不正転売を防止するブロックチェーンベースのチケット管理システムの実装に向けた詳細設計

2016SC001 相崎聖也　2016SC009 古田健斗
指導教員：石原靖哲

1　はじめに

コンサートなどのチケット販売において，「転売ヤー（転売屋）」と呼ばれる業者や個人が存在する．彼らは，希少価値の高いコンサートなどのチケットを，販売価格の数倍の値段で転売し，利益を得ようとする．このような行為は不正転売と呼ばれている．不正転売は，転売屋によるチケットの買い占めなどで，さらにそのチケットの希少性を高め価格を高騰させる．そのために，不正転売は正規の価格での購入を困難にする要因となっている．加えて，令和元年 6 月 14 日からは「特定興行入場券の不正転売の禁止等による興行入場券の適正な流通の確保に関する法律」，通称，チケット不正転売禁止法 [1] が施行されている．この法律では，特定興行入場券について，その不正転売の禁止及び不正転売目的の譲受けの禁止が定められており，社会的にも大きく問題視されていることが伺える．

本研究では，チケットの販売額面，または，それに送料などのチケット発行に関連する手数料を加えた額を正規転売価格とし，正規転売価格での取引を正規転売，それを超える価格での取引を不正転売と定義する．不正転売を防ぐためにチケットの転売そのものを禁止すると，正当な理由による正規価格での転売も不可能になる．急用などにより行けなくなったイベントのチケットを他の購入希望者に譲渡すること自体は，その売り手や買い手，またはイベントを開催している興行主にとって望ましいことである．

以上の問題の解決案として，中川ら [2] が，「正規転売を許容し，不正転売を防止すること」を実現するブロックチェーンベースのチケット管理システムを提案している．それを元に，システムの実装に向けた詳細設計を行うことを本研究の目的とする．

2　関連研究

2.1　ブロックチェーン

ブロックチェーンは，Satoshi Nakamoto によって提案されたビットコイン [3] の中核技術を原型とする P2P ネットワークを採用した仕組みの一つで，分散台帳技術，または，分散型ネットワークとも呼ばれる．ネットワーク上に分散する，ビザンチン障害を含む不特定多数のノードにデータを保持させることで，高可用性及びデータ同一性等を実現する．

ブロックチェーンはデータの書き込み権限によって大きく 2 つに分類される．一つは，ブロックチェーンにデータの書き込み権限がないパーミッションレス型ブロックチェーンで，ビットコインやイーサリアム [4] などで使用されている．もう一つは，ブロックチェーンにデータ書き込み制限があるパーミッションド型ブロックチェーンで，Hyperledger Fabric などで使用されている．

2.2　ブロックチェーンベースのチケット転売システム

現在，不正転売を防ぐ仕組みを備えたチケット管理システムがいくつか存在している．ブロックチェーンを利用したシステムは，Aventus[5] などのパーミッションレス型ブロックチェーンに基づくものと，中川らのシステム [2] などのパーミッションド型ブロックチェーンに基づくものに分類される．パーミッションレス型に基づくシステムは誰でも参加可能であるため，チケットの流動性が高くなるというメリットがある．しかし，チケット売買記録を格納した台帳に誰でもアクセスできてしまうため，たとえ購入者の ID や氏名を仮名化していたとしても，イベントに参加した履歴をトレースすることで個人を特定できてしまうリスクもある．一方，パーミッションド型に基づくシステムでは，台帳を不特定多数に公開しない設定が可能であるため，台帳にアクセスできる参加者を限定できる．この点では，パーミッションド型のほうがチケット転売システムに向いていると考える．

中川らは，Hyperledger Fabric とディジタル署名を用いて，発券済みチケットの失効機能と，チケット流通にかかわる複数者間でのデータ管理機能を備えたチケット管理システムを提案している．このシステムは，パーミッションド型ブロックチェーンを用いることで，中央管理者を置かずに，複数者間でデータの記録，共有及び改ざん防止を実現できる．それにより，自社以外の会社への問い合わせなどなしにチケット所有情報の確認が可能になるため，他社で発行されたチケットでも失効機能を実現可能になる．この失効機能を利用して，転売元のチケットを失効後，転売先へのチケット発行を確実に実行し，入場できないリスクを無くすことができる．加えて，チケット転売サービス会社の仲介で転売元からの買取と転売先への販売を別々に行うため，両者はお互いを知ることもなく，結託を防ぐことができる．

2.3　Hyperledger Fabric

Hyperledger Fabric は Linux Foundation が管理する Fabric プロジェクトの一つであり，オープンソースのパーミッションド型のブロックチェーン基盤のことを指す．Hyperledger Fabric はモジュラー型のアーキテクチャであるため拡張性が高く，ビジネス向けの分散型アプリケーションの開発・運用に適している．

Hyperledger Fabric では，ブロックチェーンネットワークに参画する組織を表す論理的な単位のことを Organization という．Organization は主に Peer，Orderer，CA の 3 種類の構成要素を持ち，それらは以下のような役割を持つ．

- Peer:クライアントから送付されたトランザクションの検証や実行，ブロックへの書き込みを行う．
- Orderer:検証済みのトランザクションの順序付けを行い，ブロックチェーンネットワーク内の Peer に送信する．
- CA:ブロックチェーンネットワーク内のユーザと Peer の情報の登録，証明書の発行をする．

また，Peer は stateDB とブロックチェーンから成る台帳を有しており，同じく有するスマートコントラクトを実行することで台帳の状態を変化させる．スマートコントラクトはブロックチェーンネットワーク上で動作するプログラムである．

3 詳細設計対象のチケット管理システム

中川らが提案しているチケット管理システムは，興行会社，チケット販売会社，転売サービス会社，イベント会社の 4 つの組織がブロックチェーンネットワークのノードとなり，チケットの販売からイベント入場までの処理を行う．図 1 に当該システムにおけるチケットの販売及び転売の流れを示す．はじめに，興行会社がブロックチェーン上に記録する複数のイベントを識別するためのイベント ID を決める．そのイベントの全てのチケットに対して，チケット ID を準備する．次に，チケット販売会社がチケット ID 毎にディジタル署名の鍵ペア（検証鍵，秘密鍵) を発行する．秘密鍵はチケットとしてチケット購入者に送付され，検証鍵は署名の検証用としてブロックチェーン上に記録される．イベント入場時には，イベント会社が，入場用メッセージとチケット購入者自身の秘密鍵を使用して作成された署名に対し，検証鍵を用いてチケットの正当性を検証する．署名が承認されるとブロックチェーン上にその署名を記録し，入場を完了させる．また，チケットを転売する場合には，転売サービス会社が転売用メッセージと秘密鍵を使用して作成された署名に対し，検証鍵を用いてチケットが正当であることを検証する．正当性が証明されるとその署名はブロックチェーン上に記録され，そのタイミングではじめてチケット再発行が可能となり，転売を開始できる．

4 詳細設計

4.1 システムの全体像

本研究では，イベントの状態を管理するイベント管理者やチケット購入希望者及びチケット購入者の利用を想定して設計を行った．本システムはユーザインタフェースである Web アプリケーション，トランザクションの生成やチケットの発行などを行うサーバサイドアプリケーション，トランザクションをもとにデータの処理または台帳への記

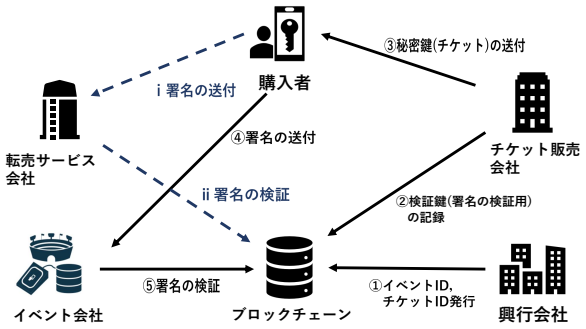

図 1 システムにおけるチケット販売および転売の流れ

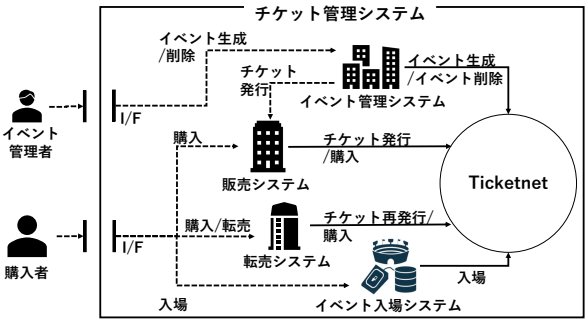

図 2 チケット管理システムの全体像

録などを行うブロックチェーンネットワークの 3 要素で構成する．サーバサイドアプリケーションについては，イベント管理システム，チケット販売システム，転売システム，イベント入場システムの 4 つのシステムを構成要素とする．また，本システムで構成するブロックチェーンネットワークを Ticketnet と呼ぶ．図 2 は本システムの全体像を表しており，実線の矢印はトランザクションを，破線の矢印はサーバサイドアプリケーションへのリクエストを示す．

4.2 Ticketnet

Ticketnet には興行会社，チケット販売会社，転売サービス会社，イベント会社の 4 つの Organization のブロックチェーン処理に関わるコードが含まれる．各 Organization は Orderer を一つずつ備える構成とする．Orderer は Hyperledger Fabric ネットワーク上で発生する全トランザクションを制御するため，ネットワーク内でただ一つであっても機能上問題はないが，単一障害点とならないように冗長的な構成をとった．このネットワーク上で，イベント及びチケットに関する情報の記録・共有を行う．

4.3 スマートコントラクトの設計

Ticketnet 上で動作するスマートコントラクトとして，イベントに対して処理を行うイベントコントラクトと，チケットに対して処理を行うチケットコントラクトの 2 つを定義した．

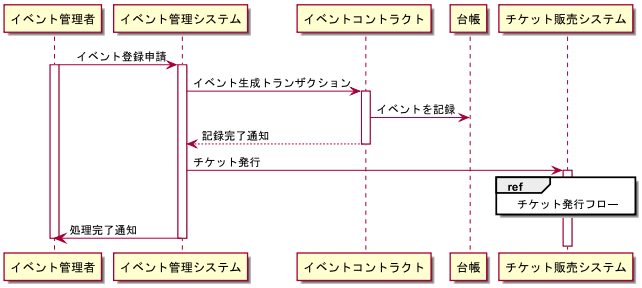

図 3　イベント生成フローのシーケンス図

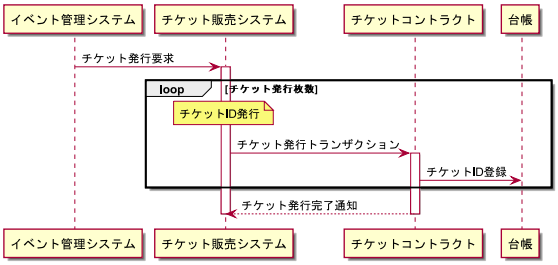

図 4　チケット発行フローのシーケンス図

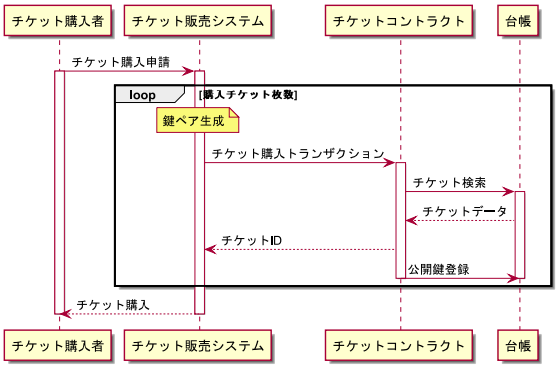

図 5　チケット購入フローのシーケンス図

4.3.1　イベントコントラクト

イベントは，イベント ID をキーとして興行会社 ID，チケット発行枚数，イベントのステータスを属性に持つ．これらのデータは，トランザクションの発行によりスマートコントラクトが実行されることで，書き込み及び更新される．しかし，トランザクション発行後以降の処理が適切に実行されるかどうかは，台帳に記録されたステータスに依存している．イベントのステータスは，開始状態からイベント生成トランザクションにより開催状態へ遷移し，その後イベント削除トランザクションにより閉幕状態へ遷移する．これらの処理はイベント生成フローとイベント削除フローに分けて実行する．以下，それぞれの処理における目的を記す．

イベント生成フロー

イベント管理者がイベントの生成を行うフローである（図 3）．イベント管理者がイベント管理システムへ入力したイベント ID，興行会社 ID，チケット発行枚数に加え，イベントが開幕状態であることを台帳へ記録する．その上，イベント管理システムは，イベントの情報と共にチケット発行のリクエストをチケット販売システムへ送信する．

イベント削除フロー

イベント管理者が生成済みのイベントを削除するフローである．イベント管理者は，イベント管理システムに削除したいイベントのイベント ID を入力する．台帳に該当するイベントが存在すれば，イベントが閉幕状態であることを記録し，削除完了とする．

4.3.2　チケットコントラクト

チケットは，チケット ID をキーとしてイベント ID，チケットのステータス，メッセージ，署名，検証鍵を属性に持つ．チケットのステータスは，チケット発行トランザクションにより開始状態から販売状態に遷移し，購入トランザクションを受けて購入済み状態になる．その後，入場トランザクションを受け取ると入場済み状態へ遷移するが，代わりに転売トランザクションを受け取るともう一度販売状態へと戻る．これらの処理はチケット発行フロー，チケット購入フロー，チケット転売フロー，イベント入場フローの 4 つに分けて実行する．以下，同様にそれぞれの処理の目的を記す．

チケット発行フロー

チケット販売システムがチケットの発行処理を行うフローである（図 4）．イベント生成フローにより送られてきたイベントの情報を受け取り，チケットの発行枚数分のチケット ID の発行処理を行う．また，台帳に対し，チケット ID 毎にイベント ID とチケットが販売状態であることを記録する．

チケット購入フロー

チケット購入希望者がチケットを購入するフローである（図 5）．チケット購入希望者がチケット販売システムにイベント ID とチケット購入枚数を入力する．チケット販売システムは，購入枚数分だけペアとなる秘密鍵と検証鍵を発行する．検証鍵は台帳に記録され，秘密鍵は台帳から取得したチケット ID と共に購入者へ送信される．なお，転売チケットの購入も同様の処理が行われる．

チケット転売フロー

チケット購入者が転売を行うフローである（図 6）．チケット購入者はチケット転売システムにチケット ID と秘密鍵を入力する．チケット転売システムは，秘密鍵を使用して作成された署名を，台帳に記録されている，チケット ID と対応する検証鍵で検証する．署名が正当だった場合，署名を台帳に記録し，チケットのステータスを販売状態に書き換える．

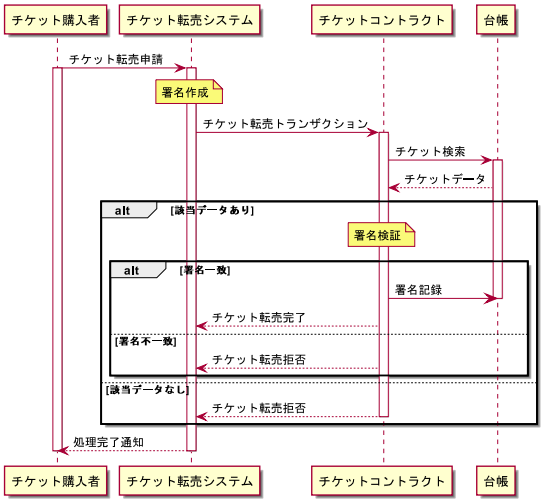

図 6　チケット転売フローのシーケンス図

イベント入場フロー

チケット購入者がイベントへの入場を行うフローである．処理の流れはチケット転売フローと同様であり，最後にチケットのステータスを入場済み状態へ書き換える．

4.4　詳細設計の正当性

本研究では，チケットやイベントの状態遷移に注目して詳細設計の正当性を定義する．具体的には，中川らが意図したとおりに状態遷移が定義されているとき，詳細設計は正当であると定義する．

チケットやイベントの状態は，トランザクションの実行によって遷移する．例えば，本詳細設計において，購入済み状態のチケットに対して転売トランザクションや入場トランザクションは実行できるが購入トランザクションは実行できない．イベントの状態についても同様に，例えば開催状態でないイベントに対して生成トランザクションは実行できるが，削除トランザクションは実行できない．

以上のように，本詳細設計では中川らが意図したとおりに状態遷移が定義されていることを確認した．

5　実装

本研究で行った詳細設計からシステムの実装を試みた．スマートコントラクトとサーバサイドアプリケーションについては，100 行から 150 行程度の複数のサンプルプログラムのコードリーディングをし，変更及び拡張を行った．Ticketnet については，docker-compose ファイルに Ticketnet の構造を記述し，また，Ticketnet へ接続するための設定ファイル，Hyperledger Fabric における暗号プロトコルを正常に動作させるための設定ファイルなどの変更を行った．

動作確認を行い，それぞれイベント，チケットのインスタンスを生成し，ユーザからの入力を元にインスタンスのプロパティが設定されることを確認した．また，イベント，チケットがブロックチェーンに記録されることも確認した．しかしながら，現時点で正常に動作しているのは，イベントの生成フローとチケットの生成フローに関する処理だけである．

6　まとめ

本研究では，中川らが提案したチケット管理システムを元に詳細設計を行い，その正当性を確認した．さらに実装を試みたが，正常に動作するのはイベントの生成フローとチケットの生成フローに関する処理のみに留まった．実装を完成させられなかった原因の考察及び，そこから得られた教訓を記す．

- プログラムのエラーに対する適切な対処の仕方を調査せず，場当たり的なデバッグを行ってしまった．分割統治法を用いたり，デバッグコードをプログラムの要所要所に挿入するなどしてバグの在処を徐々に絞り込み，時間の浪費を防ぐべきだった．
- ブロックチェーンもしくは Hyperledger Fabric に精通した人が周りにおらず，初歩的なミスについても助言を得ることができなかった．自身の能力を加味し，研究課題を適切に設定すべきだった．
- いきなり自らが実装したプログラムを実行するのではなく，Hyperledger Fabric に付属するサンプルコードから徐々に変更，もしくは拡張することから始め，バグの発生を最小限に抑える努力をするべきだった．
- Hyperledger Fabric やブロックチェーンに特有の概念や理解する必要のある事柄が多く，そこに時間がかかった．そのため，あらかじめ必要な時間を予測し，計画的に調査を進める必要があった．

今後の課題は，実装途中である本システムを完成させることである．また，リレーショナルデータベースを用いた既存のシステムなどと比較して性能評価を行い，実用性を検証することである．

参考文献

[1] チケット不正転売禁止法 | 文化庁. http://www.bunka.go.jp/seisaku/bunka_gyosei/ticket_resale_ban/index.html. 最終閲覧日 2020 年 1 月 12 日.

[2] 中川紗菜美, 佐古和恵, 小出俊夫, 梶ヶ谷圭祐. 不正転売問題を配慮したブロックチェーンベースのチケット管理システムの提案. 2018 年 暗号と情報セキュリティシンポジウム，4F1-4, 2018.

[3] Satoshi Nakamoto. Bitcoin: A peer-to-perr electronic cash system. 2008.

[4] Ethereum Project. https://www.ethereum.org. 最終閲覧日 2020 年 1 月 12 日.

[5] Aventus Protocol - A digital assets-focused blockchain-based protocol. https://aventus.io. 最終閲覧日 2020 年 1 月 13 日.