招待論文

ブロックチェーン技術の社会実装に向けた観点*

佐古　和恵[†a)]　古川　諒[†]　小出　俊夫[†]

Aspects on Designs to Implement Blockchain Technology in Real World*

Kazue SAKO[†a)], Ryo FURUKAWA[†], and Toshio KOIDE[†]

あらまし　本論文では，ブロックチェーンを履歴データを管理する台帳モデルとして切り出し，そのモデルの見地に立った際のブロックチェーン技術の分類について論じる．本モデルがブロックチェーン技術の活用を検討するサービスやシステム設計者に対して，ブロックチェーン採用の一助となるべく，各分類に対しての課題や期待されるメリットについて述べる．更に，本モデルに基づいた社会実装に対する考察について，ブロックチェーンを活用した公平性を検証可能なオンラインゲームを一例に論じる．

キーワード　ブロックチェーン，ビットコイン，システムデザイン

1. まえがき

2008 年に発明されたビットコイン [1] が大きく注目されるなかで，そこで使われるブロックチェーン技術も，仮想通貨以外の応用を含めて，各社で開発がすすんでいる．しかし，学術的にこれぞ「ブロックチェーン」という定義が共通認識としてあるわけではない．サービスの実装・提供が先行し，それぞれが独自のモデルで「ブロックチェーン」を呼称した結果，技術的な共通要素で特色的なものをくくり出すことが困難になってしまった．また，ブロックチェーンの分類も permissionless，permissioned の他，open，public，private，consortium など，さまざまな呼称が使われ，整合した議論になりにくいのが現状である．

本論文では，その状況から一歩すすめるべく，ブロックチェーンを履歴データを管理する台帳モデルとして切り出し，permissioned と permissionless の分類を試みる．特に，ブロックチェーン活用を検討するサービスやシステム設計者に対して，ブロックチェーンの採用検討の一助となるべく，各分類に対しての課題や，期待されるメリットについて述べる．最後に，具体的にブロックチェーンを活用して公平性が検証可能なオンラインゲームを設計した際に，どのような考察を経てブロックチェーンを選択したかについて紹介する．

† NEC セキュリティ研究所，川崎市
NEC Security Research Labs., 1753 Shimonumabe, Nakahara-ku, Kawasaki-shi, 211–8666 Japan
a) E-mail: k-sako@ab.jp.nec.com
* 本論文は，システム開発・ソフトウェア開発論文である．
DOI: 10.14923/transcomj.2017JBI0001

2. 本論文で扱うブロックチェーンモデル

ブロックチェーンに関しての議論が明確になるように，本章では，ブロックチェーンの一つのモデルを設定する．「ブロックチェーン」を，履歴データを管理する「台帳」とみなし，図 1 に示されるような登場人物とその機能を想定する．

2.1 登場人物

想定する役割は下記のとおりである．(図 1 参照)

(1)　履歴データ発生者：台帳に登録すべき履歴データを発生する人．履歴データ成型機能を用いて成型した履歴データを履歴データ登録者に送付する．

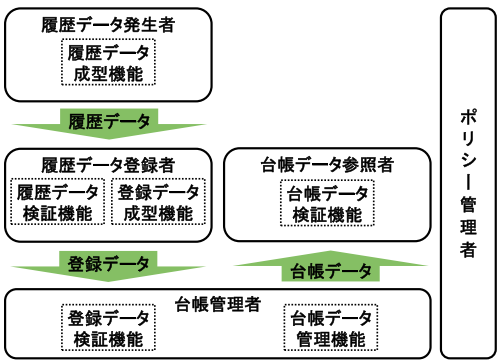

図 1　ブロックチェーンのモデル
Fig. 1　A blockchain model.

(2) 履歴データ登録者：履歴データ発生者から履歴データを受け取り，履歴データに対して一定の検証を行い，台帳へ登録するデータを成型する人．履歴データを，履歴データ検証機能を用いて検証し，また，最新の台帳データを参照し，登録データ成型機能を用いて成型した登録データを台帳管理者に送付する．

(3) 台帳管理者：履歴データ登録者から送付された履歴データを台帳として管理する人．登録データを，登録データ検証機能を使って検証し，それを台帳に追記する．また，台帳データ保持管理機能をもつ．

(4) 台帳参照者：台帳管理者が管理する台帳データを読み出す人．台帳管理者から受信した台帳データを，台帳データ検証機能を用いて検証する．

(5) ポリシー管理者：上記各種機能の仕様をきめ各役割を誰が担うかなどの方針を決める人．

2.2 ブロックチェーンとしての特徴

上記の五つの登場人物は履歴データを管理する台帳システムとして，自然なものである．ただ従来は，多数の「履歴データ発生者」及び「台帳参照者」に対して，「履歴データ登録者」，「台帳管理者」，「ポリシー管理者」の役割を一つの組織が兼ねて担うことが多く，これら三つの役割の違いが明確ではなかった．ブロックチェーンの新しい観点は，「履歴データ登録者」並びに「台帳管理者」が複数存在することである．しかも，この複数存在する「履歴データ登録者」や「台帳管理者」が，ポリシー管理者が想定するプロセスに必ずしも従わない場合も想定している．このように，必ずしも信頼できるとは限らない複数の履歴データ登録者並びに台帳管理者が，それぞれのデータを検証し合いながら，整合をとって台帳を管理運用することが，ブロックチェーンの特徴である．

分散データベースも，複数の台帳管理者がいる点では類似しているが，データ登録者が実質的に単独であることが大きな違いである．「実質的」と述べたのは，複数であってもどのデータ登録者も，ポリシー管理者が想定するプロセスに従って，登録データを決定するモデルである点が，ブロックチェーン技術と異なると言える．

なお，本論文では，ポリシー管理者の総意としてシステムのポリシーが決められるとするので，ポリシー管理者の数は問わない．

2.3 本モデルから見たビットコイン

ビットコインは「履歴データ管理台帳」を用いて安全な仮想通貨システムを実現している．その流れの概略を，上記のモデルに当てはめて解説する．

(1) 履歴データ発生者：台帳に登録する履歴データを発生する人は，仮想通貨を使用する人である．また，このときの履歴データは「B さんに X コインを支払う」というトランザクションデータである．履歴データ成型機能を通じて，自分が X コインを所有した利用者であることを示すデジタル署名が付与される．

(2) 履歴データ登録者：仮想通貨を使用する人から履歴データを受け取り，履歴データに対して一定の検証を行い，台帳に登録をする人は，「マイニングノード」と呼ばれる人である．過去の台帳を参照し，仮想通貨を使用する人のトランザクションの正しさを確認し，「ブロック」とよばれる，台帳に追記される登録データをつくり，台帳管理者（後述）に同報する．

ビットコインでは誰もがマイニングノードに立候補できる．台帳データ成型機能としては，Proof of Work とよばれるアルゴリズムが採用されている．Proof of Work ではマイニングに成功した証拠である情報とともに，ブロックを生成する．そして，複数のマイニングノードがそれぞれ生成したブロックのどれが台帳管理者に採用されるかは，早い者勝ちで決定される．

(3) 台帳管理者：履歴データ登録者であるマイニングノードが作成したブロックを台帳として管理する人である．この人は，マイニングノードが生成したブロックの正しさを確認し，正しければ，それを最新のブロックとして，台帳に追記する．このようにブロックがチェーン状に追記されつながることが，ブロックチェーンとよばれる所以である．なお，ビットコインではマイニングノード同様，誰でも台帳管理者になれる．ビットコインネットワークに加入し，近隣の台帳管理者のもつ台帳と同期した後，マイニングノードが同報するブロックを受信できればよい．あるいは，ビットコインユーザに台帳参照サービスを提供しているビットコインブロックエクスプローラ [2] も，台帳管理者としてみなせる．また，マイニングノード（履歴データ登録者）自身を台帳管理者であるとみなすことができる．

ちなみに，ビットコインでは誰でも台帳管理ができる一方，誰かが「台帳管理の責任」を負っているわけではない．同様に，誰も「履歴データ登録」の責任を負っているわけではないが，自分が成型したブロックが台帳に組み込まれればビットコインで報酬がもらえるため，競って履歴データ登録者に立候補するモチベーションがあるようにしかけが施されている．

(4)　台帳参照者：台帳管理者が管理する台帳データを読み出す人．ビットコインでは誰でも台帳を参照することができる．台帳を参照する必要が発生するのは，「仮想通貨の持ち主が，自分の通貨を台帳上で確認するとき」「仮想通貨の受け取り手が，自分が受け取り手であるトランザクションデータが台帳に登録されたことによって，仮想通貨を受け取れたことを確認するとき」「マイニングノードが，トランザクションの正しさを検証するとき」「マイニングノードが，最新の台帳データを参照するとき」「台帳管理者がブロックの正当性を確認するとき」「第三者が，仮想通貨が適切に運用されているか検証するとき」など，多くの場合がある．

(5)　ポリシー管理者：ビットコインにおいて基本的な仕様を定めた人は，Satoshi Nakamoto と名乗る人物である．Satoshi Nakamoto は，トランザクションデータにデジタル署名を付与するデータ仕様や，そのデータを検証するアルゴリズム，また，トランザクションデータをマイニングしてブロックに成型する仕様を決定した．その後，細かい実装上の仕様を更新決定しているのは，ビットコインデベロッパーと呼ばれるビットコインのソフトウェアを開発している人たちのコミュニティと，実際にマイニングをしているマイニングノードである．状況に応じて，過去の台帳の仕様には手を加えることなく，将来のブロックの満たす条件などを変更したり，データフォーマットを改良したりしている．ポリシー管理者間の熱い論争の一例として,「ブロック」の上限サイズが現状は 1MB であるところ，今後トランザクション数が増えたときの遅延を小さくするために，この上限を変更すべきかどうかが議論されている．

上記概説したように，ビットコインは，ビットコインネットワークに参加するノードの立候補によりマイニングノードが履歴データ登録者の役割を果たし，特別な台帳管理者を置くことなく，誰もが台帳を管理・参照できる仕組みになっている．また，マイニングノードがブロック生成に成功したあかつきには，成功報酬が支払われることで，マイニングノードに立候補するインセンティブも与えているのも，既述のとおりである．

各種検証アルゴリズムの詳細の解説は，文献 [1] や文献 [3] に委ねるが，デジタル署名やハッシュ関数の性質をうまく使うことにより，暗号アルゴリズムの安全性に基づく不正防止機能が実現されている．また，事前の登録処理などもなく，誰でもソフトウェアをインストールすれば仮想通貨を使用できる状況になるため，いったん仕様が固まった後は，全体の運用を管理する集中機関が不要であることも，画期的とされている点である．

社会実装の点からみると，ビットコインで採用されている Proof of Work のアルゴリズムが，安価な電力料金の国に在住するマイニングノードに有利に働き，また，ソフト開発者やマイニングノードがポリシー決定に関与できることから，将来的にどこまで健全性が保証されるかは不透明である．しかし，現在最も多くの利用者を巻き込んで実際に使用されているブロックチェーンの一つであることは間違いない．

2.4　本モデルから見た「東証のシステム」

JPX が 2016 年 8 月にブロックチェーンを用いた実証実験システムを報告した [4]．本システムは，市場管理者の存在のもと，金融機関が投資家の代行で証券の取り引きする際の取引データをブロックチェーンで管理するシステムである．提供する代表的な機能として，

- 会社の新規上場に伴う証券の新規発行
- 金融機関による証券取引の発注
- 取引成立データの記録
- 投資家による保有証券の確認
- 上場会社による証券保有者の確認

がある．

本システムを **2.1** のモデルの観点で見ると，下記のとおりになる．

(1)　履歴データ発生者：履歴データを発生し，台帳に登録をしようとする人は，証券を取引する人である．具体的には，証券を新規に発行する市場管理者，証券取引を発注する金融機関，取引成立後のデータを書き込む金融機関である．これらの機関はあらかじめ，市場管理者によってその権限を行使することが認められている人であり，別途 ID 登録をしている前提である．

(2)　履歴データ登録者：証券の取引（移転）データを受け取り，取引データに対して検証を行い，登録すべきデータであると認定する人は，報告 [4] では検証ノードと呼ばれている．具体的には，市場管理者と希望する金融機関からなる，あらかじめ指定された組織が運営する．この複数の検証ノードは，PBFT と呼ばれる合意形成アルゴリズム [5] を用いて，相互に登録すべき履歴データの整合性をとり，それぞれのノードが台帳を管理する．

(3)　台帳管理者：履歴データ登録者と同一である．

ただし，台帳として記録されている間は，各取引データは暗号化されており，台帳管理者は暗号化されている状態の取引データしか参照できない．

(4)　台帳参照者：台帳管理者が管理する台帳データを読み出す人は，市場管理者，金融機関，金融機関が管理する投資家，証券の発行主体である上場会社である．台帳参照者は，あらかじめ，市場管理者が運営する認証局に登録し，電子証明書を受領する必要がある．この電子証明書を用いた参照者認証が行われた後，参照者本人に関係した台帳データのみが提供される．

(5)　ポリシー管理者：本システムの全体の仕様を決めているのは，市場管理者と金融機関のコンソーシアムである．

本実証実験でのシステムは，ビットコインのような不特定多数が利用する仮想通貨ではなく，国内有数の金融機関が利用する証券取引を記録するシステムであるため，あらかじめ指定された履歴データ発生者，履歴データ登録者，台帳管理者，台帳参照者からなることが想定されている．また，他の金融機関に自社の取引内容をしられたくないという要件から，台帳管理者であっても，取引内容を参照できない仕組みも導入している．その他，文献 [4] では，金融市場業務との親和性，処理性能，可用性，コストの面から本システムを検証している．

なお，同システムでは「スマートコントラクト」を採用している．スマートコントラクトは，本論文のモデルの対象外にしたが，台帳に記載されたデータを基に，それぞれをプログラムデータ，状態データ，入力データとみなしてプログラム（コントラクト）を実行し，その結果を状態データとして台帳に追記するものである．

3.　ブロックチェーンの分類

3.1　履歴データ登録者の観点

2. のモデルを参照すると，台帳に記載される情報を大きく支配するのは「履歴データ登録者」である．履歴データ登録者が認めたものしか，台帳に記載されないからである．このような権限をどういう履歴データ登録者に提供するかという方針から，ブロックチェーンを分類することができる．

履歴データ登録者が複数存在することがブロックチェーンの特徴であるが，お互い顔がみえる固定された人のみが履歴データ登録者になれる方式が，分類の一つにあたる permissioned と呼ばれる方式であり，JPX の実証実験の方式はこれに属する．一方で，履歴データ登録者が固定であれば，その全員が結託すれば台帳の内容が支配できる懸念から，不特定多数の誰でも履歴データ登録者になれる方式を実現したのがビットコインやイーサリウム [6] である．本論文ではこれを permissionless と呼ぶ．ビットコインのインパクトは，自分以外にどのような履歴データ登録者がいるのかわからなくても，全体で一貫した台帳を管理可能な方式を編み出したことである．一方で，顔の見えない人が履歴データ登録者をしている不安と，履歴データ登録者の人数が限定されていることによって合意形成アルゴリズムが運用しやすいことから，JPX の例に代表されるように，permissioned の方式を好む業界もある．なお，巷にも permissioned，permissionless，あるいは，open，closed，public，private，あるいはコンソーシアム型という分類があるが，それらが一貫した定義が不明であるため，本論文では，独自にこのような定義を採用した．

まとめると，

(1)　permissionless：誰でも履歴データ登録者になれる可能性があるもの．例えば，ビットコイン，イーサリウムがこの分類に含まれる．ここに分類される方式は，履歴データ登録者だけでなく，履歴データ発生者，台帳管理者，台帳参照者のどの役割に対しても制限なく，誰でもなれることが多い．

(2)　permissioned：履歴データ登録者がポリシー管理者によって限定されているもの．JPX の方式はこちらに分類される．ここに分類される方式は，台帳管理者，台帳参照者も，場合によっては履歴データ発生者も，ポリシー管理者によって限定されることがある．

3.2　permissionless ブロックチェーンの課題

permissionless に分類されるブロックチェーンは，上述のとおり，多数のデータ登録者が存在しても，台帳の一貫性が担保される画期的なアルゴリズムが採用されている．しかし，その方式は様々な前提のもとに妥当性が保証されており，社会実装をする場合には，この前提が担保されている環境であるかを確認する必要がある．

例えば，ビットコインでは，多数の履歴データ登録者はそれぞれ同じ程度の計算力をもち，皆が平等にネットワークにつながっていることを暗黙の前提としている．この前提であれば，どの履歴データ登録者も公平に，同じ確率でブロックを生成できるため，ビットコインの方式を妥当とする根拠になっている．とこ

ろが実際には，この前提が成り立たない社会的な事情が，既に見え始めている．まず，同程度の計算力の前提については，マイニング専用のハードウェアが開発され，そのような設備投資を行った人の計算力があがっている．更に，地域によって電気料金に差があるため，同じ計算力を実現するためのコストが人によって異なる事情が発生している．更に，ネットワークの接続については，地理的なネットワーク環境（帯域や接続の安定性）の差により，人によってはデータ受信に遅延が発生する可能性がある．その結果，現在のビットコインでは，マイニングに成功するノードの半数は特定のマイナー組織になっている [7]．

また，ビットコイン方式では，台帳を管理するインセンティブがないので，増え続けるデータを管理してくれる台帳管理者の存続を心配する人もいる．インターネットもボランティアで運営され，世界隅々にまで到達できていることを考えると，ボランティアの登録者や台帳管理者がいなくなることは考えにくいかもしれないが，この点は既存の permissionless ブロックチェーンの採用にあたっては，考慮すべき点であろう．

既存の permissionless ブロックチェーンを採用する以外に，新規の permissionless ブロックチェーンを開発・構築することも選択肢としてはある．この場合は，台帳管理のインセンティブのみならず，履歴データ登録者のインセンティブもしっかり設計する必要がある．インセンティブがなく，数人の履歴データ登録者しか活動しない場合，いくら permissionless であっても，公平性が高いとはいえない．

3.3 permissioned ブロックチェーンの課題

上記の懸念から，permissioned ブロックチェーンに関心をよせる組織や企業は多い．これはサービスを提供する組織や企業が「ポリシー管理者」となり，データ登録者や台帳管理者をあらかじめ選定することができるから，台帳管理者が不在になったり，登録者が見ず知らずの人になってしまったりする懸念を回避できる．

一方で，permissioned ブロックチェーンに対する批判もある．そもそも「ブロックチェーン」という名称は permissionless であるビットコインで使われているものであった．そしてビットコインが支持され普及してきた背景には，ポリシー管理者によって台帳登録者が限定されないことであった．ポリシー管理者が台帳登録者を選定するのであれば，そもそも単一集権型の既存の分散データベースでロバスト性（一部のサーバが異常なふるまいをしても，台帳管理に影響がないこと）があればよいのではないか，という批判である．

これは，サービス設計者が，提供するサービスにおいて，何を信頼して，何を価値とみなすのかという観点から議論すべきことである．ビットコインでは，仮想通貨を使うユーザの観点から，ビットコインのソフトウェア実装とアルゴリズムが想定する仮定が成り立つと信頼し，ポリシー管理者やサービス提供者の意図によらず，自分自身の仮想通貨取引が成立することを価値とみている．実際の通貨が，サービス提供者である政府の意向により，銀行から引き出せなかったり，使えなくなったりすることに比較しての価値である．一方で，JPX の実証実験のように，そもそも関係者があらかじめ特定できるのであれば，不特定多数向けの仕組みを使う必要はないのである．このような場合であっても，例えば，上場企業や投資家への透明性の確保を容易にできるのであれば，それはブロックチェーンを活用する価値がある．

このように，どのようなサービスにおいて，誰の観点，誰の価値を追求するかによって，適切なブロックチェーン技術が異なってくるのである．

3.4 ブロックチェーンを使う理由

2.2 で議論したように，ブロックチェーンの特徴は，「履歴データ登録者」並びに「台帳管理者」が複数存在することによって，一部のサーバが異常なふるまいをしても，台帳データの読み出しに影響がないロバスト性を担保していることである．異常なふるまいには，ウイルスに感染するようなことも含まれる．分散データベースと異なり，それぞれの管理者がお互いに信頼できない場合でも，台帳から読みだされたデータで，検証機能により確認できたものは信頼できるような方式になっている．

そのことから，ブロックチェーンを検討する契機として，従来，単一の信頼されるサーバが存在すれば台帳管理が可能になる業務を，以下の理由で実現できないときに，活用を考えるのがよいと思われる．

1） サービス利用者から見て，信頼できるサーバが存在しない．
2） サービス提供者から見て，信頼できるサーバが存在しない．
3） サービス提供者から見て，信頼できるサーバを構築するのにはコストがかかる．

このうち，1）はビットコインの場合に相当する．ビットコインは，単一の通貨発行サーバあるいは通貨管理

サーバを信頼できない点から考案された．また，例えばサービス提供者にとっては信頼できるサーバであっても，そのサーバをサービス利用者が信頼してくれそうにない場合もここに含まれる．

2）の例としては，例えば，同業者からなるコンソーシアムの集合体としてポリシーを管理している場合がある．コンソーシアムといえども，メンバーがライバル関係にある業者同士であれば，誰かひとりのメンバーに独占的にサーバ運営を任せられないかもしれない．その場合には，複数のメンバーで分散台帳管理をするのが有効な場合もある．

3）のコスト面に関しては，今後の検証が待たされるが，一つのサーバを信頼に足るようにしっかり運営管理するより，複数に分散させて，幾つかのノードに不具合があっても，アルゴリズムによって復旧できる方式を採用することにより，全体の管理コストが小さくできる可能性がある．また，今後，ブロックチェーン技術が普及し，標準化がすすむに従い，それぞれの異なるベンダーが開発した分散サーバを採用ことにより，マルチベンダー環境による信頼性の向上と，データ構造設計をはじめとするシステム設計時間の短縮化などの効率化も期待できる可能性がある．

4. 実装例：検証可能な乱数発生の実装

本章では，上記のブロックチェーンの性質を活用して，公平にサイコロの目が決められていることを示して，利用者に納得感のあるゲームサービスを提供する設計を紹介する．提案方式自体は既に文献 [8] で報告されているが，本章では，**3.** までの議論に即して，なぜこのようなブロックチェーンを選択したかの思考過程を述べる．

4.1 概　　要

文献 [8] では，バックギャモンなどのオンラインゲームにおいて，サーバが発生させるサイコロの目が，ゲームの進行に応じてサーバが意図的に操作していないことを，プレーヤーが検証できる方式を提案している．具体的には，ゲーム開始前にサーバとプレーヤーが決定した「シード」をもとに，あらかじめ公開されている疑似乱数発生器の仕様に基づいて，サイコロの目を決定していることを，ゲーム後に確認できるようにしている．一方で，プレーヤーが事前にサイコロの目を予測してゲームを進行しないように，サーバが選んだシードは，プレーヤーがシードを選ぶ前に「ハッシュ値」として台帳に記録する．プレーヤーは，ゲーム終了後に，サーバからシードを入手し，台帳の「ハッシュ値」と突合し，サーバのシードが正しいこと，更に，サーバが生成したサイコロの目が適切であることを確認できる．そのためには，台帳の「ハッシュ値」が書き換えられていないことを担保する必要があり，そのために，ブロックチェーンが活用できる．

4.2 オンラインゲームシステムの要件の検証

本システムでは，「履歴データ発生者」がゲームサーバ，「台帳データ登録者」と「台帳データ管理者」がビットコインのノード，「台帳データ参照者」は，ゲームプレーヤーや問題があったときの調停者となる．また，「ポリシー管理者」はビットコインのブロックチェーンのポリシー管理者となる．

本例において，**3.4** に議論した「ブロックチェーンを利用する理由」について述べる．本システムは，サービス提供者であるゲーム会社から見て，サービス利用者であるゲームプレーヤーに，不当なサービスではないことと納得してもらうのが目的である．ゲーム会社から見て信頼できるサーバが存在しなかったわけではない．コストをかけて，信頼サーバを構築したとしても，ゲームプレーヤーの納得が得られなければ，意味がないのである．したがって，ゲームプレーヤーから見て信頼できるデータ管理サーバとして，ブロックチェーンが採用された．

次に，どのようなブロックチェーンを採用するか，という議論になる．permissioned と permissionless があるが，permissioned にするには，ゲーム会社がポリシー管理者としてふるまい，データ登録者を指名する必要があるが，果たして，そのようなデータ登録者がゲームプレーヤーから信頼してもらえるか，という問題になる．では，permissionless にしてこのシード値のコミットメントを記載するだけの台帳管理に協力してくれるデータ登録者や台帳管理者がいるか，という課題に直面する．そこで，ゲーム会社とは別に，既に確立しているブロックチェーンの上に，このシード値のコミットメント情報を登録することにしたのである．

実際，ビットコインのブロックチェーンは，ビットコインの取引データの上に，一定のビット数以内であれば，任意のメモ情報を追記することができる．ビットコインの取引を，ゲーム会社が所有する二つの口座間の資金移動とし，その移動データにシード値のハッシュ値（256 ビット）をメモとして記載すれば，資金移動記録とともに，このメモの情報もブロックチェーン上に残り，改ざんされないように記録される．資金

移動のための手数料は，毎回減額されることになるが，自前でブロックチェーンを運営するコストと比較すれば，その負担はゲーム会社にとって許容範囲であろう．

また，ビットコインのブロックチェーンのデータは誰でも読める仕様になっているが，ここに記載するのは乱数用のシードのハッシュ値のみであり，プレーヤーの個人情報は記載しないことから，これを誰かにみられても，問題が発生するとは考えにくい．

なお，ビットコインのトランザクションが確定して台帳に記載されるまでには時間がかかるが，ゲームの進行とは独立にすることができる．具体的には，ゲーム開始時に，乱数用のシードのハッシュ値をメモ欄に記載したトランザクションの ID をプレーヤーに伝えたのち，ゲームは進行する．その時点でそのトランザクションが台帳に存在しなくても，プレーヤーはゲーム終了後に，そのトランザクション ID で台帳を検索して，乱数用シードのハッシュ値を確認できれば，ゲーム中にサーバが不正なく，サイコロの目を提示していたこを確認できる．

4.3 考察及び今後の発展

本システムでは，プレーヤーとゲームサーバの 2 者しかいない状況で，プレーヤーが「ゲームサーバがサイコロの目を操作しているのでは」という不信感を，ブロックチェーンという中立な，ゲームサーバであっても関与できない台帳を利用することによって解決した．

ブロックチェーンを使わなくても，台帳に記載すべきデータを，ゲームサーバ側でデジタル署名を付与してゲーム開始前に提供することが可能である．しかし，この場合は，乱数発生の正しさは 2 者間でしか共有できない．しかも，例えばプレーヤーがゲーム会社が送ったデータを「受け取っていない，送られていない」といいはった場合には，水掛け論になってしまう．ブロックチェーンに記載することで，データ授受に関する曖昧性が解消されるのである．

将来は，例えば，監査人に対しても，正当に乱数発生を実施していることを示せるような発展形も考えられる．また，仮想通貨の基盤と連動させることにより，様々なゲーム界の取引の正当性を可視化する方向にも寄与できると考えている．

5. むすび

本論文では，信頼できる単一のサーバが存在しない環境においても，複数のノードでロバストなデータ管理を実現するブロックチェーンを，履歴データを管理する台帳モデルとして切り出した．また，ビットコインや JPX における実証システムについて本モデルに基づいた整理を行った．更に，本モデルを用いてブロックチェーン技術を分類し，それらに期待される性質を論じることで，システム設計者がブロックチェーンを適用して実装するにあたっての指針を紹介した．そして，そのような指針に基づいた社会実装例として，公平性を検証可能なオンラインゲームの設計について論じた．

一方で，ユースケースの特徴をしっかり押さえておかないと，今の時点では，なぜブロックチェーンでなければならいのか，という疑問の声もあがりかねない．しかし，ブロックチェーン導入可能性の観点から，既存のシステムを見直すと，旧前の慣例にのっとった不要なプロセスがあぶりだされるかもしれない．必ずしも現状はブロックチェーンでなくてよいシステムであっても，例えば，データ連携レイヤーの新しい標準仕様に基づいたシステム設計にすると，データ連携システムの設計コストが低減されたり，今後のシステム拡張が容易になるようなメリットも期待されている．前提が異なるシステムの連携には細心の注意が必要ではあるが，ハイプが過ぎた後にも，人間社会に貢献する骨太な技術として残り，様々なイノベーティブなサービスを産み出していくことを願ってやまない．

文　献

[1] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Crash System,” 2008. https://bitcoin.org/bitcoin.pdf

[2] ビットコインブロックエクスプローラ, https://blockchain.info/

[3] 佐古和恵, “公平性と透明性を実現するブロックチェーン,” 情報処理, vol.57, no.9, pp.864–869, Sept. 2016.

[4] 山藤敦史, 箕輪郁雄, 保坂　豪, 早川　聡, 近藤真史, 一木信吾, 金子裕紀, “金融市場インフラに対する分散型台帳技術の適用可能性について,” JPX ワーキング・ペーパー, no.15, Aug. 2016. http://www.jpx.co.jp/corporate/research-study/working-paper/tvdivq0000008q5y-att/JPX_working_paper_No15.pdf[5]

[5] M. Castro and B. Liskov, “Practical Byzantine fault tolerance,” Proc. Third Symposium on Operating Systems Design and Implementation, pp.173–186, 1999.

[6] Ethereum, “A next-generation smart contract and decentralized application platform,” White Paper, 2016, https://github.com/ethereum/wiki/wiki/White-Paper

[7] ビットコインブロックエクスプローラ, Bitcoin Hashrate Distribution, https://blockchain.info/pools

[8] 佐古和恵，井口圭一，“ブロックチェーンを用いたオンラインゲーム用公平性を検証可能な乱数発生，” 暗号と情報セキュリティシンポジウム，SCIS2017-2E4, Jan. 2017.

（平成 29 年 3 月 29 日受付，6 月 28 日再受付，
7 月 18 日早期公開）

佐古　和恵　（正員）

京都大学・理（数学）卒．卒業後 NEC 中央研究所に入社．以来，セキュリティの向上やプライバシー保護，公平性保証を目的とした暗号プロトコルの研究に従事．日本学術会議連携会員．博士（工学）．

古川　　諒

2008 年東京工業大学総合理工学研究科博士前期課程了．同年 NEC 入社．以来，アクセス制御，プライバシー保護，ブロックチェーンの研究に従事．

小出　俊夫　（正員）

2004 年創価大学大学院工学研究科博士後期課程了．同年 NEC 入社．以来中央研究所にて DHT 等の分散データベース，SDN/NFV 等のネットワーク制御の研究を経て，現在ブロックチェーンの研究に従事．博士（工学）．