ブロックチェーンがもたらす情報と産業への革命

2016年5月18日

国際大学グローバル・コミュニケーション・センター
研究部長／准教授／主幹研究員

高木聡一郎

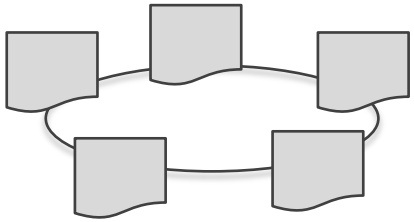
自己紹介

- 国際大学GLOCOM　研究部長／主幹研究員／准教授
- GLOCOM　ブロックチェーン経済研究ラボ代表
- 専門は情報経済学
  - 研究ドメインは「情報技術×経済学」
  - オフショア開発と雇用・生産性
  - クラウドコンピューティングのマクロ経済への影響
  - ITと組織形態（オープンデータ、マスコラボレションなど）
- 情報経済学から見たブロックチェーン

ブロックチェーンの考え方を用いてイノベーションを起こしていきたい！

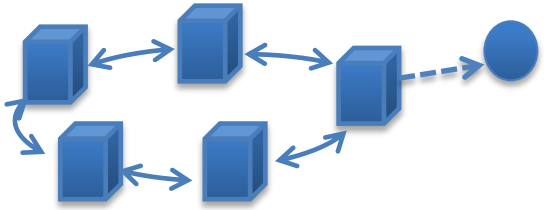
はじめに

インターネット



情報のネットワーク

ブロックチェーン



主体のネットワーク

- ブロックチェーンは、単なる情報のネットワークではなく、**主体間の関係、主体とモノ（資産）の関係**を定義する
- 情報の流通管理のための**経済的インセンティブ**を作り出す
- **多様な経済圏を作り出すことのできる**画期的・破壊的なイノベーション

目次

1. ブロックチェーンとは何か
2. 暗号の基礎知識
3. ブロックチェーンの全体構造
4. ブロックチェーン2.0へ
5. ビジネスにどう使うか

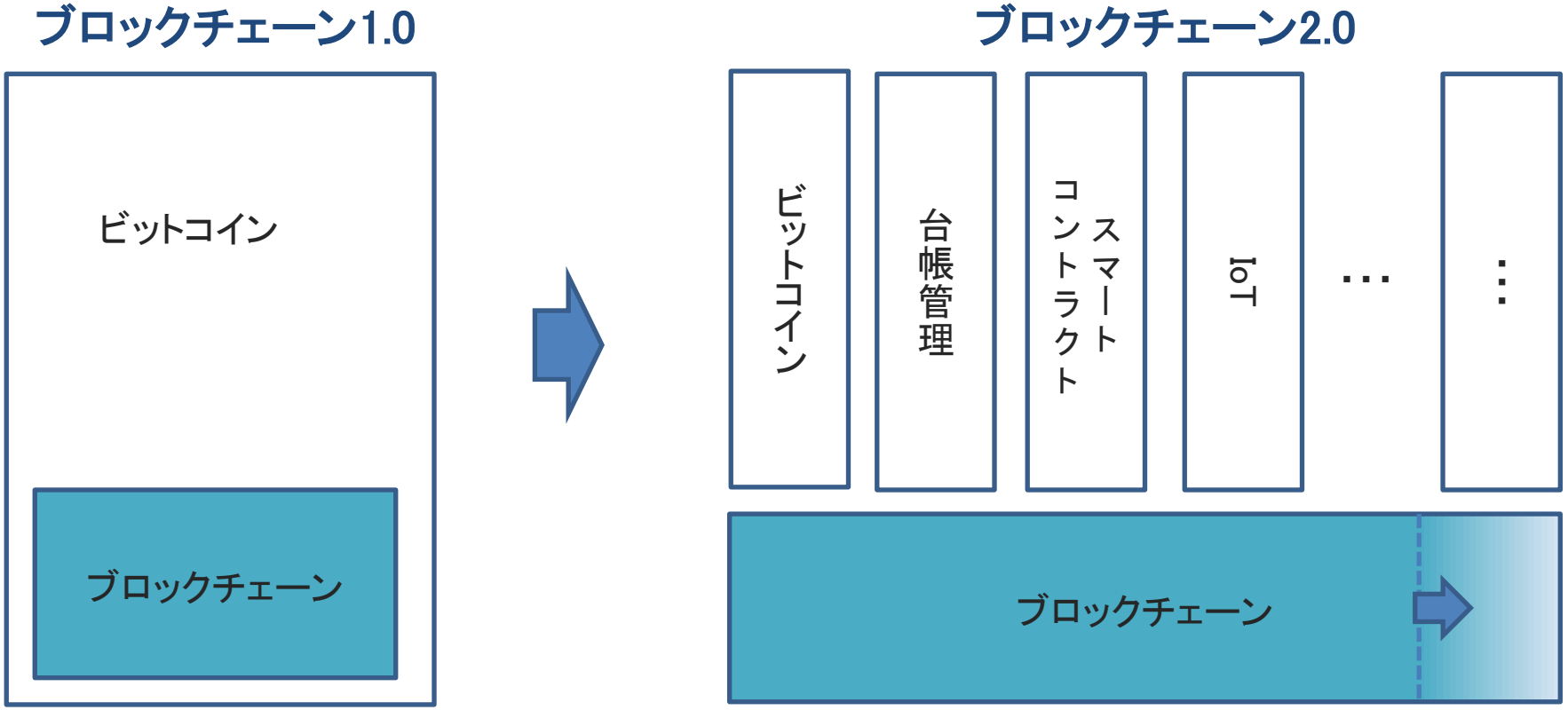
ブロックチェーンとは何か

ブロックチェーンとは何か

- 創生ブロックから始まり、先行ブロックに**延々とつながっている**認証されたブロックのリスト（Antonopoulos）
- これまでに実行された全てのビットコイン取引に関する、公開された**台帳**（Swan）
- 多くのレコードを単一のシートに集めるのではなく、ブロックに入れる形の**データベース**（英国政府レポート）
- **主体に紐づく取引データが連結・凝縮された一連の電子ファイルである（高木）**

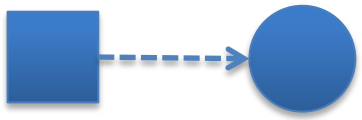
多種多様な定義が存在している。なぜか？

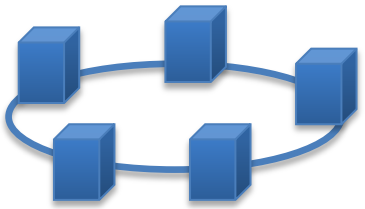
位置づけが変わりつつある ブロックチェーン



アプリケーションもプラットフォームも進化中

ブロックチェーンの3大要素





データの**連結**

＋

情報資産と
エンティティの紐付け

＋

P2Pでの
データ管理

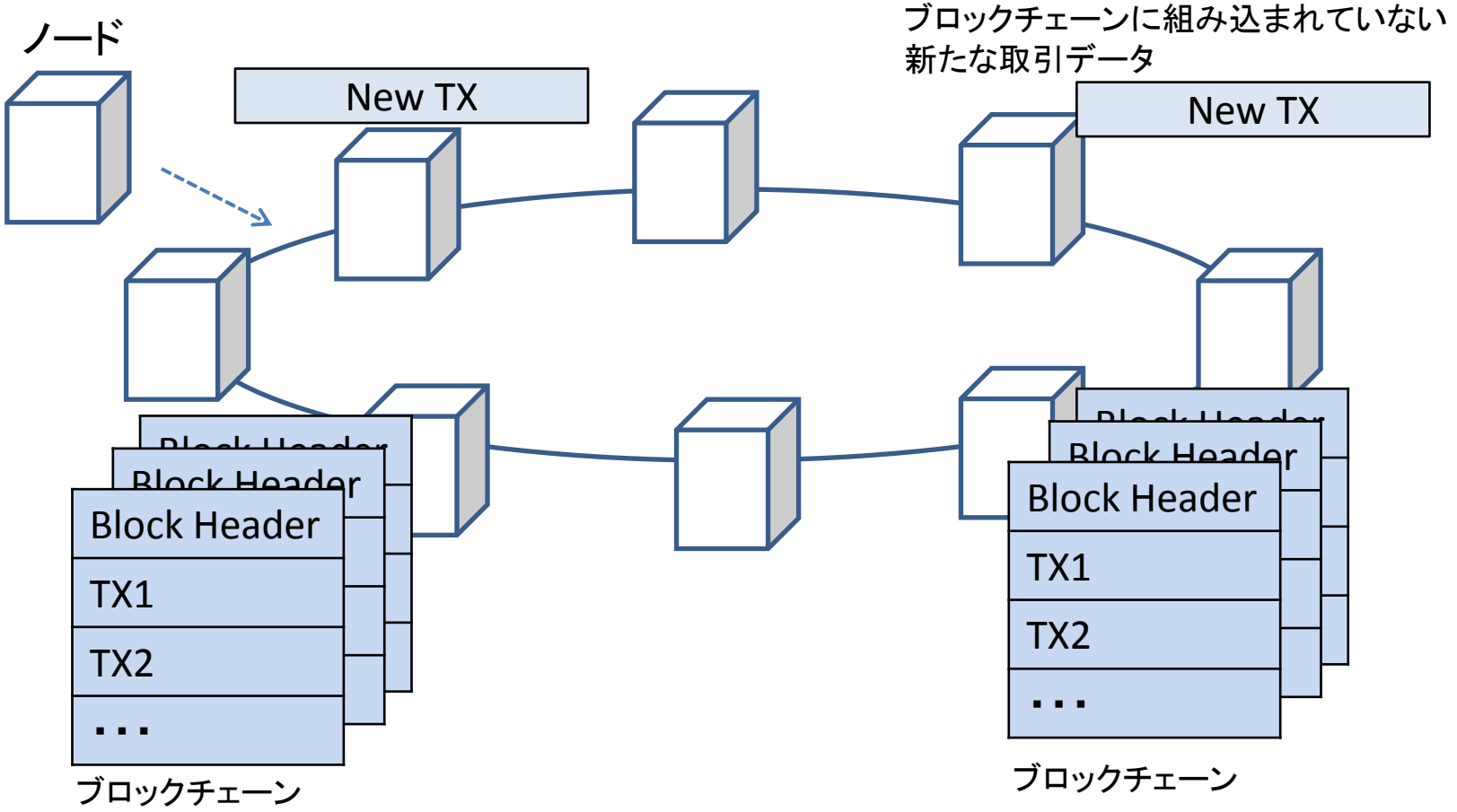
改ざん困難

情報の所有、流通、用途の管理

信頼性向上
中央管理者不要

分散型台帳とも呼ばれる

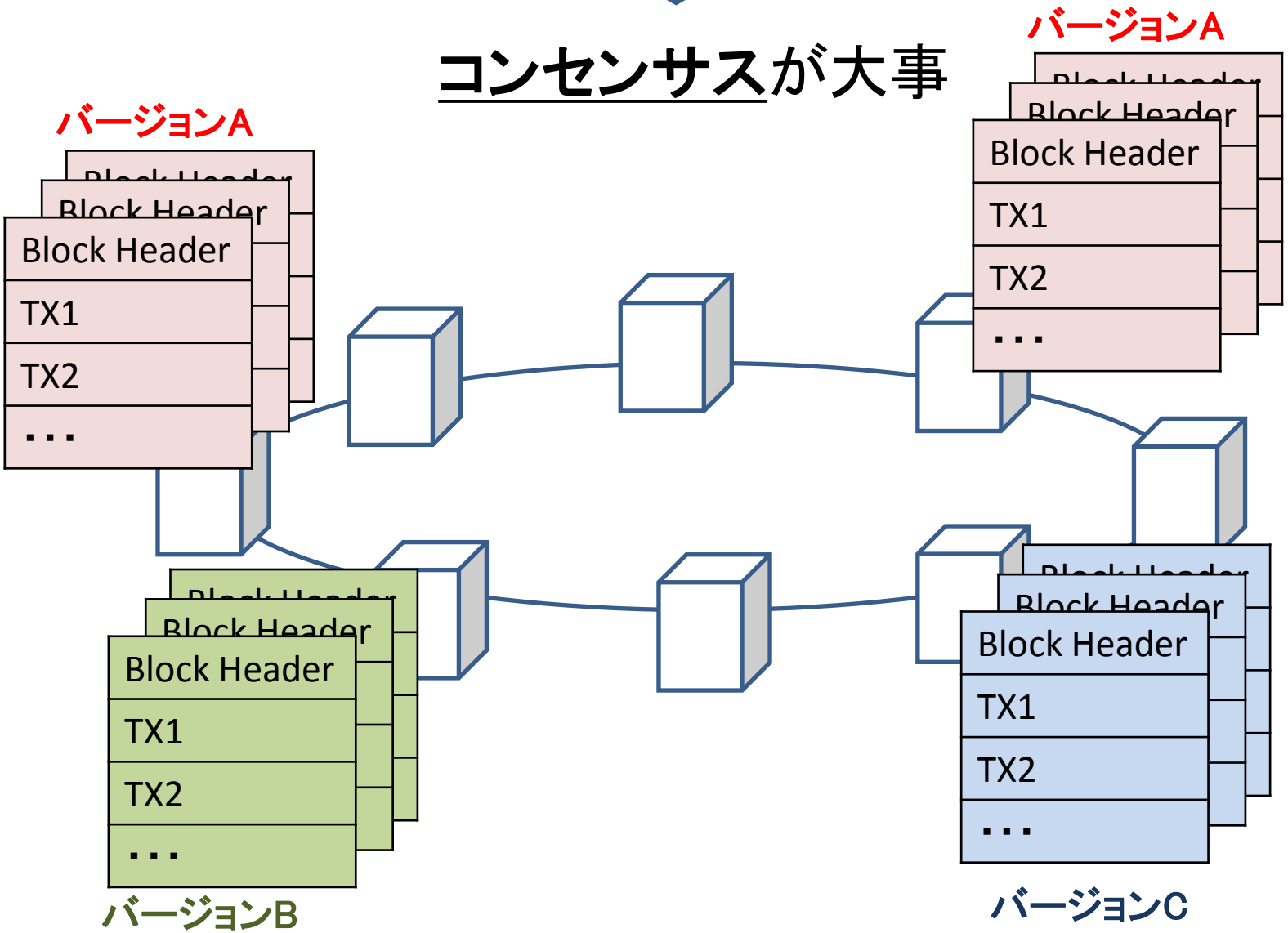
（Distributed Ledger）



- P2Pに参加しているノードが分散（**重複**）してデータを保管
- 公開＝**誰でも**データを追加・編集できる

どの台帳を正しいとするか？

コンセンサスが大事



暗号の基礎知識

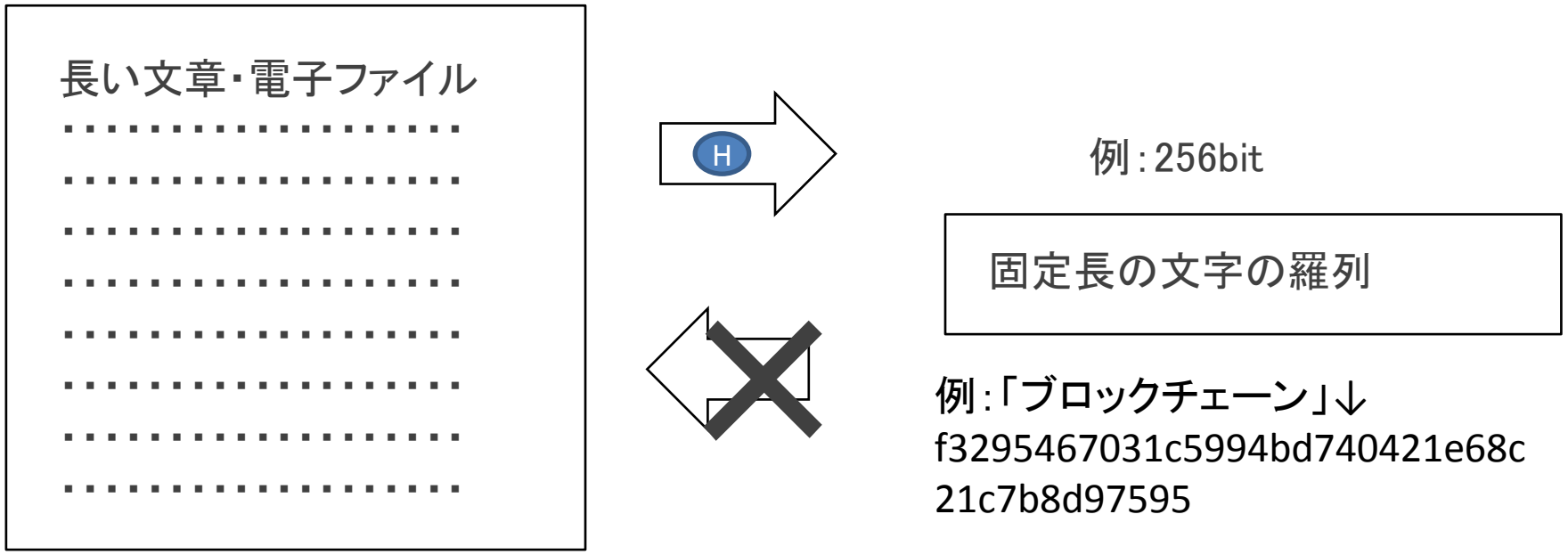
前提知識

5つのキーワードを理解すれば大丈夫です！

① ハッシュ関数
② 共通鍵暗号
③ 公開鍵暗号による暗号化
④ 公開鍵暗号による署名
⑤ 電子署名

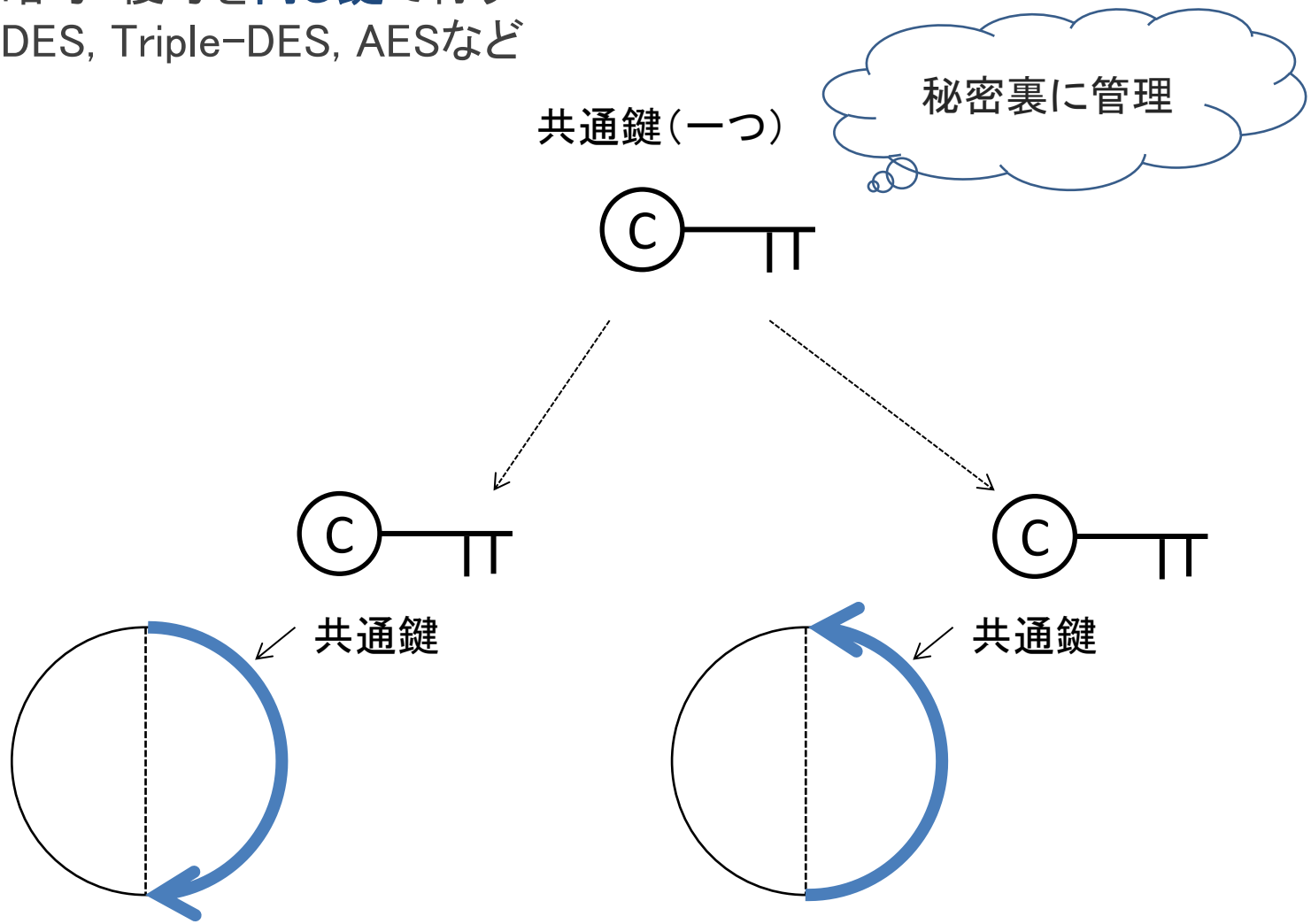
前提知識①：ハッシュ関数

- どんな大きさの情報も**短い文字列に圧縮（但し、元には戻せない）**
- どんな文字の羅列になるかは**予測不可能**
- 同じ元ファイルからは**必ず同じハッシュ値**が生成される
- **元の情報が少しでも違えば異なるハッシュ値になる**
- SHA−1, MD5, **SHA−256**



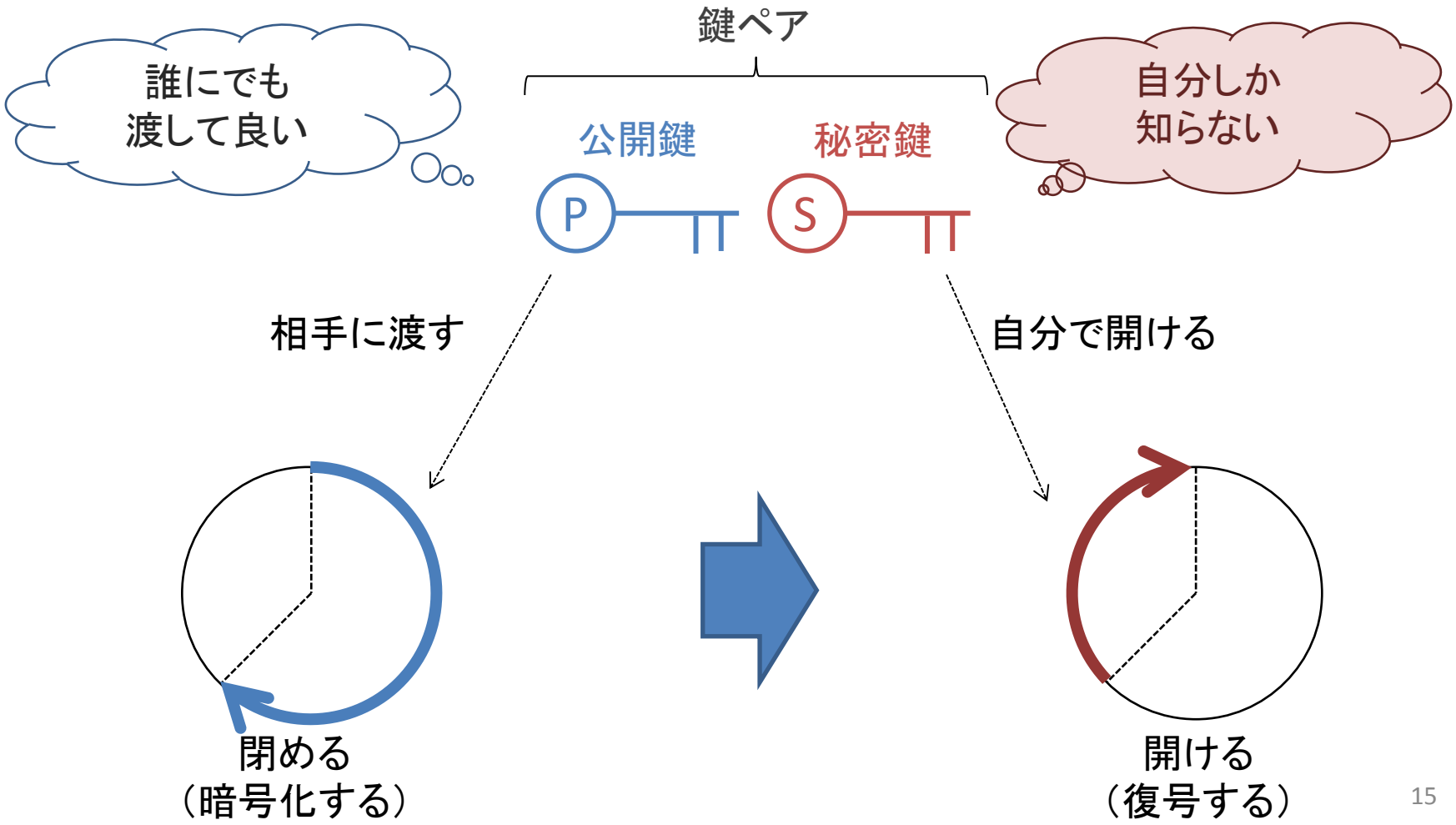
前提知識②：共通鍵暗号

- 暗号・復号を**同じ鍵**で行う
- DES, Triple-DES, AESなど



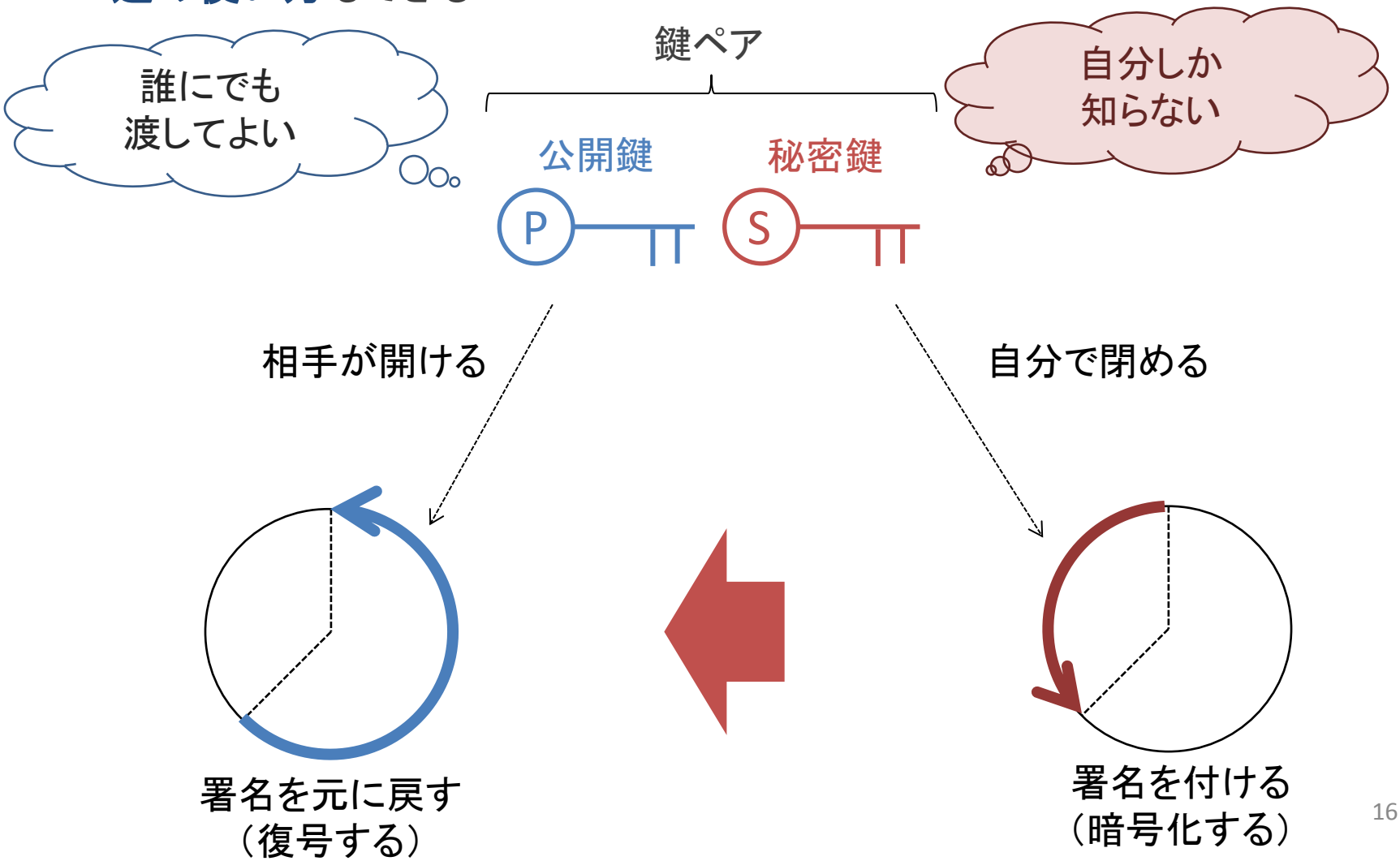
前提知識③：公開鍵暗号による暗号化

- 暗号・復号に**異なる鍵**を使う
- RSA, ECDSA（楕円曲線暗号）



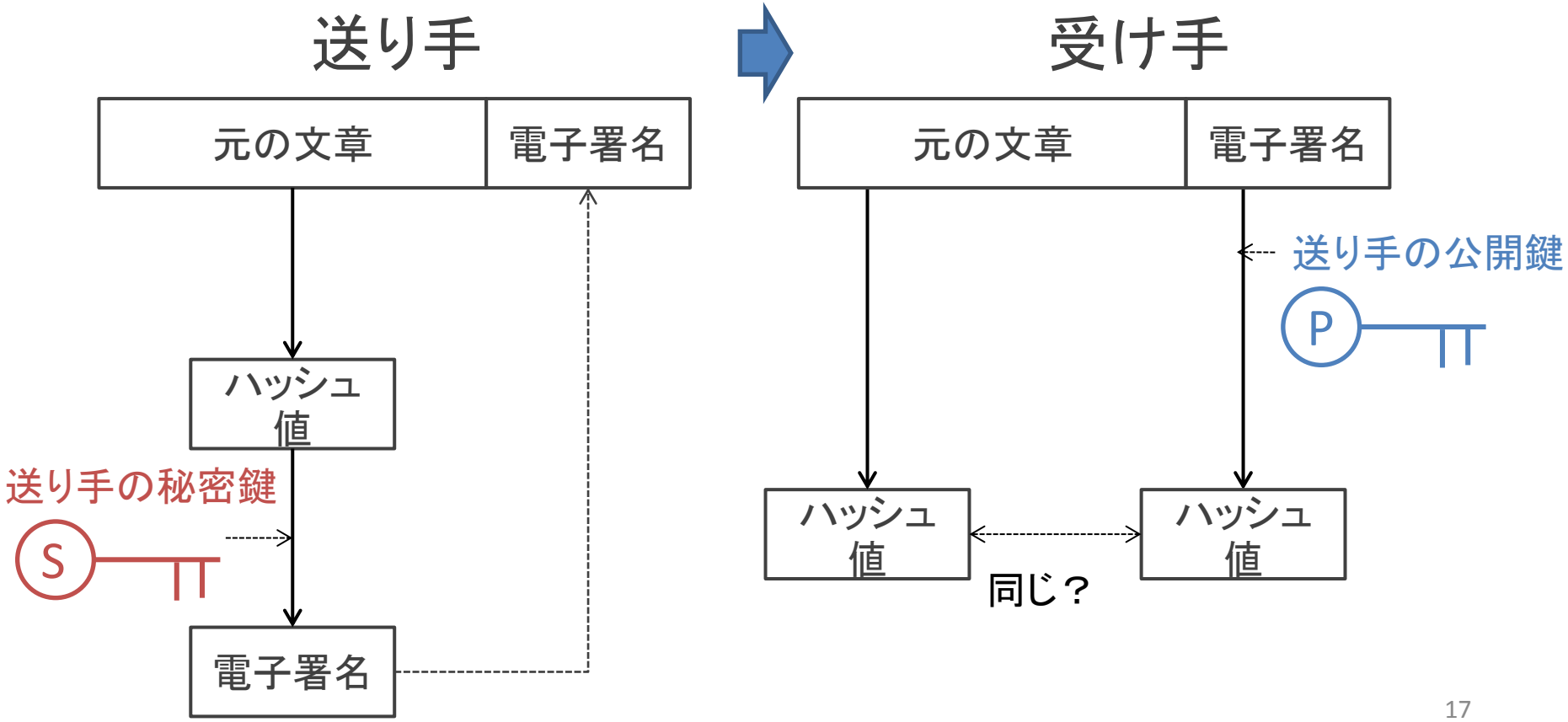
前提知識④：秘密鍵による署名（逆回し）

- **逆の使い方**もできる！



前提知識⑤：電子署名

- 元の文書が**改ざんされていない**こと、**本人が作成**したことを確認
- 本人＝対応する秘密鍵を持っている

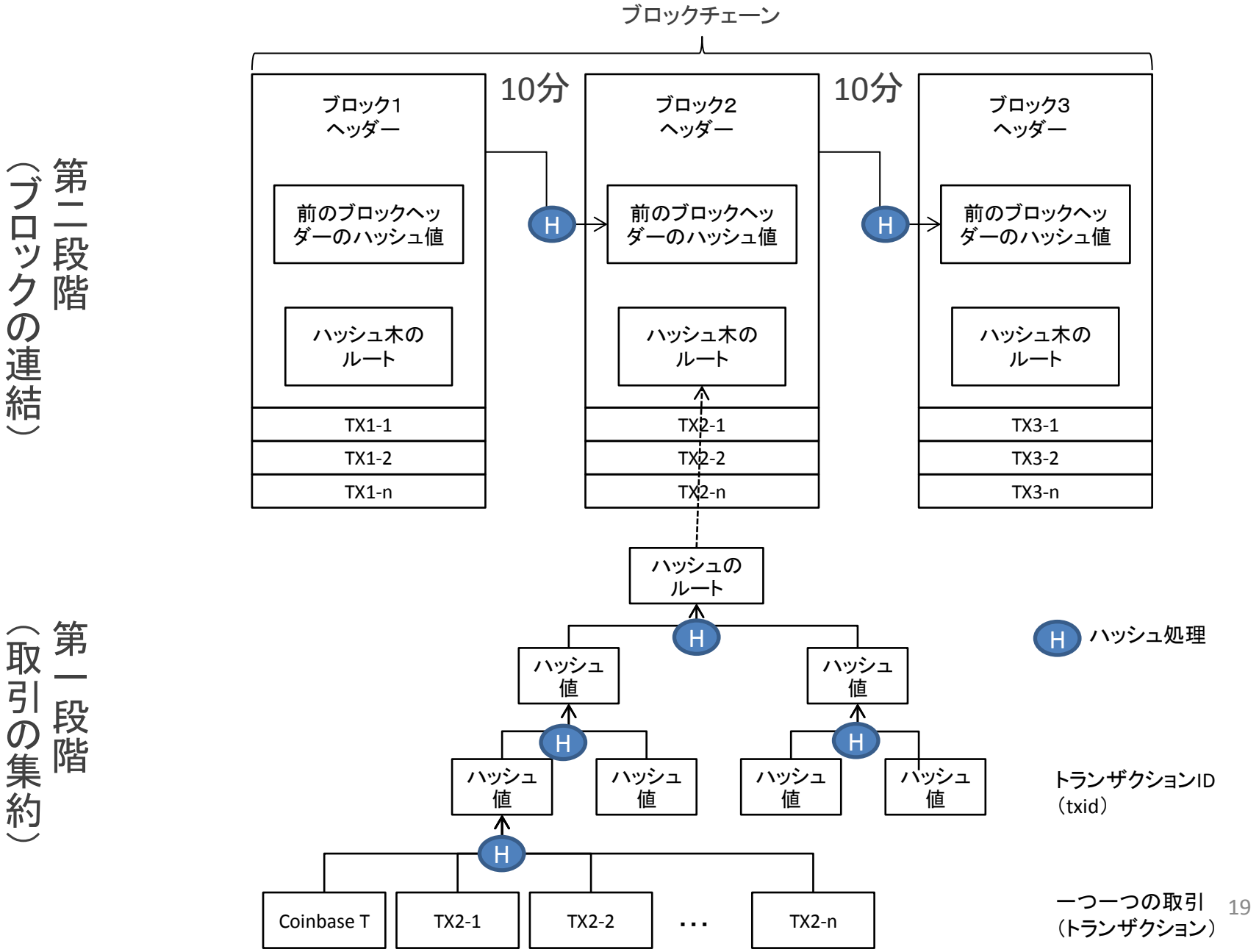


それではいよいよ・・・

ブロックチェーンの全体構造

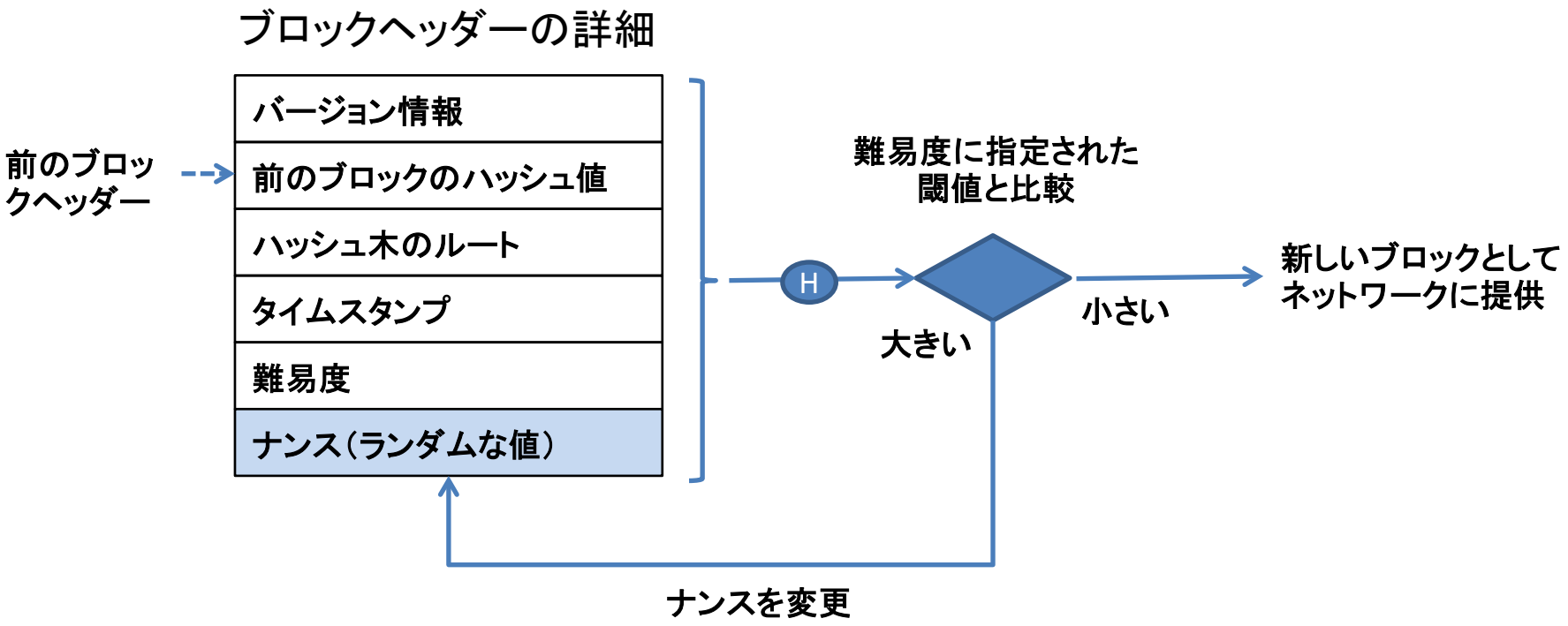
（ビットコインを題材に）

ブロックチェーンの全体像：2段階処理



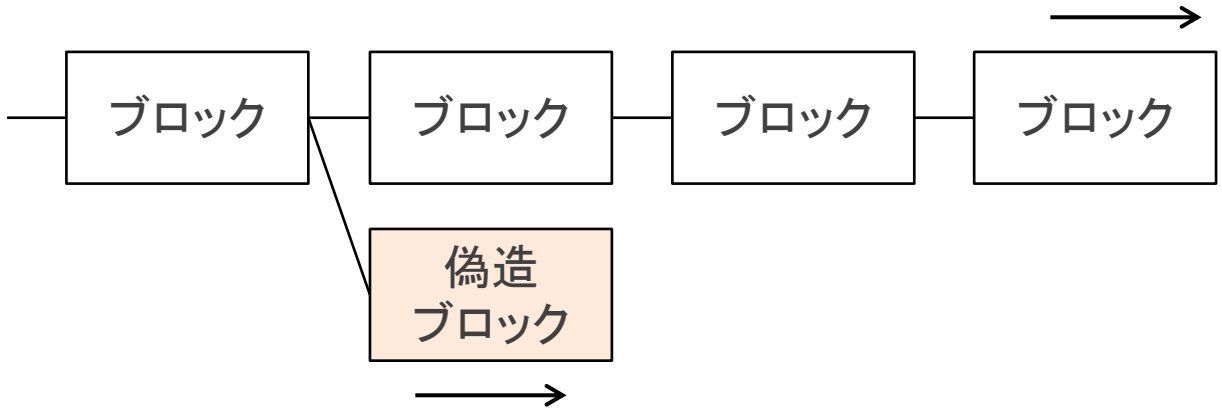
Proof of Work

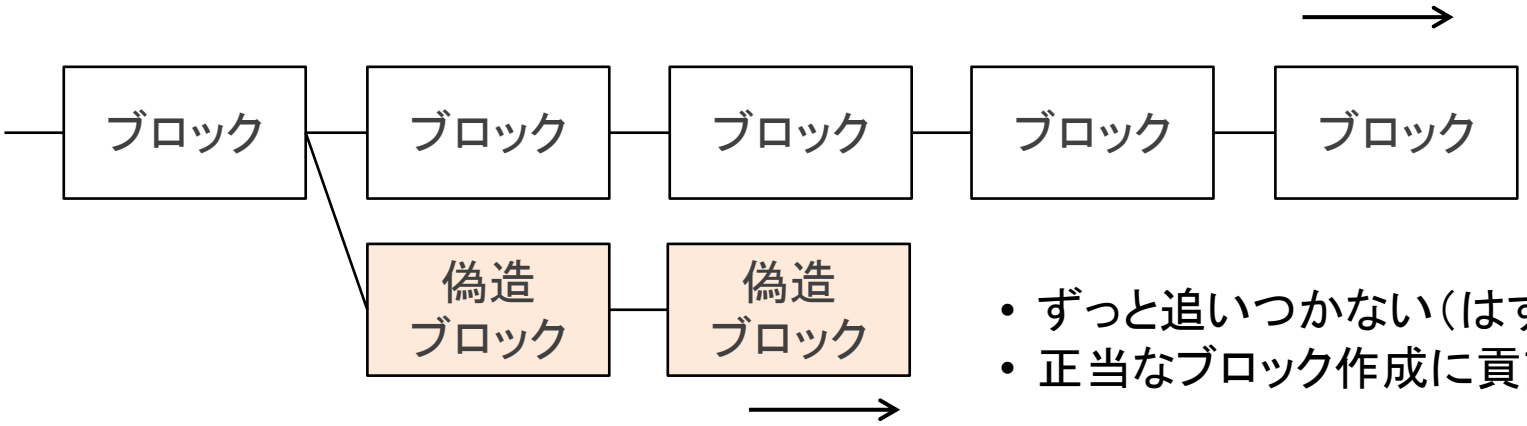
- ブロックができるまで**あえて**時間（労力）がかかるように設定
- **閾値**をクリアするまで繰り返す
- 平均すると**10分**で閾値をクリアするよう設定
- 新ブロックを作成した人には**新規のビットコイン（現在は25BTC）**が発行される



※当日の資料から修正

なぜPoWをかけるのか

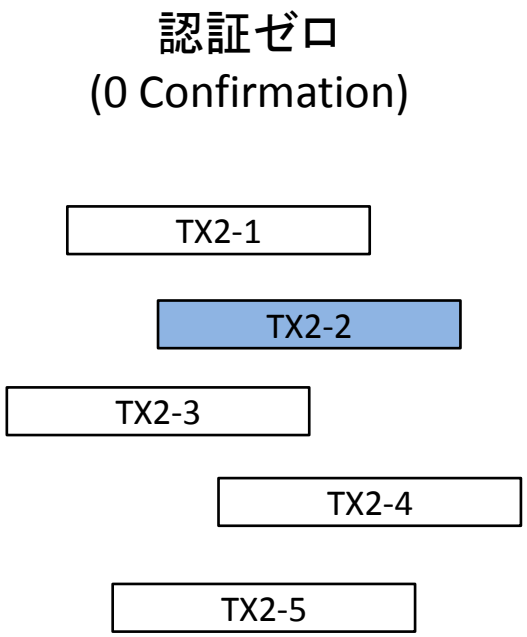




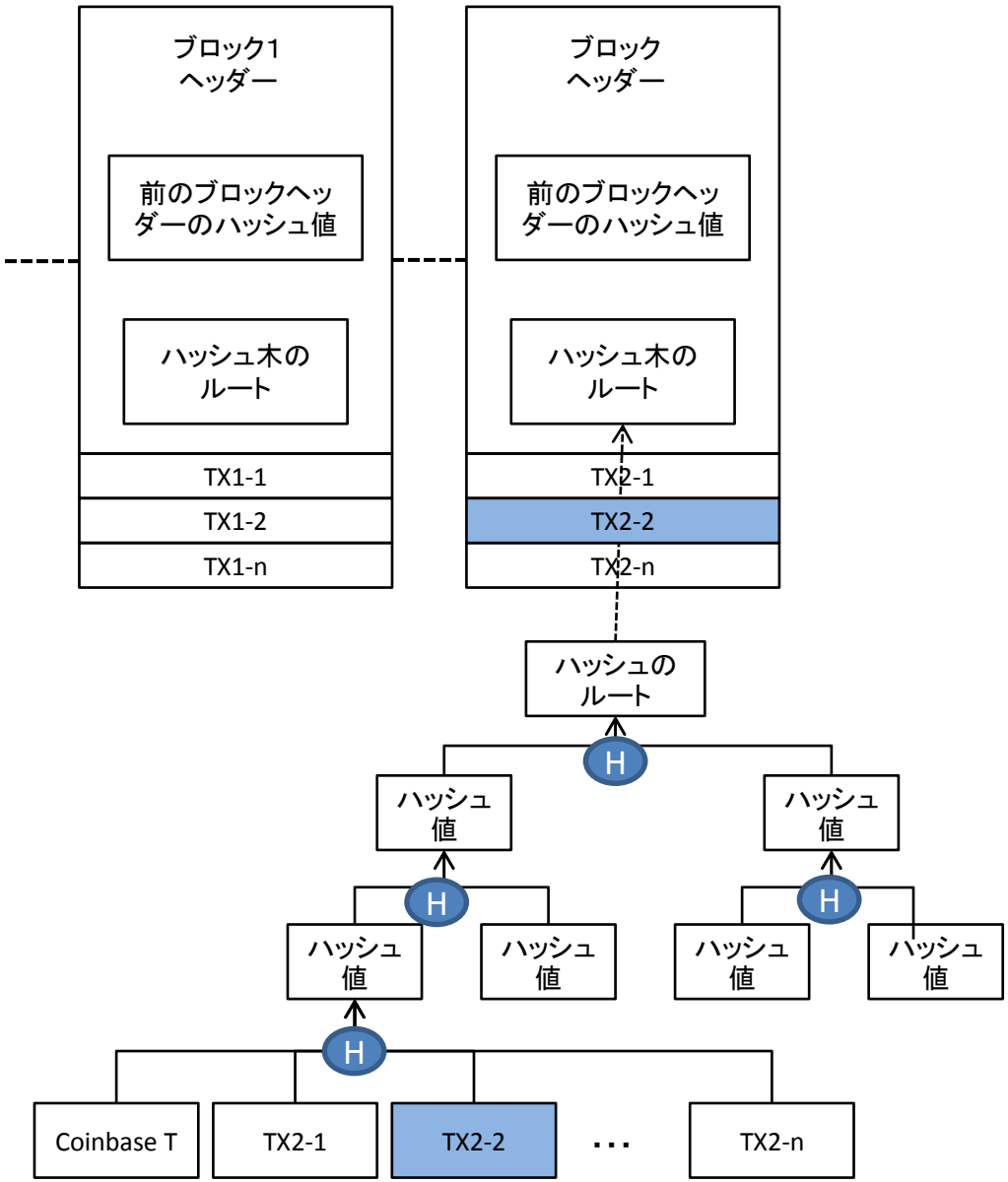
- ずっと追いつかない（はず）
- 正当なブロック作成に貢献した方が得

安全性にはレベルがある

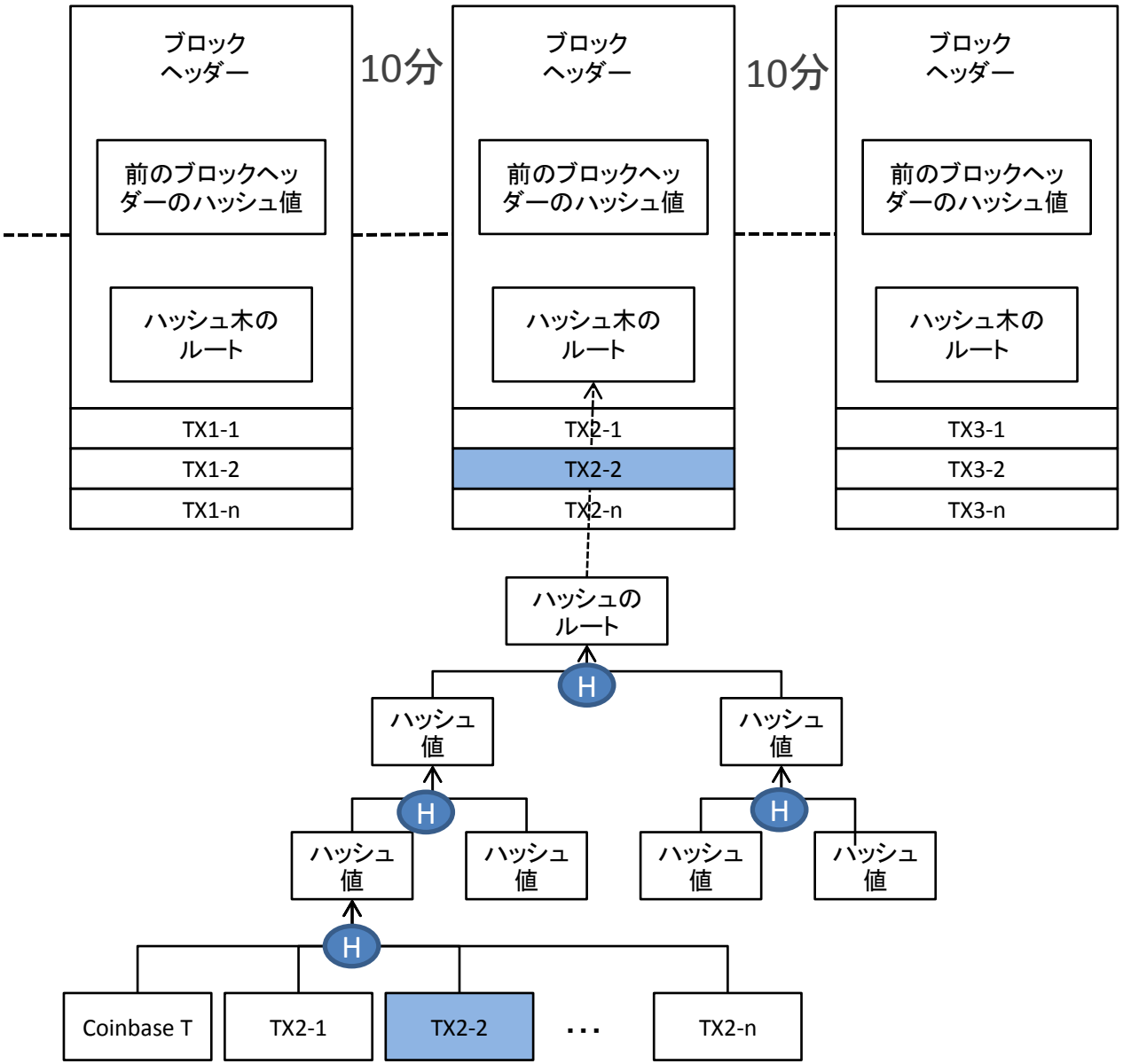
データ作成直後は認証ゼロ（誰も確認できていない）



認証レベル1＝10分

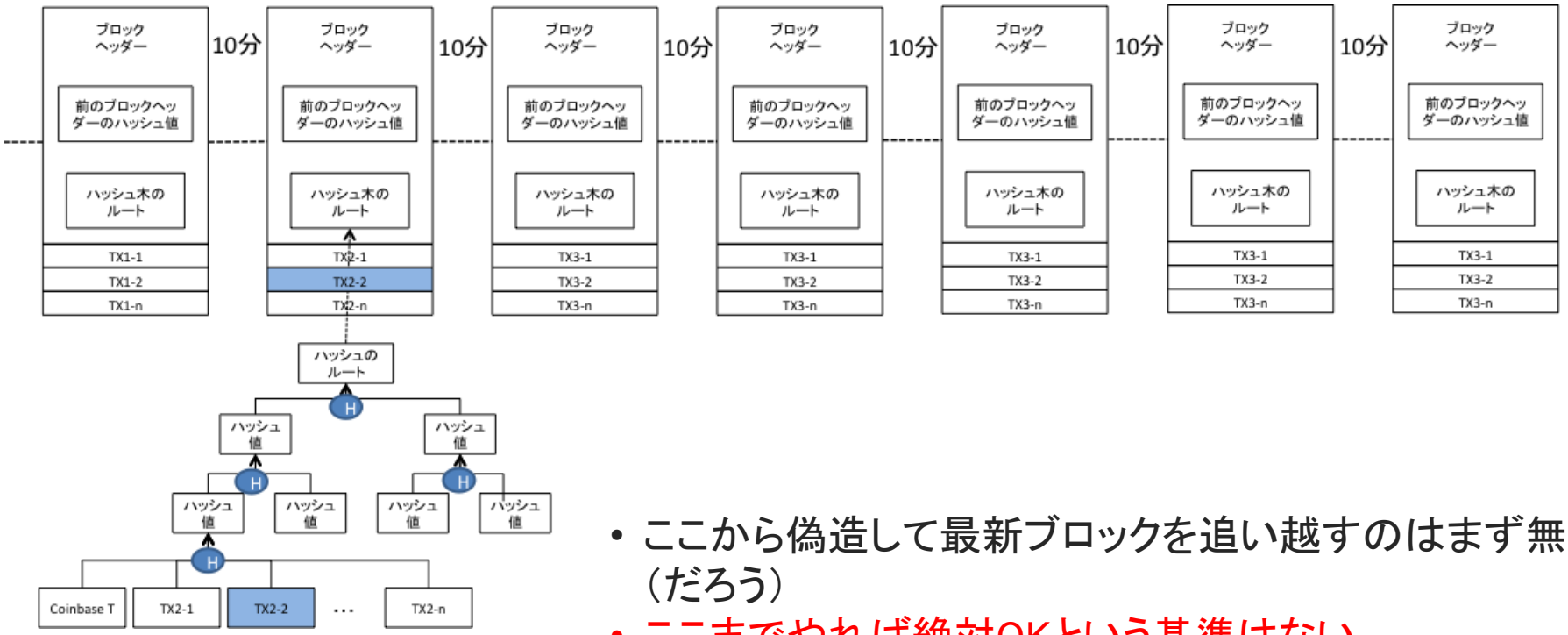

ブロック1
ヘッダー
前のブロックヘッダーのハッシュ値
ハッシュ木のルート
TX1-1
TX1-2
TX1-n
ブロック
ヘッダー
前のブロックヘッダーのハッシュ値
ハッシュ木のルート
TX2-1
TX2-2
TX2-n
ハッシュのルート
H
ハッシュ値
ハッシュ値
H
H
ハッシュ値
ハッシュ値
ハッシュ値
ハッシュ値
H
Coinbase T
TX2-1
TX2-2
・・・
TX2-n

認証レベル2＝20分

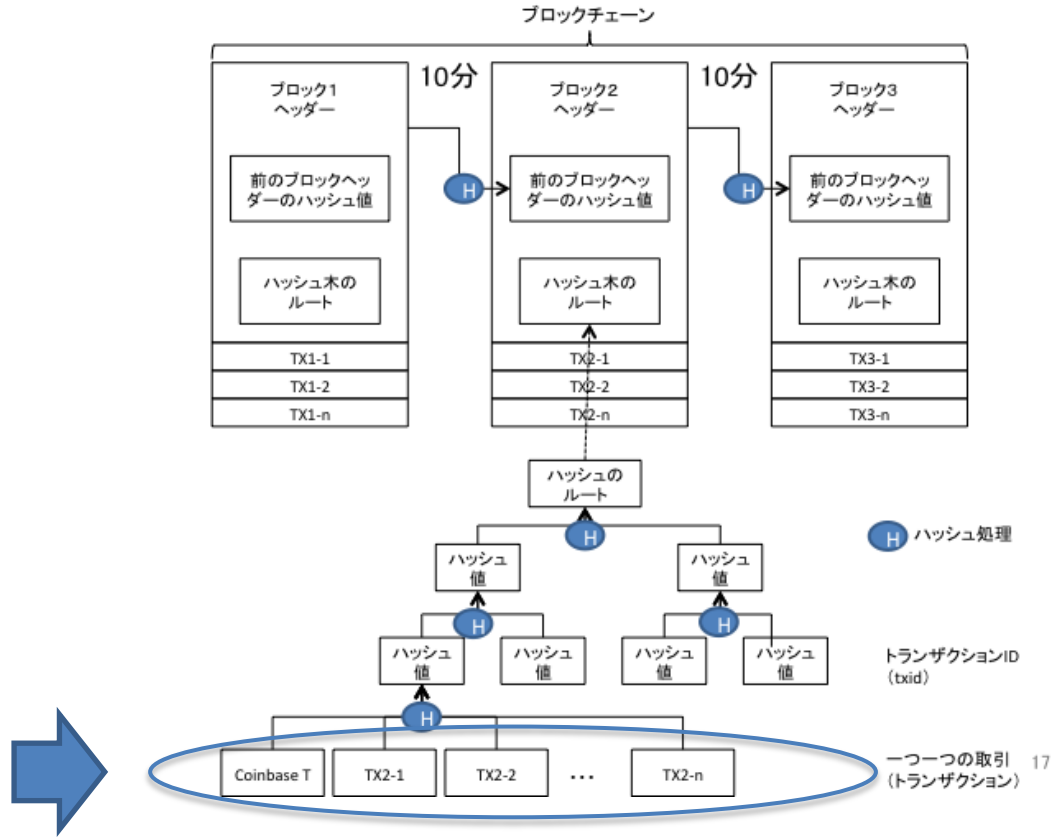

ブロック
ヘッダー
前のブロックヘッダーのハッシュ値
ハッシュ木のルート
10分
ブロック
ヘッダー
前のブロックヘッダーのハッシュ値
ハッシュ木のルート
10分
ブロック
ヘッダー
前のブロックヘッダーのハッシュ値
ハッシュ木のルート
TX1-1
TX1-2
TX1-n
TX2-1
TX2-2
TX2-n
TX3-1
TX3-2
TX3-n
ハッシュのルート
H
ハッシュ値
ハッシュ値
H
H
ハッシュ値
ハッシュ値
ハッシュ値
ハッシュ値
H
Coinbase T
TX2-1
TX2-2
・・・
TX2-n

認証レベル6＝1時間

認証レベル6＝1時間で、ほぼ安全（偽造不可能）だとみなされる。



- ここから偽造して最新ブロックを追い越すのはまず無理（だろう）
- ここまでやれば絶対OKという基準はない

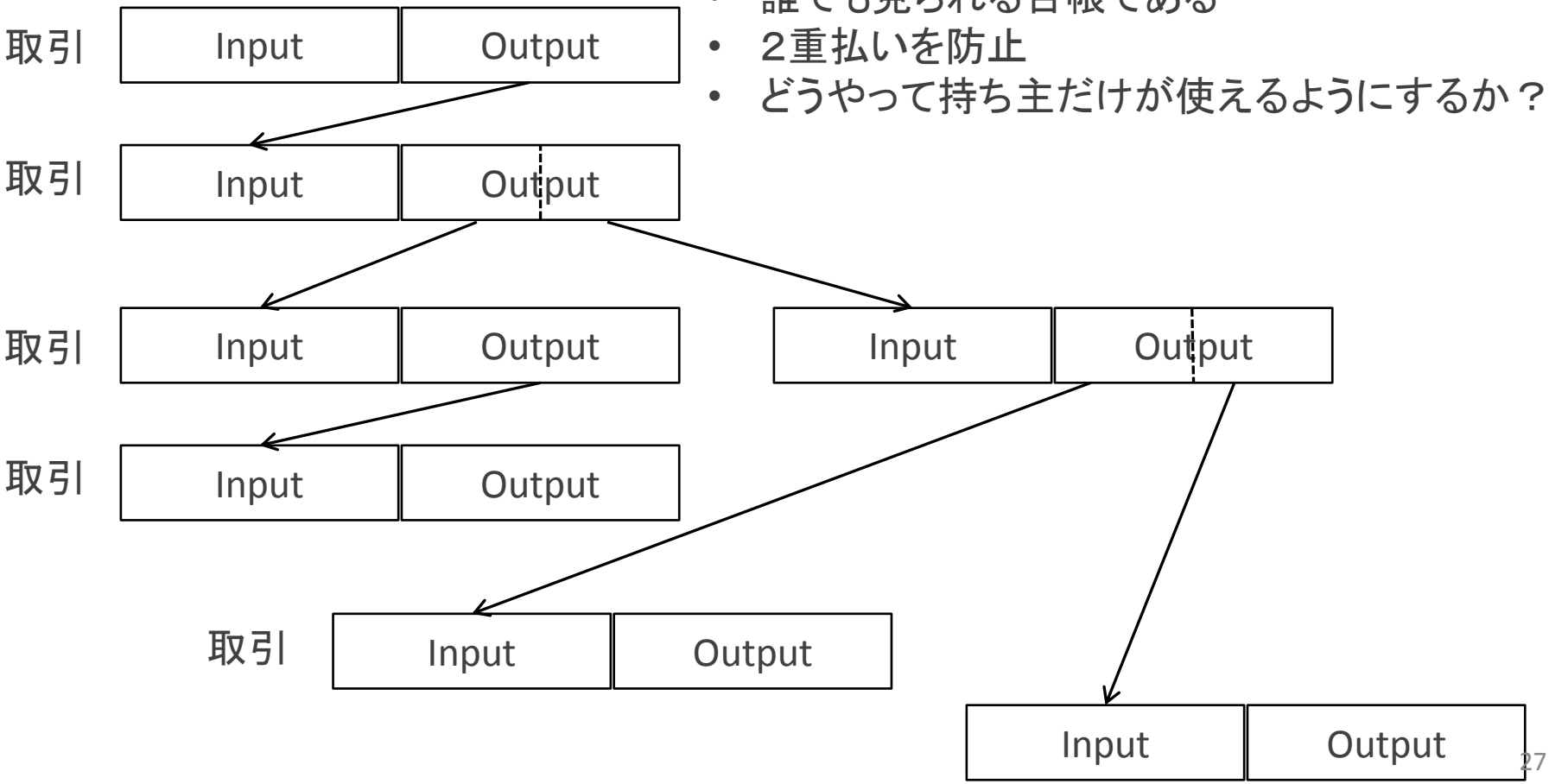

ブロックチェーン
ブロック1
ヘッダー
10分
ブロック2
ヘッダー
10分
ブロック3
ヘッダー
前のブロックヘッダーのハッシュ値
ハッシュ木のルート
TX1-1
TX1-2
TX1-n
TX2-1
TX2-2
TX2-n
TX3-1
TX3-2
TX3-n
H
ハッシュのルート
ハッシュ値
ハッシュ処理
トランザクションID
(txid)
Coinbase T
TX2-1
TX2-2
TX2-n
一つ一つの取引
(トランザクション)

取引データの詳細

個々の取引データを見ると・・・

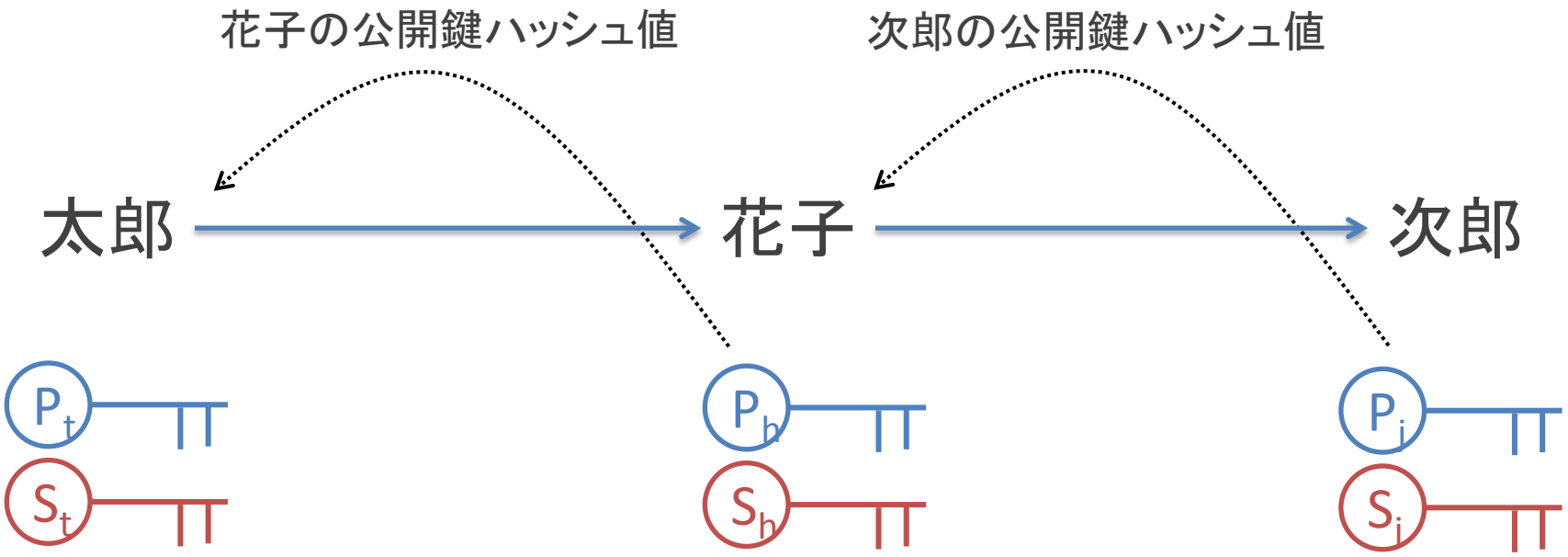
「受け取ったものを使う」ことで連結

- 誰でも見られる台帳である
- ２重払いを防止
- どうやって持ち主だけが使えるようにするか？

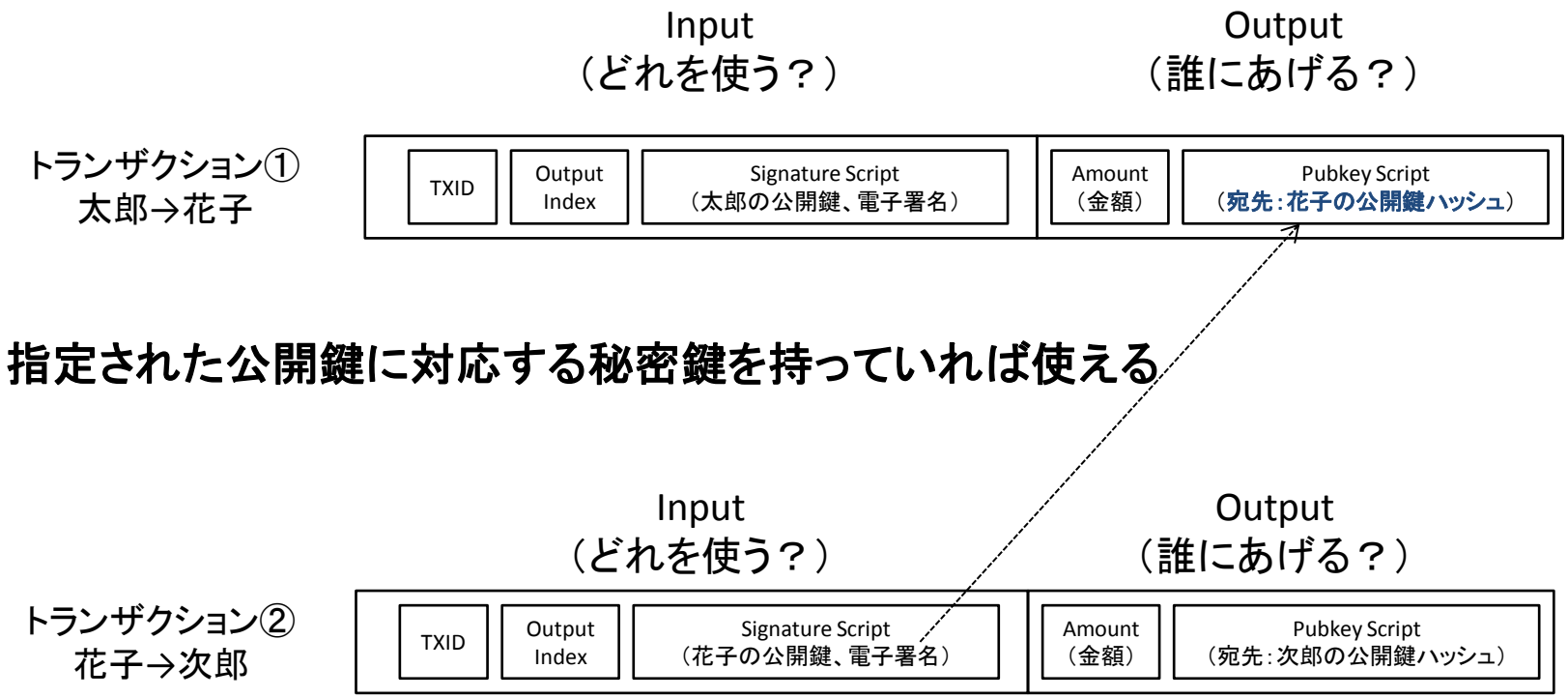


取引の方法（事前準備）

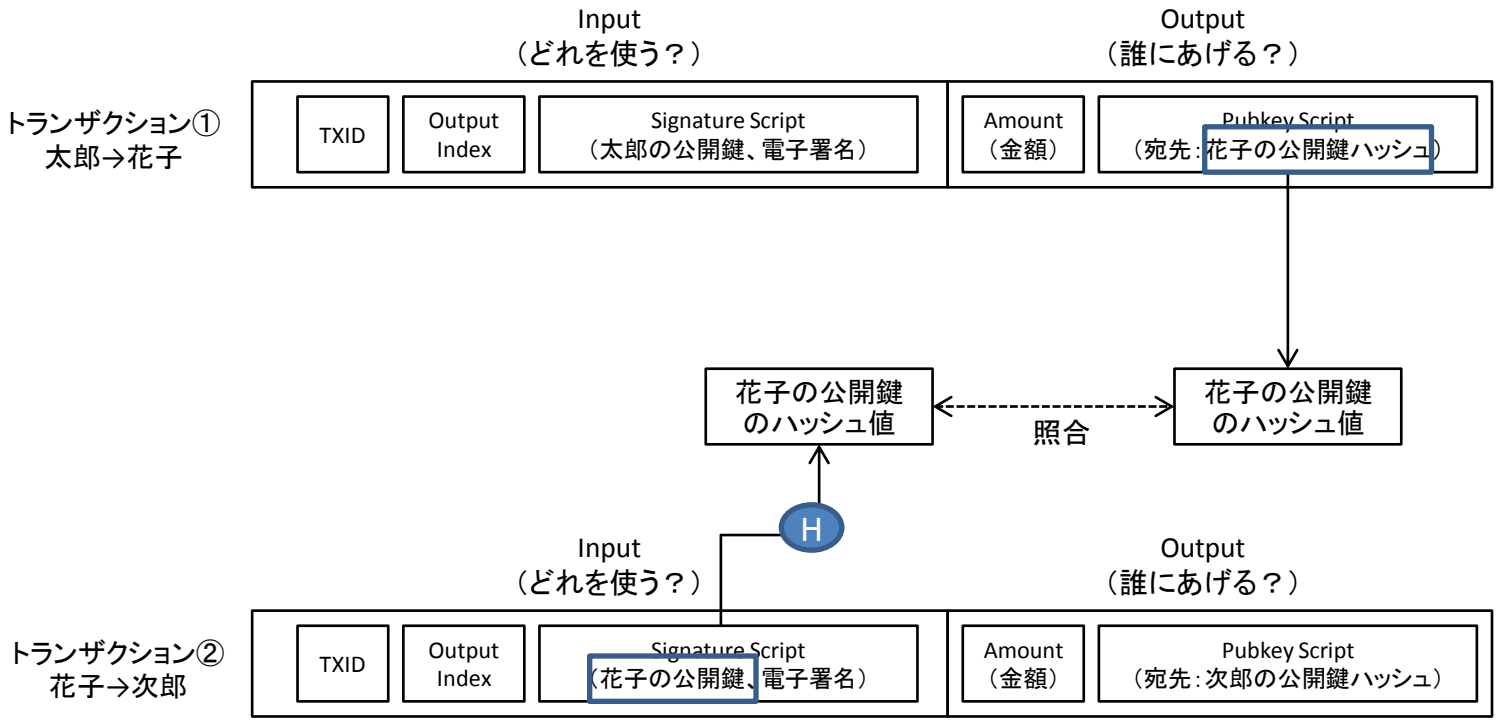
- 太郎が花子に支払い、そのお金を花子が次郎に支払う場面を想定
- 事前に受け手は送り手に**自分の公開鍵ハッシュ値（＝宛先）を教えて**おく。



取引データの中身

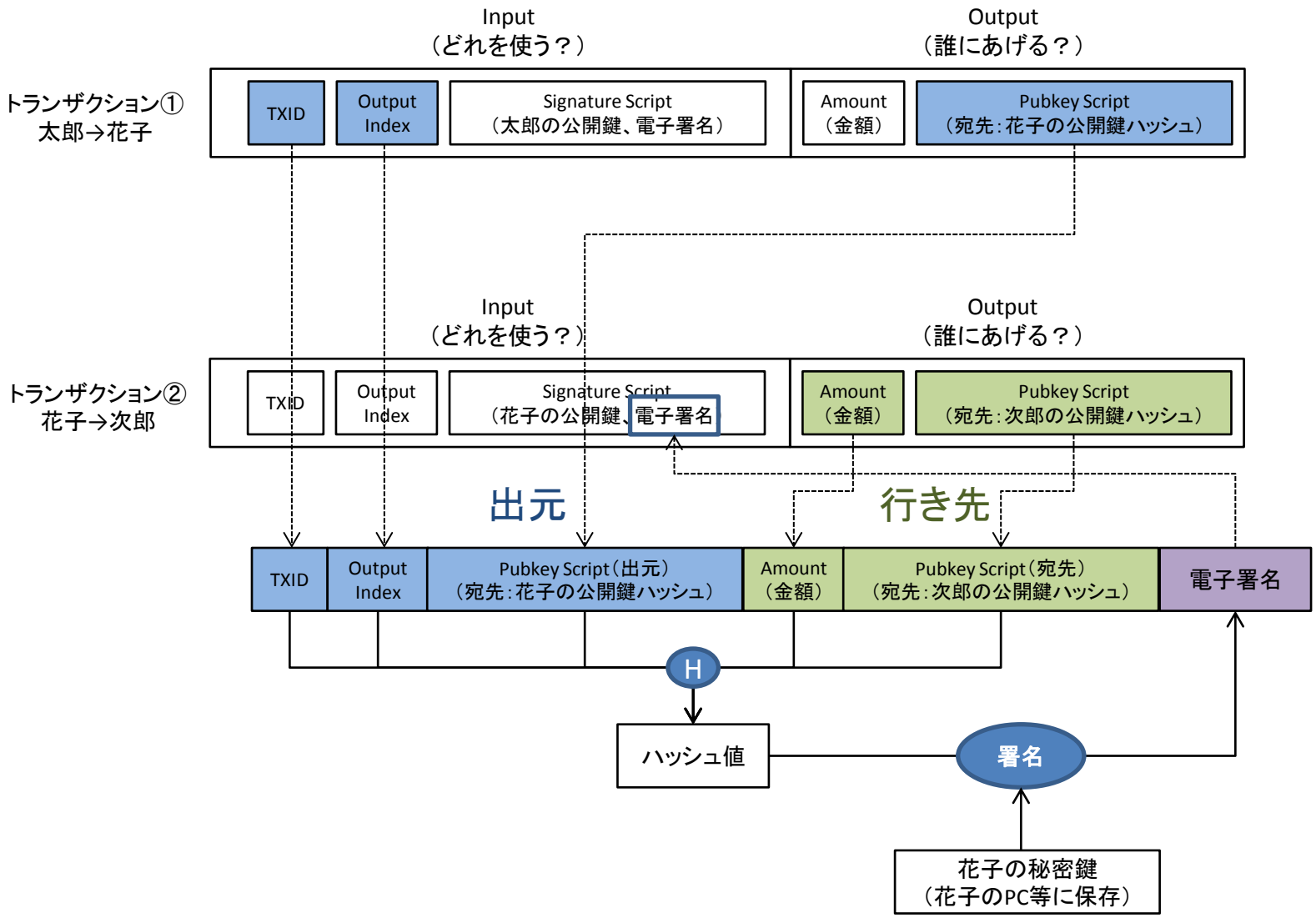

Input
（どれを使う？）
Output
（誰にあげる？）
トランザクション①
太郎→花子
TXID
Output Index
Signature Script
（太郎の公開鍵、電子署名）
Amount
（金額）
Pubkey Script
（宛先：花子の公開鍵ハッシュ）
指定された公開鍵に対応する秘密鍵を持っていれば使える
Input
（どれを使う？）
Output
（誰にあげる？）
トランザクション②
花子→次郎
TXID
Output Index
Signature Script
（花子の公開鍵、電子署名）
Amount
（金額）
Pubkey Script
（宛先：次郎の公開鍵ハッシュ）

権利の確認①

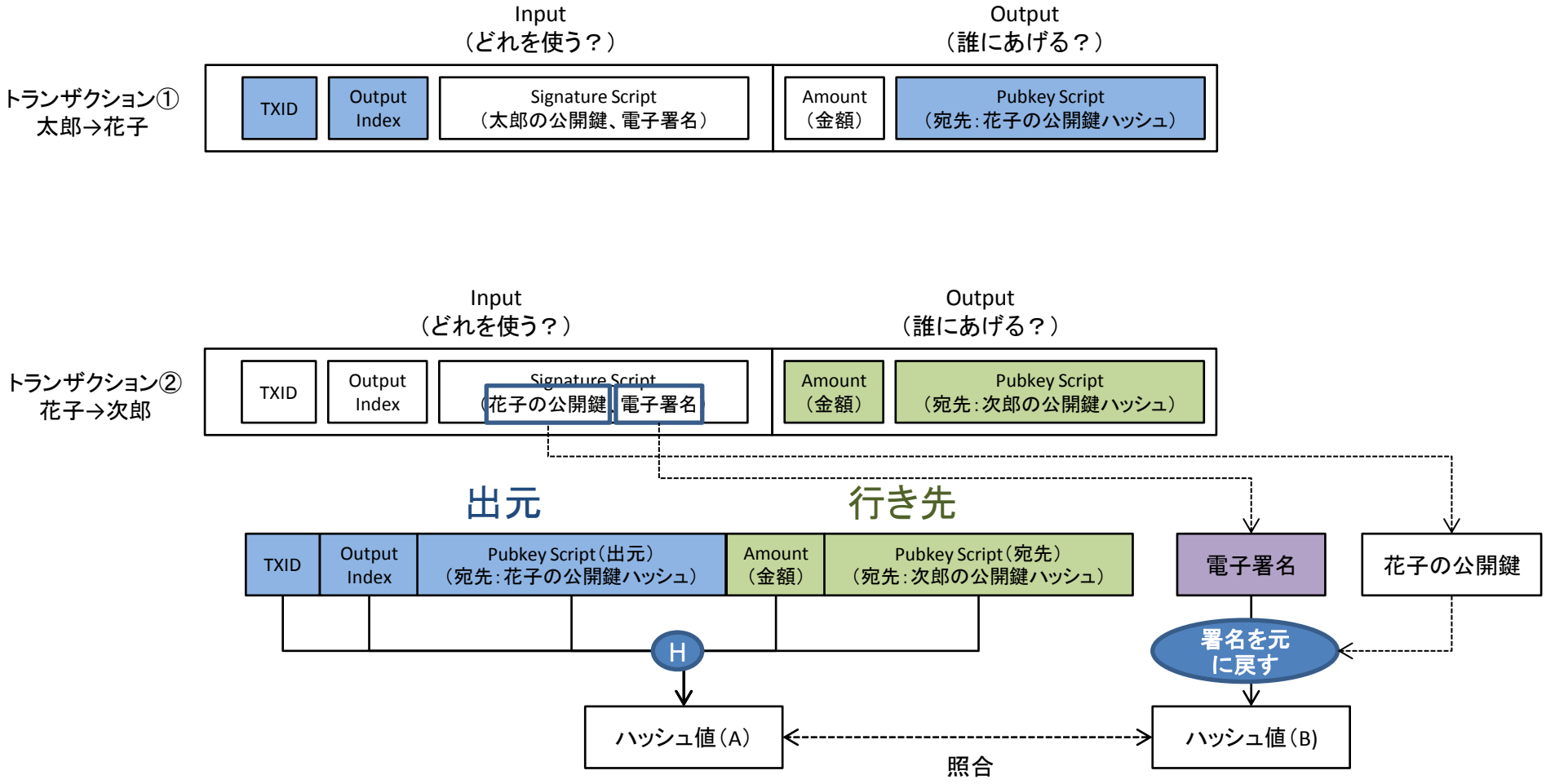


権利があるか＝公開鍵に対応する秘密鍵を持っているか？

権利の確認②


Input
（どれを使う？）
Output
（誰にあげる？）
トランザクション①
太郎→花子
TXID
Output Index
Signature Script
（太郎の公開鍵、電子署名）
Amount
（金額）
Pubkey Script
（宛先：花子の公開鍵ハッシュ）
Input
（どれを使う？）
Output
（誰にあげる？）
トランザクション②
花子→次郎
TXID
Output Index
Signature Script
（花子の公開鍵、電子署名）
Amount
（金額）
Pubkey Script
（宛先：次郎の公開鍵ハッシュ）
出元
行き先
TXID
Output Index
Pubkey Script（出元）
（宛先：花子の公開鍵ハッシュ）
Amount
（金額）
Pubkey Script（宛先）
（宛先：次郎の公開鍵ハッシュ）
電子署名
H
ハッシュ値
署名
花子の秘密鍵
（花子のPC等に保存）

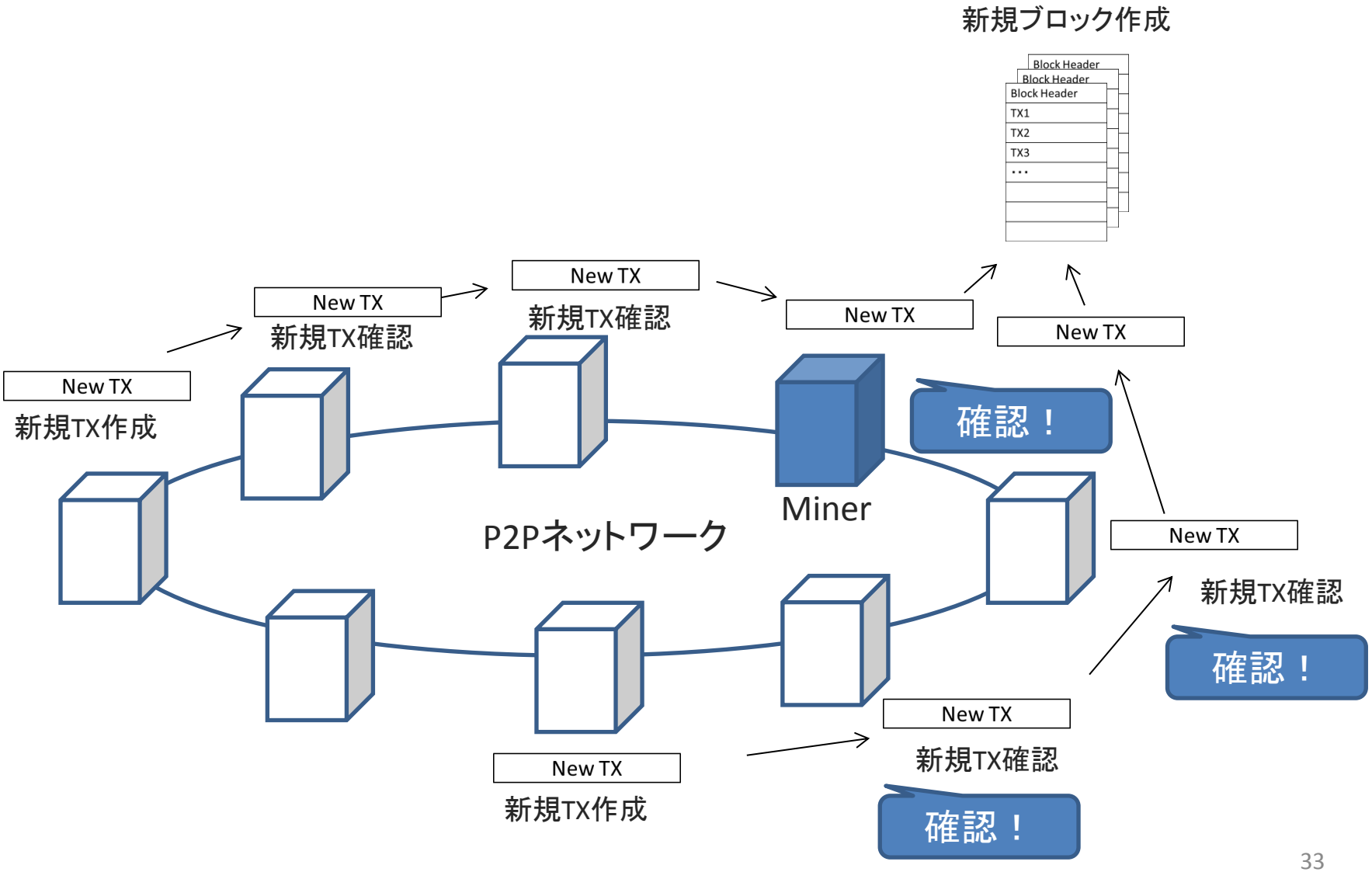
権利の確認③



一致する＝花子の公開鍵に対応する秘密鍵を花子が確かに持っている

誰が確認するのか？

P2Pネットワークにおけるデータの流れ

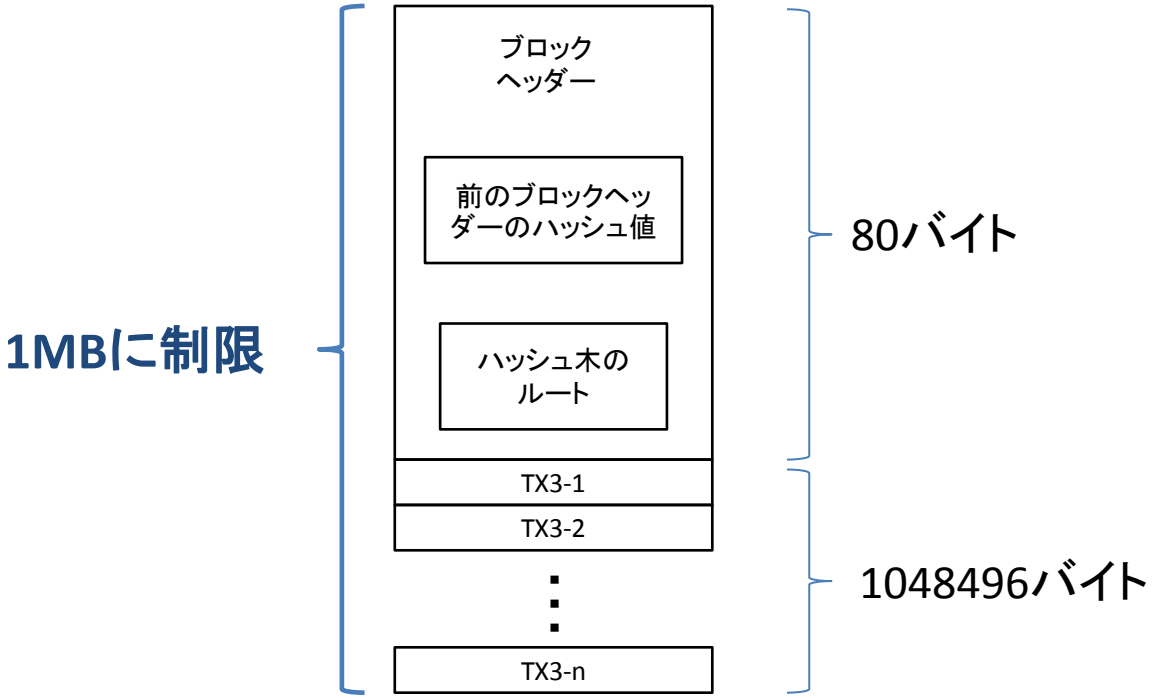


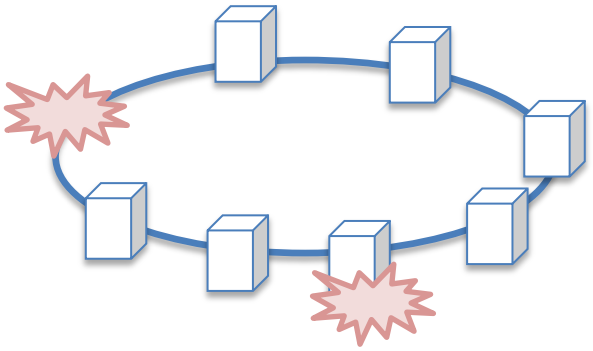
この方式で何ができるのか？

- 受け取り手に指定されている人だけが（対応する秘密鍵を持っていれば）そのリソースを使うことができる
- 出元と行き先を証明→送り手は2重にリソースを使うことができない
- 誰が誰に何を渡したのか証明されている（否認できない）

ところで！

ブロックサイズ問題





1048576バイト− 80バイト＝1048496バイト

1048496÷平均250バイト＝4194取引

4194÷10分÷60秒≒**7取引／秒**

< **2000取引／秒（VISA）**
115取引／秒（Paypal）

マイニングをめぐる経済性

- P2Pでのシステム維持←マイニングで実現
- 1ブロックあたり25BTCをマイナーに新規発行
  - 4年ごとに半減、最終的に2100万BTCで発行終了

- 設備投資、電気代というコストを払って新規発行コインを得る

- 実は新規発行とは、他人の持ち分の希薄化
  - 通常通貨の場合
    - 追加発行＝量的緩和＝貨幣の価値の下落＝インフレ
    - 株式の増資も同じ
  - 本来であればビットコインは減価されていくはず
  - なぜ上がっているのか？
    - 投機的期待（例：株価）
    - 増加分の打ち止め予定→乱発の回避が期待される
    - 貨幣としての需要増加期待

- 他人の持ち分希薄化でマイナーの労力が賄われている
  - しかしその外部貨幣に対する価値は、貨幣需要への期待に依存

基本構造から浮かび上がる特徴と課題

○ **主体とリソースの関係**を論理的にリンク可能
　　⇒情報資産の流通管理に幅広い応用
○ **分散型でデータを管理**
　　⇒単一ポイントの脆弱性は無い
○ P2Pでシステムを維持する**インセンティブ**を埋め込み
　　⇒組織不要・管理者不要・上司不要での業務遂行

どこに魅力を感じて使っていくか？

△ **認証に時間がかかる**
　　⇒Proof of Work（約10分）の問題
△ **セキュリティ**
　　⇒情報の秘匿性
　　⇒改ざんは検知する努力が必要
　　⇒秘密鍵の管理
　　⇒公開鍵特定によるプライバシーの問題
△ 情報システムアーキテクチャとして
　　⇒過去のデータは修正不可能→**データは単調増加**
　　⇒データの検索性
　　⇒**スケーラビリティの問題**
△ ビットコイン以外に応用した際の**マイニングのインセンティブ**

これらを解決するための工夫が百花繚乱中

ブロックチェーン2.0について

認証時間の短縮、対象の拡張、汎用化、秘匿化等へ

ブロックチェーン1.0

- ビットコインを実現
- オープンネットワーク
- Proof of Work

ここまでの話

ブロックチェーン1.5

- ビットコインの**フォーク**
  - パラメータの調整、部品の入れ替え
    - Litecoin
    - DASH
    - Peercoin
  - 載せるものの変更（ネームコイン）
- ビットコインへの**相乗り**
  - ビットコインのブロックチェーン上に独自発行アセットを載せる

ブロックチェーン2.0

- 通貨から**汎用化**
  - 契約
  - 自動実行
  - プログラム環境
- **許可型**BC

2.0へ：応用可能性の拡大

- 様々なプロパティ（資産）や取引をブロックチェーン上で登録、確認、移転できる可能性

ブロックチェーンで管理できる可能性があるプロパティの例

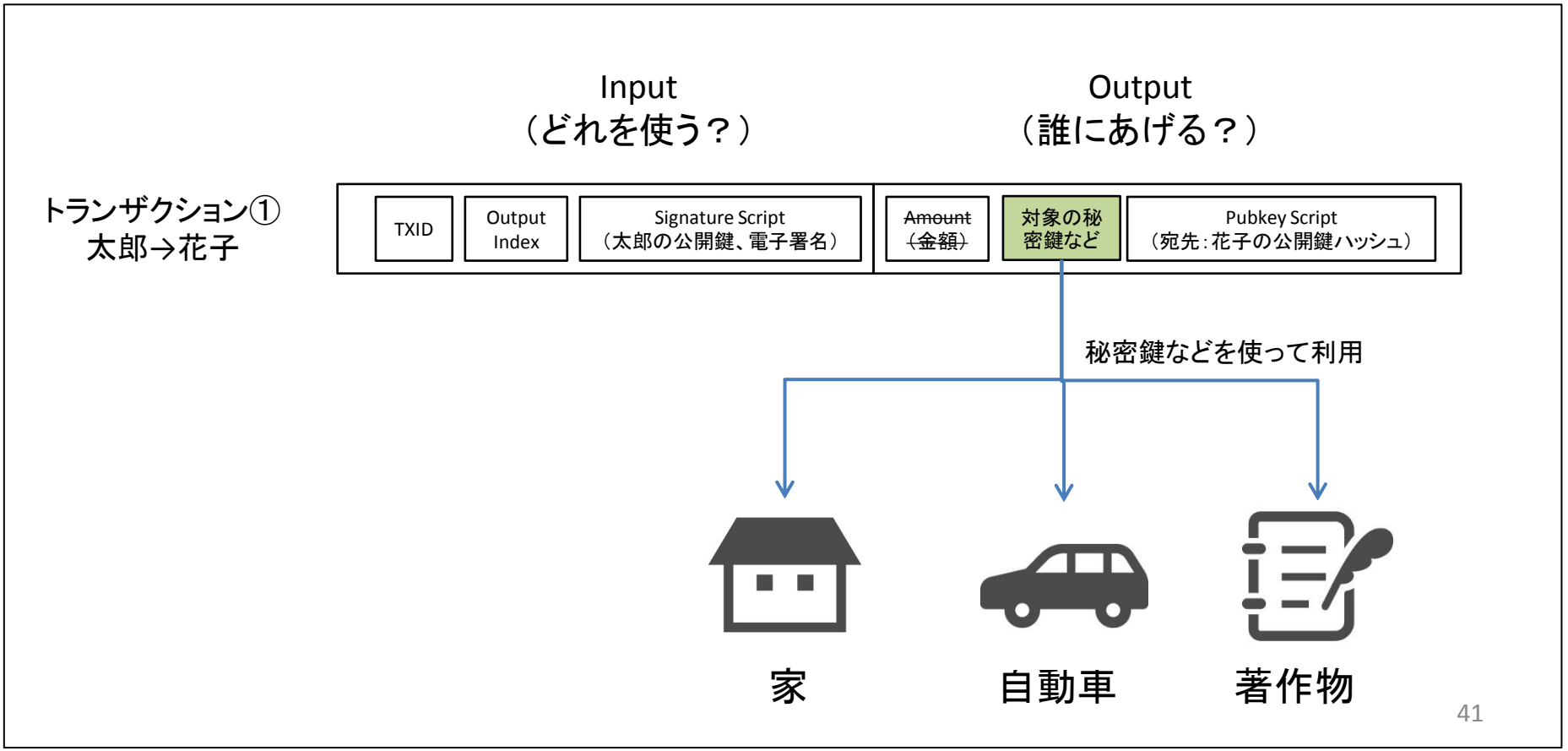
種別	例
一般	エスクロー取引、担保付取引、第三者裁定、複数者取引
金融取引	**株、未公開株**、クラウドファンディング、債券、投資信託、デリバティブ、年金保険、年金
公的情報	**不動産登記、自動車登録、事業者登録**、結婚証明、死亡証明
ID	運転免許、IDカード、パスポート、有権者登録
民間	借用証書、ローン、契約、賭け、署名、遺言、信託、エスクロー
各種証明	**保険証明**、所有証明、公証
有形資産の鍵	家、ホテルの部屋、**レンタカー**、自動車へのアクセス
無形資産の鍵	**特許、商標、著作権**、予約、ドメイン名

出典：Swan, 2015

スマート・プロパティ

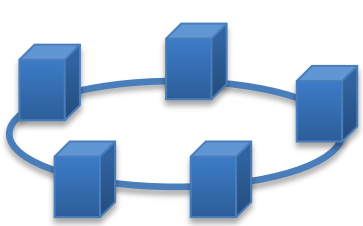
- 現実の資産と紐付くデジタルのシンボルをブロックチェーン上で管理
- どうやって資産とデジタルシンボルを紐づけるか？

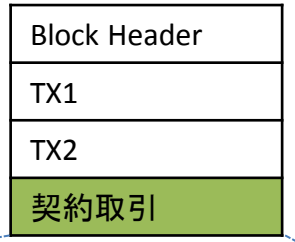
概念図（例）



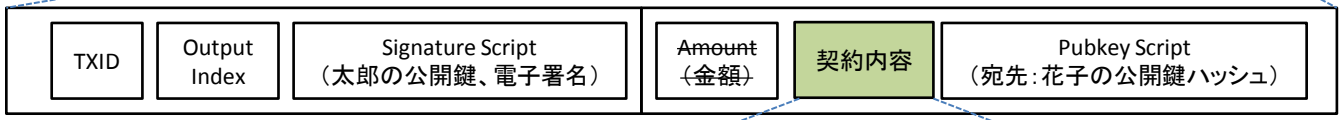
スマート・コントラクト

- ブロックチェーンの取引内に**契約内容を記述したプログラム**を格納





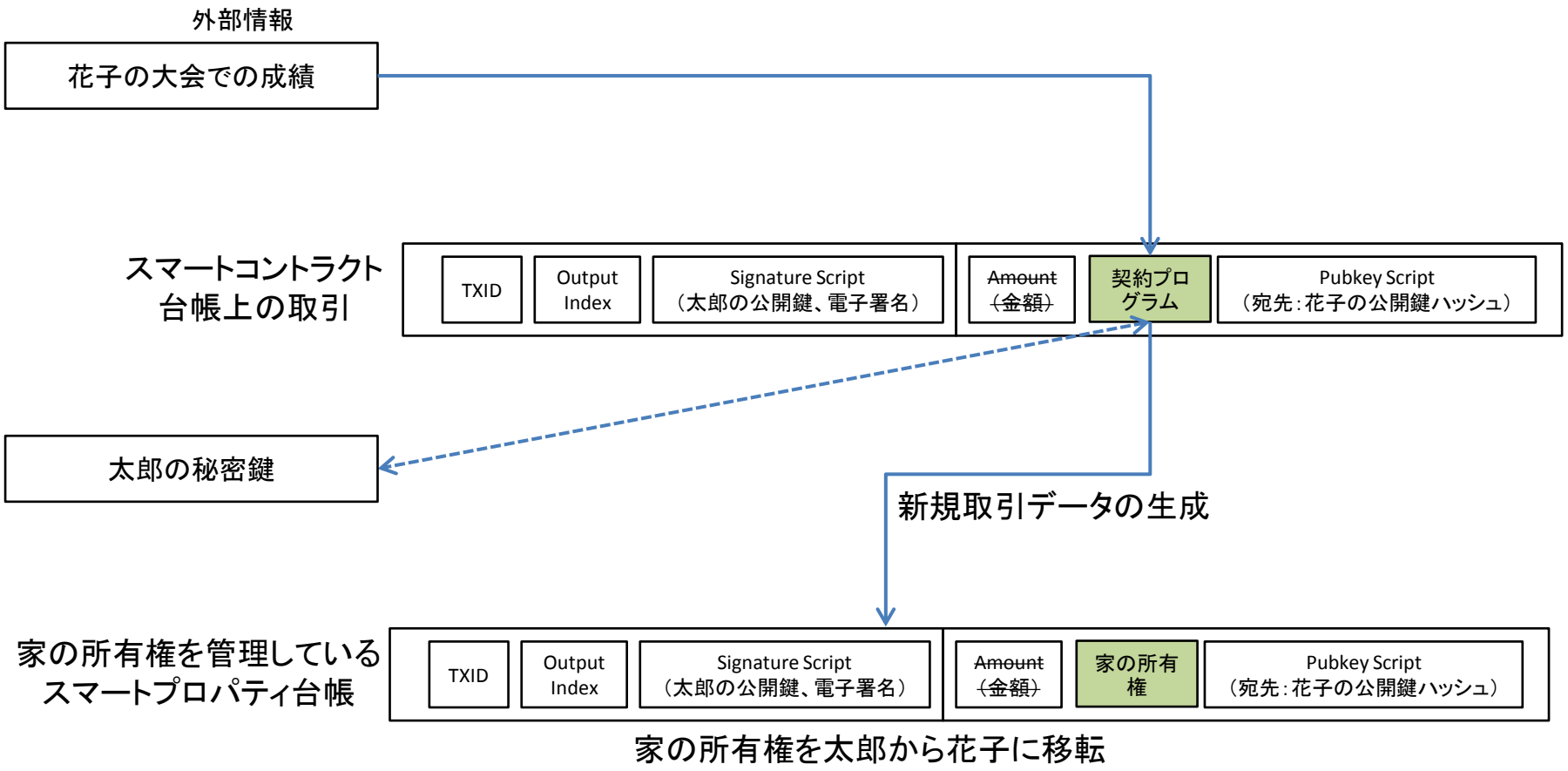
スマートコントラクト
台帳上の取引



契約内容（例）
IF
もし花子が大会で1等賞を取ったら

THEN
太郎は花子に家をプレゼントする

スマート・コントラクトの動作（例）



- 契約プログラムは第三者のエージェントとして契約の実行を自動管理
- 外部情報に依存（大会結果、第三者の秘密鍵）

Ethereum（イーサリアム）

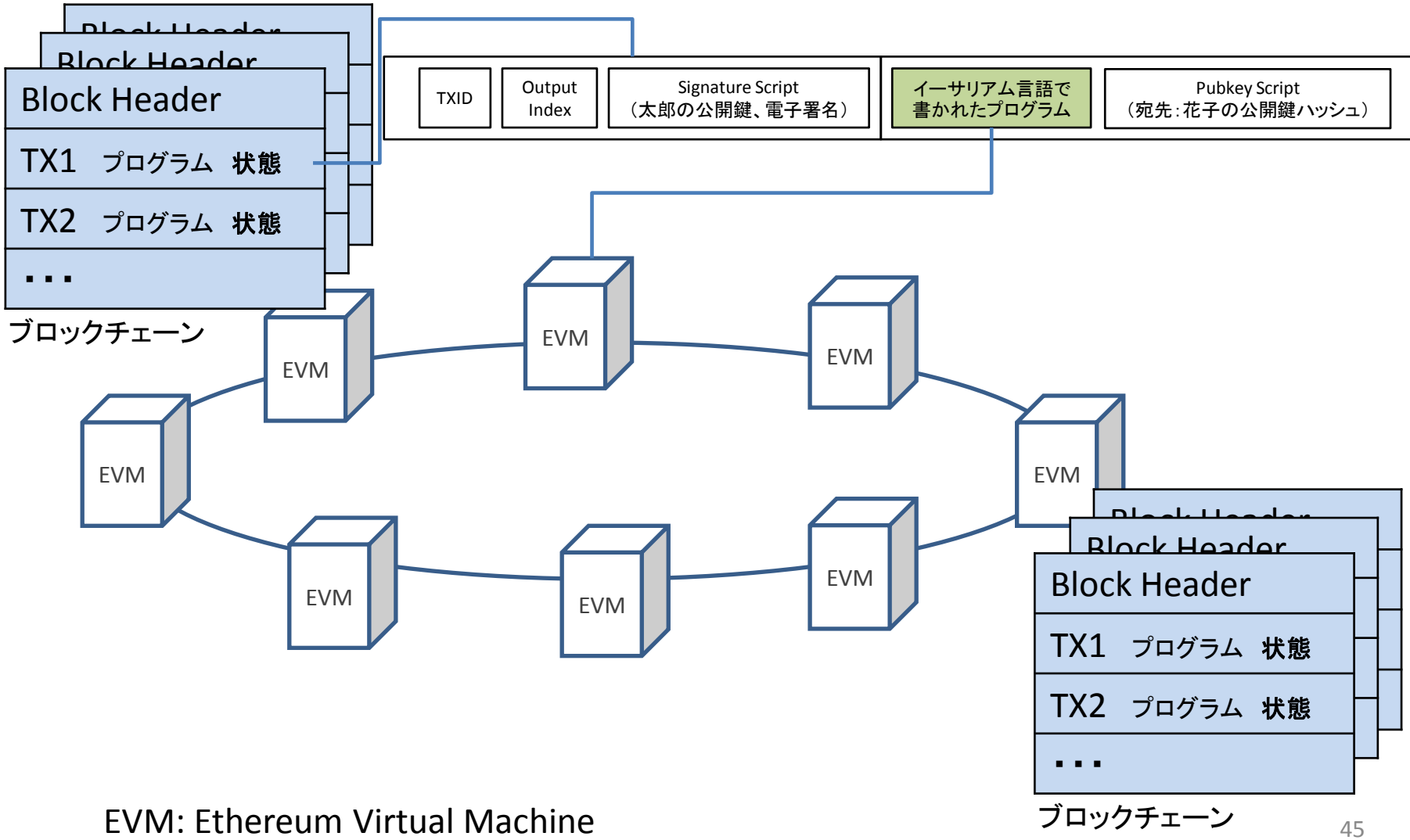
- 特定の仕様や環境に制限を受けず、**もっと万能にブロックチェーン上でプログラムを実行したい**
  - チューリング完全性
  - Solidity（JAVAに似た言語）

- **分散アプリケーションを構築・配布するためのプラットフォーム**であり、プログラム言語である。（Swan）

- クラウドファンディングを原資にEthereum Foundation（スイス）にて開発され、サービスは2015年7月に開始（創生ブロック作成）

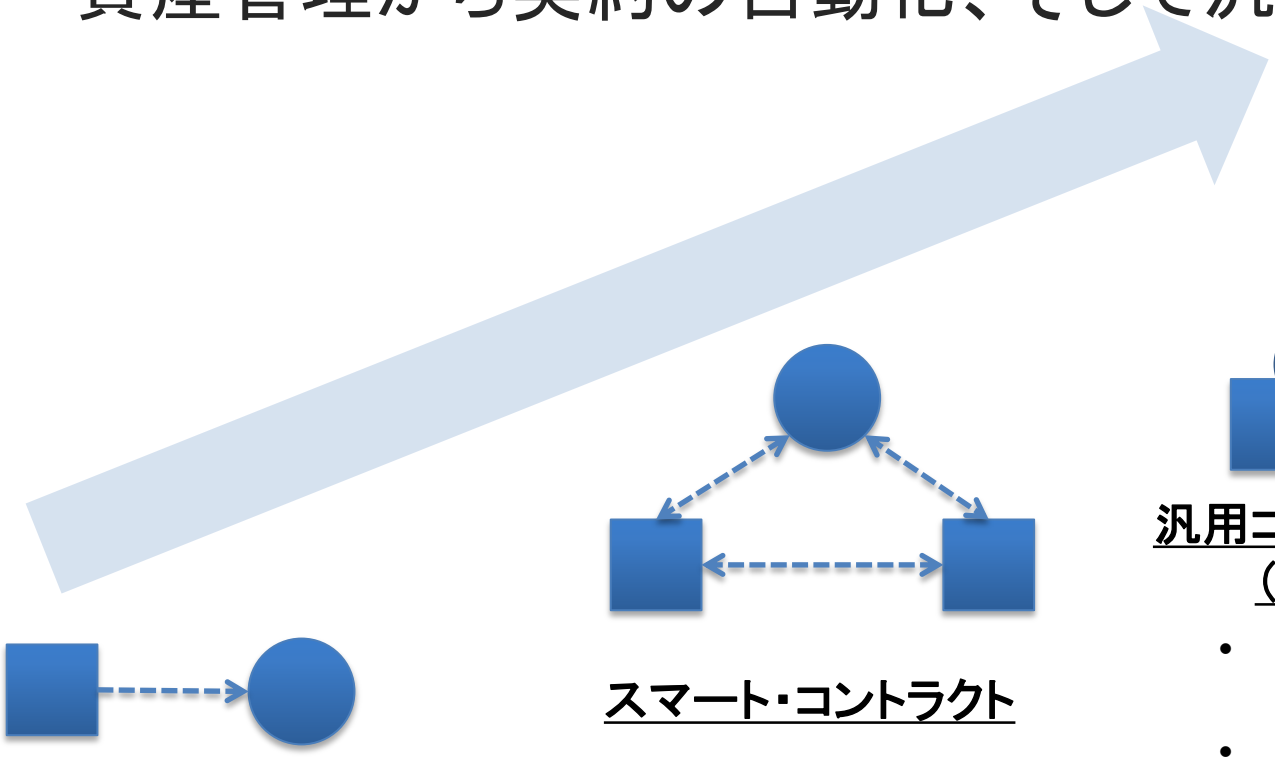
参考：WikipediaのEthernet→名称の「イーサ、ether」は、古典物理の時代に光の媒質として宇宙の隅々まで満たしているのではないかと考えられた仮想の物質、「エーテル」(Ether、Aether) から付けられた

分散コンピューティング環境
⇒ブロックチェーンは**状態遷移**の登録台帳



EVM: Ethereum Virtual Machine

資産管理から契約の自動化、そして汎用化へ



スマート・プロパティ

- エンティティとモノの関係を定義

スマート・コントラクト

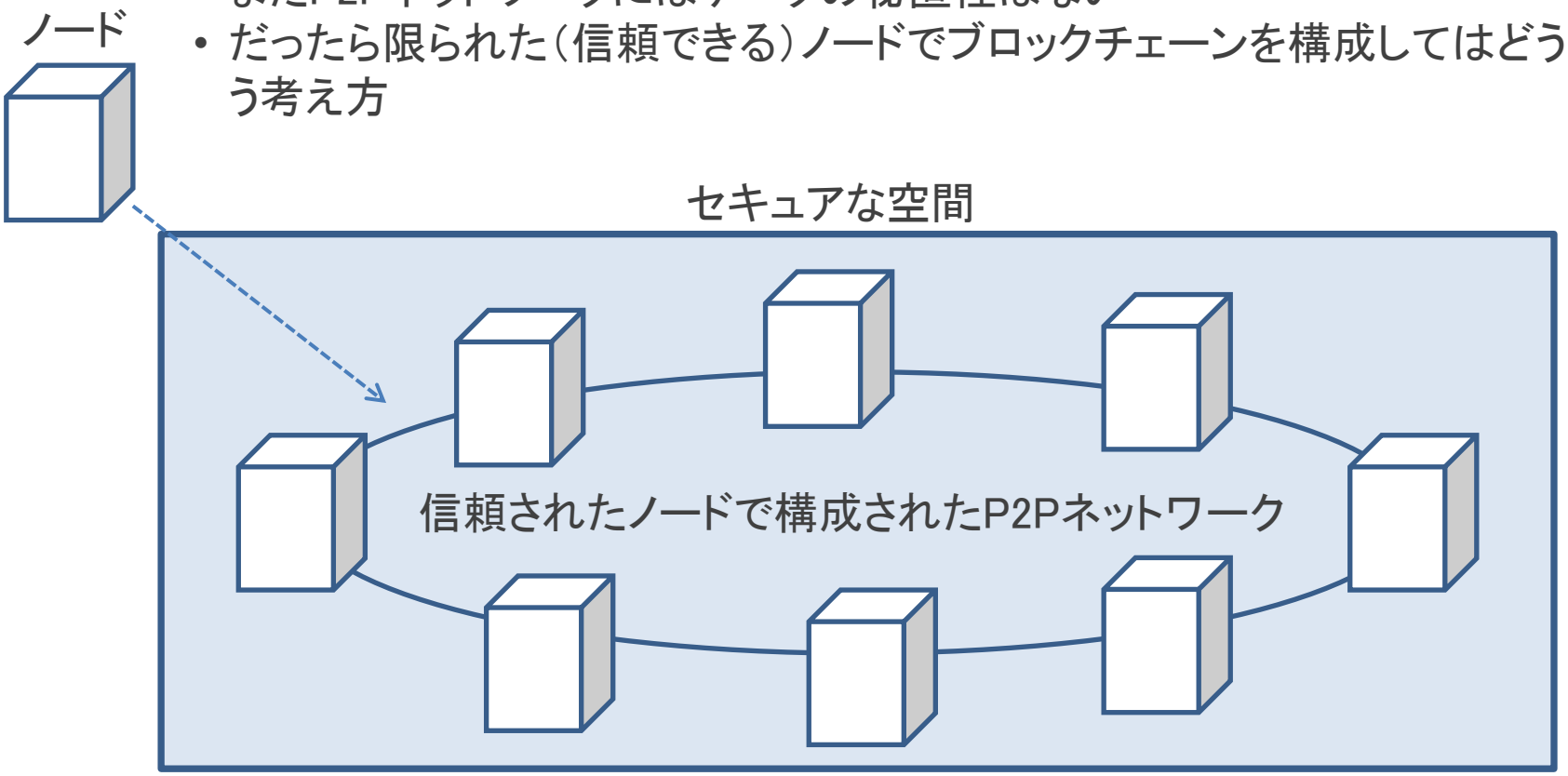
- エンティティとモノの関係を定義
- エンティティ同士の関係・アクションを定義
- 自動的に実行可能

汎用コンピューティング環境（イーサリアム等）

- エンティティとモノの関係を定義
- エンティティ同士の関係・アクションを定義
- アクションをブロックチェーンに内部化可能
- 定義できるアクションを汎用・万能化
- 自動的に実行

許可制(Permissioned)ブロックチェーンの登場

- Proof of workによる認証時間の問題などは、ノードが信頼できないことによる
- またP2Pネットワークにはデータの秘匿性はない
- だったら限られた(信頼できる)ノードでブロックチェーンを構成してはどうかという考え方



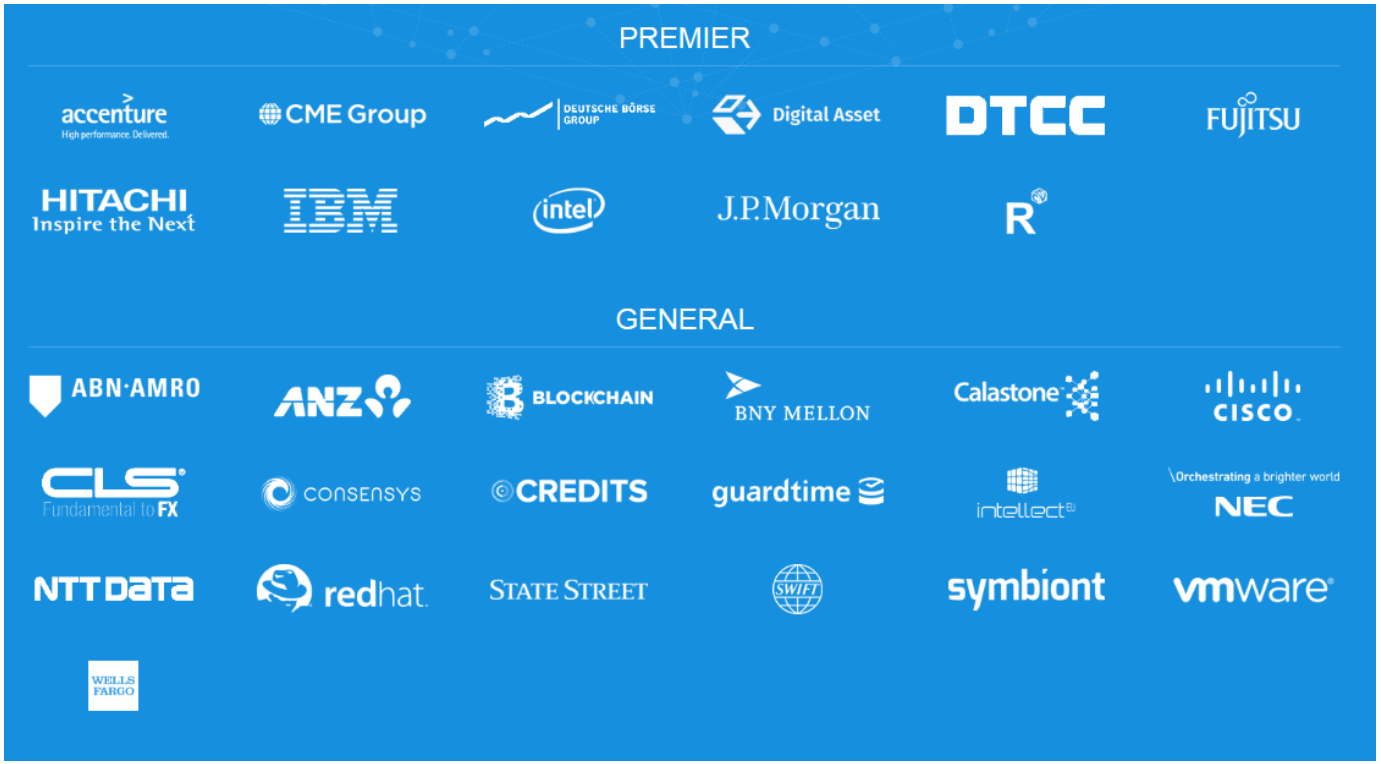
- オープン性に起因した問題解決手法(PoWなど)が不要
- 情報の秘匿性に配慮

許可制ブロックチェーンの登場

			認証（マイニング）の担い手	
			誰でも参加可能	限られた人だけ
			Unpermissioned	Permissioned
台帳を見られる人	誰でも見られる	Publicly Shared	Bitcoin **Ethereum**	Ripple
	限られた人だけ	Privately Shared		Bankchain **Hyperledger** **R3 Corda**

Hyperledger

- ブロックチェーンのオープンソースプロジェクト
- Linux Foundationが事務局
- Blockstream, Digital Asset, IBM, Rippleなどから技術提供
- 汎用性のある分散型台帳の標準プロトコルとコードを提供か



https://www.hyperledger.org/

驚くべき点

- データの秘匿性確保、但しコンプライアンス対応
  - Permissioned, Sharedを前提
  - ノード（参加者）への認証制度を導入（Registration Authority）
  - 取引内容を暗号化し、当事者しか見られない仕組みを導入
  - 但し、当局から監査の際は認証機能がアクセス権を付与

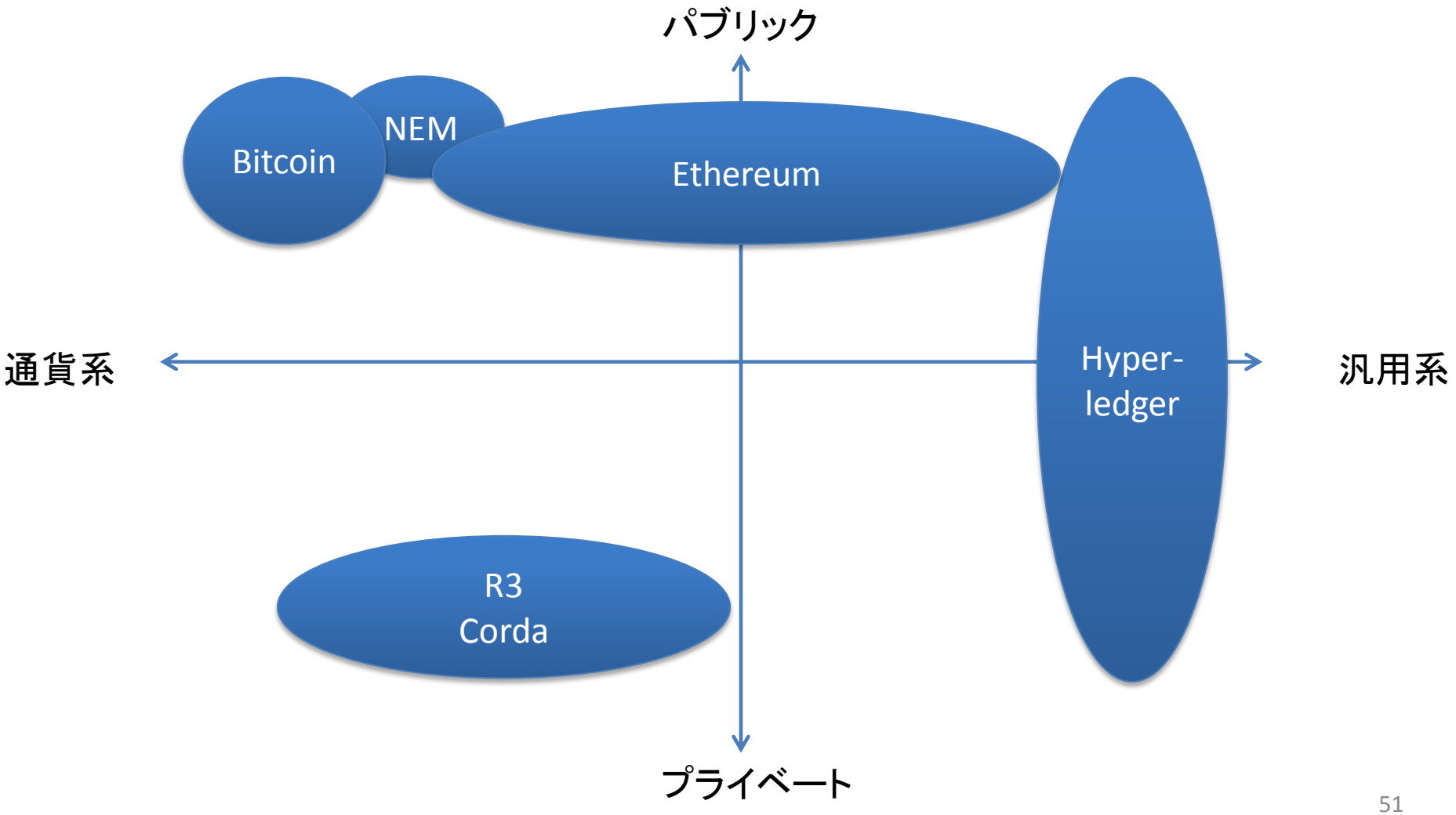
- スケーラビリティの向上
  - 15ノードで10万取引／秒

- 汎用性のあるプログラム動作環境
  - Chaincode
  - スマートコントラクト、Ehtereumに似た概念だがさらに汎用性を高めた
  - プログラム言語を問わず実行可能（なプラットフォームを提供）

- チェーン間相互運用性を考慮
  - World of many chains

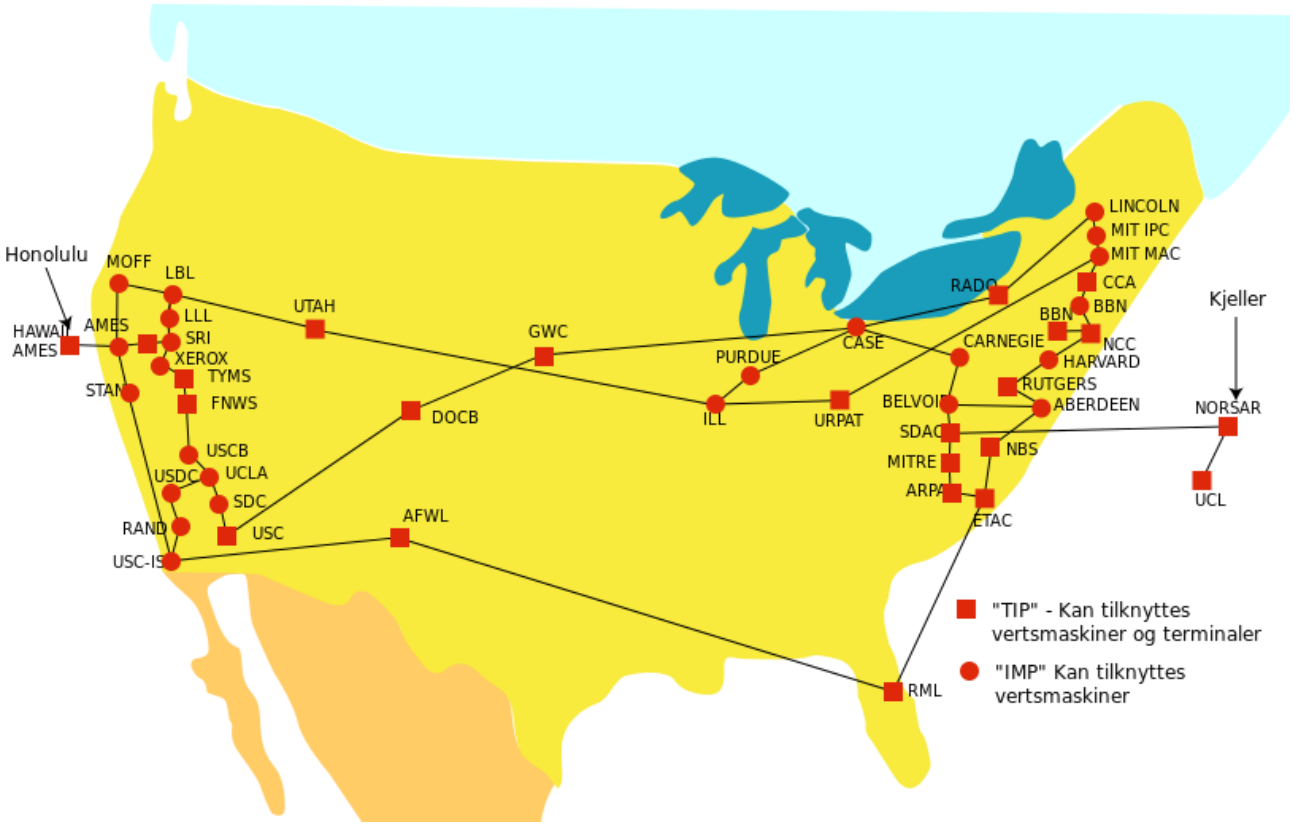
出典：Hyperledger White Paper
https://docs.google.com/document/d/1Z4M_qwILLRehPbVRUsJ3OF8Iir-gqS-ZYe7W-LE9gnE/edit?pref=2&pli=1#heading=h.m6iml6hqrnm2

ブロックチェーン・プロバイダーのビジネスモデル



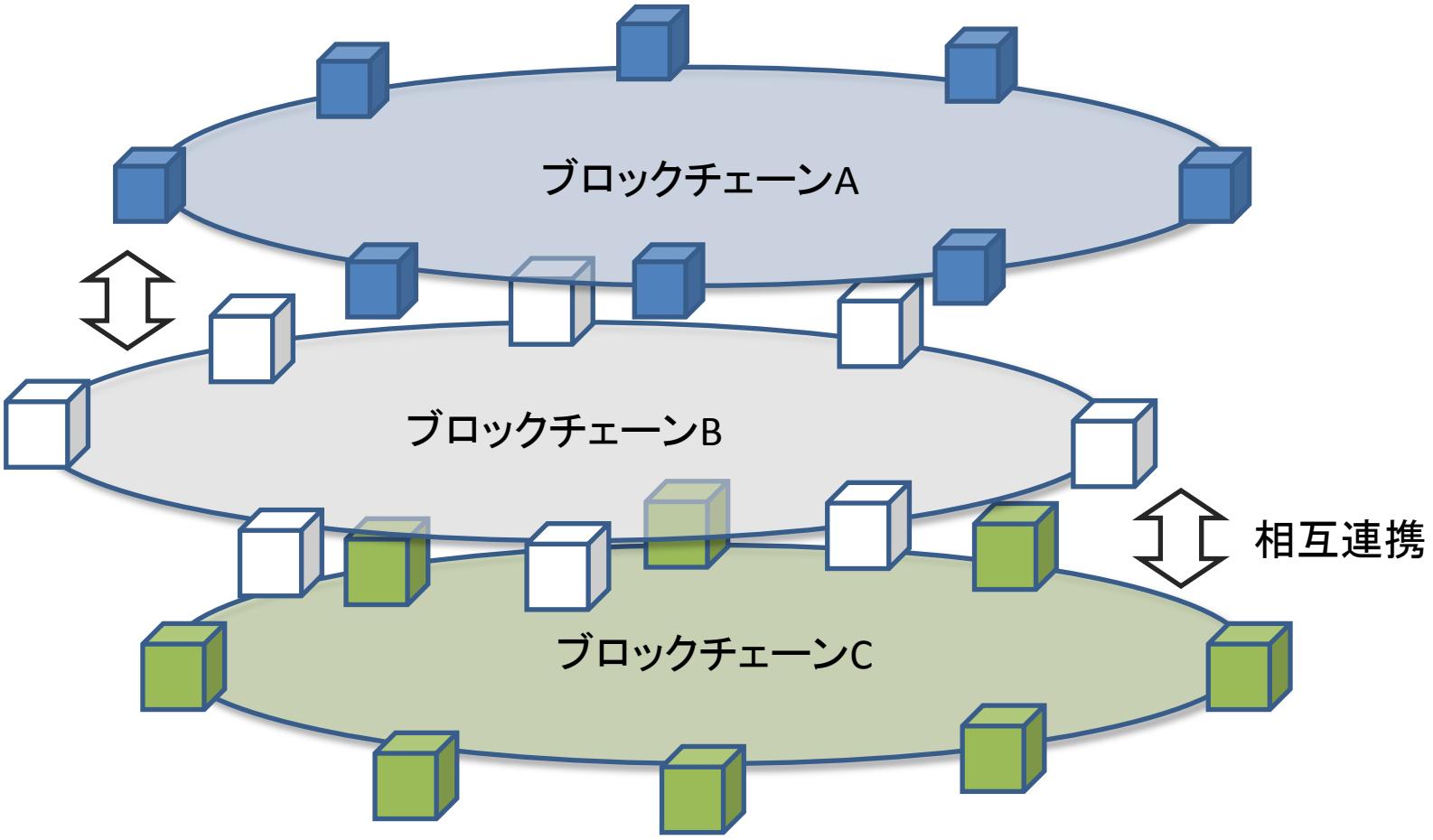
インターネット(1974)

現在のブロックチェーンはこの状態

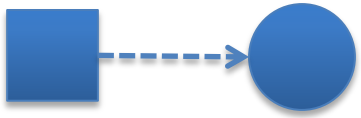
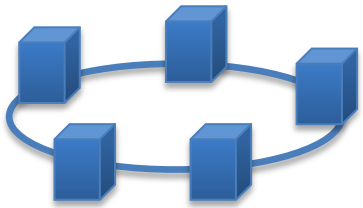


https://ja.wikipedia.org/wiki/ARPANET#mediaviewer/File:Arpanet_1974.svg

World of many blockchains

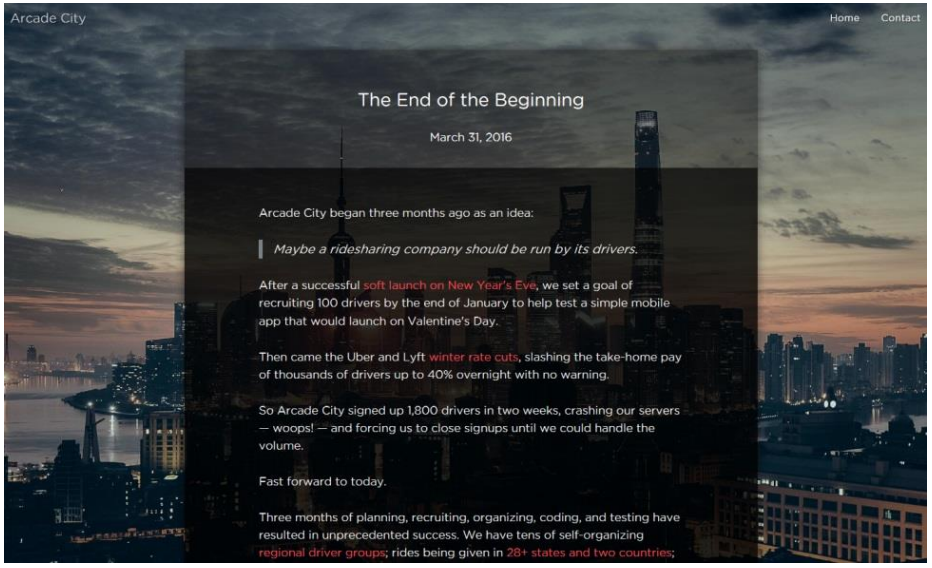


ブロックチェーンからビジネスモデルをどう発想するか？

3大要素	着想のヒント
データの連携による改ざん不可 	• **秘匿性はあまり求められないが、偽造されては困るもの** • **第三者を含めて確認・検証できることが望ましいもの** 例：各種登記、事業所登録、偽装防止（建築、自動車、決算）
エンティティと情報資産の紐付け 	• **資産の所有者・利用者が転々と変更** • **情報資産の状態が変化** 例：デジタルコンテンツの流通・課金、企業間の交換、電子書籍の中古販売、シェアリング、貸会議室等
P2Pによる管理者不要・信頼性向上 	• **耐障害性** • **分散型組織（DAO）の側面を活かしたサービス** 例： 航空発券　（但し、超高速処理には向かない） Arcade City(ライドシェア）、Open Bazaar（商取引）、Colony（仕事の受発注）、電力のP2P取引

Ethereum活用事例：Arcade City

- Uberのようなライドシェアリングのサービスを提供
- Ethereumの仕組みを活用した「ドライバーが所有するライドシェアリング企業」(1)
- ドライバーが料金を自由に設定、配達や乗客のサポートなど付加サービスも自由に行うなどの自律性
- DAO（Distributed Autonomous Organization）の一例



組織としての付加価値を高めるために、ドライバーが働くことで、自分のエクイティの持ち分が増える、あるいはエクイティの価値が上がるような仕組みが内在化されているか？

(1) http://www.prnewswire.com/news-releases/ridesharing-startup-arcade-city-launches-in-27-states-300235186.html

英国政府が提案する5つのユースケース

ケース1：重要インフラの防御
ブロックチェーン技術により、ソフトウェアの改ざんを即時に検知する仕組みを作り、重要インフラのソフトウェア改変による影響を防ぐ。

ケース2：社会保障支出の運用改善
社会保障支出に際して、仮想通貨等を利用することで受給者へ直接受け渡すことを可能にして、中間的な取引コストを削減する。また、ブロックチェーン技術で受給者のなりすまし等を防ぐことで、不正受給を防ぐ。

ケース3：国際援助の運用改善
仮想通貨によって国際送金にかかる為替コストを削減するとともに、スマートコントラクトを活用して、被援助者が現地政府の関与なしに自ら契約履行できる仕組みを構築する。また、中間組織を介在せずに、直接的に支援を必要とする人に支援物を届けることを可能にするとともに、本来の目的に沿った用途以外では使えないような仕組みを組み込む。

ケース4：取引コストの削減とイノベーションの推進
知的財産、特許、遺言、公正証書、ヘルスデータ、年金等の登録にブロックチェーン技術を活用することで、中小企業にとっての取引コストを削減する。また、マイクロペイメントの考え方等を活用して新たな公的業務とビジネスの運用コストを再発明する。市民は自分のパーソナルデータがどのように使われているのか管理できるようになる。

ケース5：付加価値税の徴税
スマートコントラクト等の普及により、取引を見える化し、徴税漏れを防止する。

出典：https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/492972/gs-16-1-distributed-ledger-technology.pdf

まとめ

- ブロックチェーンはネットワーク型のコンピューティング基盤へ
- 主体間の関係の定義・処理の自動化
- 業界標準をめぐる競争のスタート