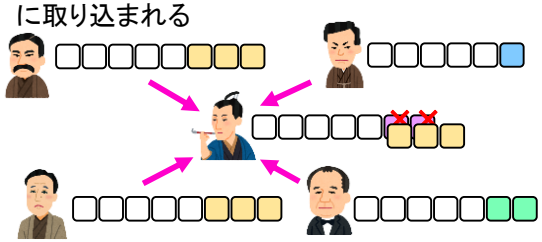
- ブロックの生成が報告されると、ブロック内の全取引の正当性を各自が確認し、承認する
  - 承認した「正しいブロック」は、各自のチェーンの最後尾に接続される
  - 「不正なブロック」を承認させるには、過半数の計算力を継続する必要があり、実行は困難



10

ブロックチェーンの分岐と再編成

- ある時点におけるチェーンのデータは、全員が同じとは限らない
  - ブロック生成は、並列的に同時進行で試行される
- 最長チェーンを採用し、負けたブロックは解消
  - 解消されたブロック内の取引は、新しいブロックに取り込まれる



11

ビットコインの問題点

- スケーラビリティ問題
  - 取引の確定（決済）に時間が掛かる（10分～）
  - 通信量、全取引履歴のデータ量の増加
  - 必要な計算力（＝電力消費）の増加と寡占化
- トランザクション展性問題（マリアビリティ問題）
- コイン分裂直後など、少人数下の51%攻撃
- 本質的にコインの分裂が避けられない
  - ハッシュアルゴリズム SHA-256 の寿命と SHA-3 への世代交代
    - MD5やSHA-1は既に破られ、同じ Merkle-Damgård 構造方式のSHA-256も時間の問題（Li, 2012）（Laie, 2017）
    - 電子署名 ECDSA Secp256k1 のほうは、大丈夫
  - CAP定理

12

トランザクション展性問題

- 展性（Malleability）
- 攻撃用取引データを用意
  - まだブロックに取り込まれていない取引データを取得
  - 同じ意味を持ち書き方の異なる署名スクリプトへ書き換え、それによりハッシュ値が本物と異なる
- 本物の取引より先にブロックに取り込まれ、承認されることがある
  - 取引が完了していないと、支払元に誤認させることができる（支払元がハッシュ値のみで確認の場合）
- 再度の請求に支払元が応じると二重支払になる

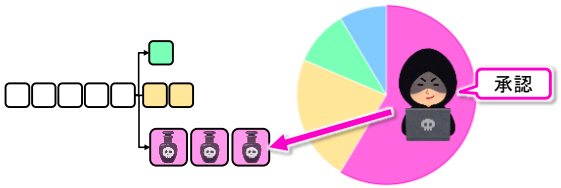
例： コイン交換所 Mt. Gox の被害（2014.2）

13

51%攻撃

- 過半数の計算力を占める必要があり、実行困難なはずの「不正なブロック」を承認させることが可能な状況がある
  - ブロック作成の利益が減り、参加者が減る将来
  - ハードフォーク直後（後述）

例：ビットコインゴールド（BTG）（2018.5）



14

ハードフォークとソフトフォーク

- 仕組みのアップデートによるチェーンの分裂
  - 新旧両方のコインが二重に利用可能な状態に
- ハードフォーク
  - これまでとは互換性がない。古いチェーンは残して、分裂以降は新しいルールに従って作成

  例： ビットコインキャッシュ（BCH）
  - ▸ブロックサイズの拡大
  - ▸スマートコントラクト技術
    （自動販売機のように、契約と決済を自動化し、クラウドファンディングなどに利用できる）
- ソフトフォーク
  - これまでと互換性があり、古いチェーンとも連結

15

UASF（User-Activated Soft Fork）

- ブロック作成者（マイナー）によるチェーン分岐ではなく、ユーザ主導でのチェーン分岐

例： SegWit （Segregated Witness）
- SegWit非対応のブロックは、不正ブロックとみなすという新ルール（BIP148, 2017.8）
- P2Pなので、新ルールを強制することは不可能
- 旧ルール遵守側にとって新ブロックは正常なブロック

- その後の可能性
  - 新側が優位：何の問題もなく、旧側を吸収する
  - 新旧が拮抗：分岐が頻発し、取引確定が遅延
  - 旧側が優位：将来、新側が優位になったら……

16

CAP定理(ブリュワーの定理)

- 分散データベースでは、次の3つを同時に満たすことは不可能(予想Brewer 2000, Gilbert, Lynch 2002)
  - ① 一貫性(無矛盾性)(Consistency)
  - ② 可用性(Availability)
    - ▸ 電源OFFや故障などで一部の計算機が停止しても、他の生きている計算機は、正常に応答をする
  - ③ 分断耐性(Partition-Tolerance)
    - ▸ ネットワークが複数の部分に分断しても、各部分が継続して機能し続ける(互換性のない新旧ルール分裂も分断)
- ビットコインは、分散データベースであり、通貨として機能するためには、①と②は必須
  - つまり、③を犠牲にするしかない

17

まとめ

- P2Pシステムでは、機能のアップデートに伴うネットワーク分断が避けられない
- CAP定理により、分断された部分のどれかは、機能を維持できない
- その部分で実施された取引の「確定」は、巻き戻される
- どこまでの期間の内容が巻き戻されるか、分からない
- 現金取引など他の決済手段でも、同様のこと(錯誤無効など)は発生するが……

18

主な参考資料

- 赤羽ら,「ブロックチェーン -仕組みと理論」(リックテレコム), 2016.10
- 安田ら, "暗号学的ハッシュ関数" 電子情報通信学会 Fundamental Review Vol.4 No.1 pp57-67, 2010.7
- 竹井, "ビットコインで使われている暗号の基礎" ヒカラボ, 2017.5
- Li et.al, "Converting Meet-in-the-Middle Preimage Attack into Pseudo Collision Attack: Application to SHA-2" IACR FSE2012, 2012.3
- 斉藤, "ビットコインにおけるトランザクション、その展性と影響" WIDE Tec-Report, 2014.5
- Lynch et.al, "Conjecture and the Feasibility of Consistent, Available, Partition-Tolerant Web Services" ACM SIGACT, Vol.33-2, pp. 51-59, 2002.6

19