設計ガイド：オンチェーンとオフチェーンの使い分け

設計ガイドについて

ブロックチェーンを利用するアプリケーションは、従来のアプリケーションとは異なる特徴や制約を持ち、それらに対応した設計が必要です。設計ガイドでは、ブロックチェーンを利用するアプリケーションの開発に関する重要なポイントについて、ひとつずつトピックを取り上げ、説明していきます。今回のトピックは、オンチェーンとオフチェーンの使い分けについてです。

ベストプラクティスや事例を交えながら、ブロックチェーンの基本原理や概念に沿って効果的にブロックチェーンを活用するための指針を示します。ブロックチェーンに関する一般的な内容は、ブロックチェーンの共通の特性や仕組みに基づいて説明します。東芝独自のブロックチェーン DNCWARE Blockchain+™ (BC+)に関する固有の視点は、このプラットフォームで利用可能な機能に触れながら紹介します。BC+固有の視点は、[BC+]というラベルを付けて区別します。

はじめに

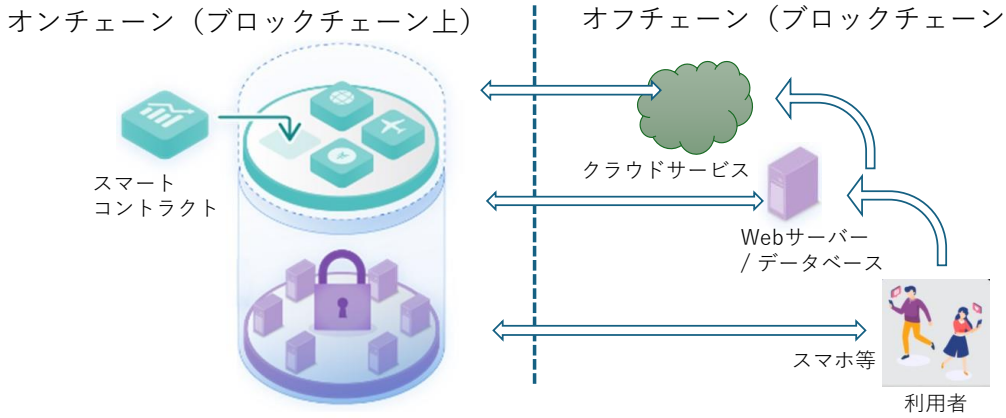
この設計ガイドでは、オンチェーンとオフチェーンの使い分けについて、以下のような観点で指針を示します。

- オンチェーンとオフチェーンの特徴やメリット・デメリットを理解する
- オンチェーンとオフチェーンの使い分けのステージを知る
- 事例を通して考える（分散型自立組織(DAO)）
- そのほかの話題　（オンチェーンの処理時間の軽減、オフチェーン・データの保存）

それでは、オンチェーンとオフチェーンの違いについて説明していきます。

オンチェーンとオフチェーンとは

- オンチェーンは、ブロックチェーン上で処理することを意味する言葉です。
- オフチェーンは、ブロックチェーンの外で処理することを意味する言葉です。
- オンチェーンとオフチェーンは、それぞれ異なる特徴を持っています。



上図の左側がオンチェーン、右側がオフチェーンになります。

ブロックチェーン上で処理を行うことをオンチェーン、ブロックチェーンの外側で処理を行うことはオフチェーンという定義なので、ブロックチェーン以外の既存のシステムはすべてオフチェーンに分類されます。ブロックチェーンの利用にあたっては、利用者はオフチェーンを経由してオンチェーンにアクセスする形となり、必然的にオンチェーンとオフチェーンをさまざまに組み合わせて使うことになります。

ブロックチェーンは、分散化されたノードがトランザクションをそれぞれ検証することで、改ざん耐性の高い記録を作る技術です。しかし、この方式には性能面での制約があります。そこで、ブロックチェーン上に記録する必要のないトランザクションやデータは、ブロックチェーンの外（オフチェーン）で処理し記録することで、この問題を回避することができます。

オンチェーンとオフチェーンの特徴

オンチェーンの特徴は、ブロックチェーンの特徴そのものであり、非中央集権的、トラストレスであって、改ざん耐性、高可用性、ビザンチン障害耐性(BFT)があり、記録は永続的で消すことができません。また、処理速度が遅い、データサイズに制限があるといったデメリットがあります。

一方、オフチェーンの特徴は、(オフチェーンにもいろいろな技術があり一概には言えないのですが) 典型的には、オンチェーンと性質が逆になります。つまり、中央集権的で、管理者へのトラストが前提、管理者による改ざんが容易であり、記録は消すことができます。オンチェーンと比較して、処理速度が速い、大規模なデータが扱えるといったメリットがあります。

オンチェーンとオフチェーンの使い分けは、アプリケーションの性質や目的によって異なりますが、一般的には、ブロックチェーンの特徴を活用したい場合はオンチェーン、ブロックチェーンの制約を回避したい場合はオフチェーンを選択します。

オンチェーンとオフチェーンは、このように異なる特徴を持っていますが、中央集権と非中央集権のどちらが優れているかという対立の図式には意味がなく、オンチェーンで行う処理と、オフチェーンで行う処理の選別を、異なる特徴を踏まえて適切に設計し、アプリケーションの目的に適合させることが重要なポイントです。

オンチェーン	**オフチェーン**
非中央集権的、トラストレス	中央集権的、管理者へのトラストが前提
改ざん耐性、高可用性、BFT がある	管理者による改ざんが容易
記録が消せない	記録が消せる
処理速度が遅い、スケーラビリティが低い	処理速度が速い
データサイズに制限がある	大規模データも処理可能

オンチェーンとオフチェーンの使い分け

オンチェーンとオフチェーンの使い分けについて、ここでは、オンチェーンの利用の割合が異なる４つのステージを示します。

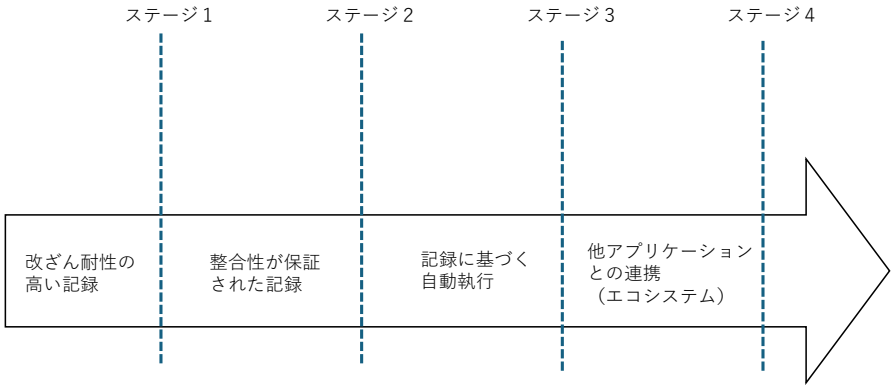
- ステージ１：改ざん耐性の高い記録
- ステージ２：整合性が保証された記録
- ステージ３：記録に基づく自動執行
- ステージ４：他アプリケーションとの連携を含んだエコシステム

この４つのステージは、オンチェーンの処理の割合が少ないものから順にならんでいます。つまり、ステージ１がオンチェーンの処理の割合がもっとも少なく、ステージ４がオンチェーンの処理の割合が最も多くなっています。

オンチェーンで行う処理とオフチェーンで行う処理の選別の設計が重要になることは上述したとおりですが、最初から最適な割合で実現するのではなく、徐々に最適な割合に近づけていくという方法が有効な場合もあります。

DX（デジタルトランスフォーメーション）一般に言えることですが、一足飛びに最適なゴールへ到達しようとすると、ギャップが大きすぎてうまくいかないということが多々あります。そのような懸念がある場合には、トランスフォーメーションを徐々にすすめていき、最終的な時点で目標にゴールするように計画することが有効です。

ブロックチェーン化（非中央集権化）においてもこれと同様なことが言えます。オンチェーンの割合が少ない変革からスモールスタートで始め、徐々にオンチェーンの利用の割合を増やしていく計画が一般に有効であると考えられます。



ステージ１：改ざん耐性の高い記録

ブロックチェーンに記録されたデータは改ざんが困難であるため、単に情報をブロックチェーンに登録するだけで、改ざんに強い記録が作成可能です。これはブロックチェーンの最も基本的な利用方法ですが、それだけでも価値あるアプリケーションが作成可能です。

例として、日報を管理するアプリケーションを挙げます。このシステムには、ブロックチェーンが導入されており、ユーザーが日報データを入力すると、そのデータがブロックチェーン上に不変の履歴として保存されます。一度登録されると、これらのデータは改ざんできず、後からの編集や削除も不可能です。さらに、日報の入力時刻もブロックチェーンによって自動で記録されるので、後日になって書き換えるなどの不正はすぐに見破られます。日報の管理システムがこの種の改ざん耐性を備えていることは理想的です。

ところで、報告データのサイズがブロックチェーンに保存するには大き過ぎる場合も考えられます。その際は、実際の報告内容をオフチェーンで保管し、そのハッシュ値をブロックチェーンに登録することで、同様の改ざん防止効果を確保する手段が取れます。なぜなら、ブロックチェーンに登録されたハッシュ値と、オフチェーンに保管された実際の日報データのハッシュ値とを比較することで日報の真正性を検証できるためです。ハッシュ値とは、もとのデータから生成される指紋のようなもので、データが改竄されていないことを確かめる目的で使用されます。すなわち、オフチェーンに保管された日報データの真正性は、ブロックチェーンに登録されたハッシュ値との照合によって保証されます。この方法により、データ量が多い問題を解決しつつ、データの改竄を防ぐ効果を維持できます。

秘匿性の高い報告内容をブロックチェーンに直接保存したくない時も、同様の方式が使えます。ブロックチェーンにはハッシュ値だけが登録されており、ハッシュ値から元のデータを再構築することは不可能です。そのため、秘匿性が確保されます。オフチェーンに保管された報告内容へのアクセス制御は、オフチェーンでの一般的な技術が使えます。

[BC+]

ブロックチェーン基盤として BC+を用いている場合には、ブロックチェーン上のアクセス権を設定する方法を利用することもできます。ブロックチェーンに保存された記録へのアクセスをオンチェーンで制限することで、例えば報告者とその上司などの関係者だけがその記録を閲覧できるようにすることが可能です。

ステージ２：整合性が保証された記録

ブロックチェーンにはスマートコントラクトと呼ばれる機能が備わっています。スマートコントラクトは、ブロックチェーンに実装されるプログラムであり、アプリケーションの目的に合わせてカスタマイズすることができます。このプログラムを使って、ブロックチェーンに送られた入力が規則を満たしているかどうかをチェックし、その条件を満たす入力のみを受け付けることが可能になります。このような仕組みで、整合性が保証された記録を実現することができます。

ステージ１の例において、日報を入力するスマートコントラクトに書式規則をプログラムすることで、日報の形式を統一することができ、検索や管理が容易になるでしょう。

また、すでに報告済みの内容を修正する再報告について、新規の報告と区別して扱う規則をプログラムすることもできます。具体的には、同一の日付での報告がすでに記録されている場合には、新規としての報告は受け付けずに再報告のみを受け付けるようにスマートコントラクトで入力時にチェックします。再報告の場合には、ひとつ前の報告へのリンクを追加してブロックチェーンに記録することもできます。こうすることで、虚偽の複数の報告を意図的に行い、後になって有利な情報を選択して公開するような不正行為を防止することが可能になります。

入力データの整合性はオフチェーンでもチェックできますが、管理者による改ざんのリスクがあります。そのため、組織ぐるみの不正が問題になっている場合、オフチェーンの検証だけでは信頼性を証明できません。一方、スマートコントラクトはブロックチェーン上で実行されており、実行結果の改竄は非常に困難で、ノード管理者であっても改ざんすることができません。このため、組織ぐるみの不正が疑われている場合であっても、スマートコントラクトによる検証プロセスは信頼できます。つまり、ブロックチェーンによる検証は暗号技術のみによって支えられており、特定の組織や人の信頼性に依存しない「トラストレス」システムなので、改ざんされていないことの証明が客観的にできるのです。

投票アプリケーションも興味深い例の一つです。これは特に DAO などで使用されることが想定されますが、DAO に関しては後続の章で詳しく取り扱います。投票アプリケーションの核心は、有権者の投票をブロックチェーンに記録することにあります。これだけでも改ざんに対する耐性を備え、大きな価値があると言えます（ステージ１に相当）。ここにさらにスマートコントラクトによる下記の検証を組み入れることで、投票の整合性を担保することが可能です（ステージ２に相当）。

検証項目

- 資格のない者の投票がないこと
- 二重投票がないこと
- 投票期間外に行われた投票がないこと

検証の流れは下記のようになります。まず、有権者は自身のブロックチェーン・ウォレットを使って投票トランザクションにデジタル署名を行い、ブロックチェーンに入力します。これを受け付けるスマートコントラクトは、トランザクションのデジタル署名を行った人に投票の権利があるかを有権者のリストと照合してチェックします。次に、同じ人がすでに投票を行っていないかをブロックチェーンの記録と照合してチェックします。また、そもそも投票期間内であるかどうかもチェックします。これらのチェックをすべて通過した投票のみがブロックチェーンに登録されます。こうすることで、ブロックチェーンに登録されている投票は上記の 3 つの整合性をすべて満たしていることが客観的に保証されます。

すこし話が逸れますが、一般的なブロックチェーンでは、個人の識別にウォレットアドレスが使われます。例えば、「NFT のオーナーは A さんである」というとき、ブロックチェーンに実際に記録されているのは、A さんのウォレットアドレスです。上記の投票アプリケーションでも同様の実装にすることが一つの方法です。

[BC+]

一方 BC+では、個人の識別にウォレットアドレスを直接使うことは推奨されません。BC+には個人を識別するための ID としてユーザーID が用意されており、ブロックチェーンによってランダムに割り当てられた分散型 ID（DID）として機能します。ユーザーID と所有するウォレットアドレスをあらかじめ紐づけることで個人を識別する仕組みです。ウォレットアドレスで直接に個人を識別する方法に比べて、ユーザーID を間に挟むことにより、運用の柔軟性が増すことが特徴となります。BC+を利用する投票アプリケーションの場合、投票を行う有権者はユーザーID によって識別されるようになります。

ステージ３：記録に基づく自動執行

スマートコントラクトを利用すれば、ブロックチェーン上の記録に基づいて自動的に処理を実行できます。例えば、NFT を購入すると仮想通貨がウォレットから自動的に支払われるなどです。スマートコントラクトを使った自動執行は、ブロックチェーンテクノロジーの代表的な機能です。これにより、非中央集権型のアプリケーションの自動化が実現可能となります。実用的なスマートコントラクトの多くがこの形をしています。

このステージ 3 で重要なことは、ステージ 2 の検証がしっかりと設計されていることを確認することです。不十分な検証は、矛盾した記録が生じる可能性を生み出し、それに基づく自動執行が行われた結果、予期せぬエラーが発生する可能性があります。これは悪意のあるユーザーによって悪用されうる弱点にもなります。

ブロックチェーンの完全性を保つためには、オンチェーンでの整合性の検証が求められます。同じような検証はオフチェーンでも、例えば入力フォームにおいて実施されています。しかし、それはオンチェーンでの検証を不要にする理由にはなりません。オフチェーンでの検証は改ざんが用意であり、ブロックチェーンの完全性に寄与するものではないからです。もちろん、入力フォームでの検証は、入力エラーを即座にユーザーに知らせるなどの利点があり、その意味では価値があるため、結論としてはオンチェーンとオフチェーンの両方で整合性をチェックするのが望ましいアプローチと言えます。

ステージ３の例として、仮想通貨を挙げます。仮想通貨は、ブロックチェーンに最初から組み込まれているものもありますが、スマートコントラクトを使って実現することも出来ます。イーサリアムの場合には、これは ERC20 トークンと呼ばれ、ICO などでよく利用されています。

仮想通貨のスマートコントラクトの主要な機能のひとつである送信(Transfer)について簡単に説明します。利用者が仮想通貨の送信を行う際に、数量と送信元と送信先をスマートコントラクトに入力しますが、このときスマートコントラクトでは、下記のチェックを行います。

・トランザクションの発行者が送信元と一致していること
・送信元が入力された数量以上の仮想通貨を所持していること
・入力された数量が正の値であること

このチェックがすべて通った時にスマートコントラクトが送信を自動的に執行します。すなわち、入力された数量の仮想通貨を送信元の持ち分から減算し、同量の仮想通貨を送信先の持ち分に加算します。

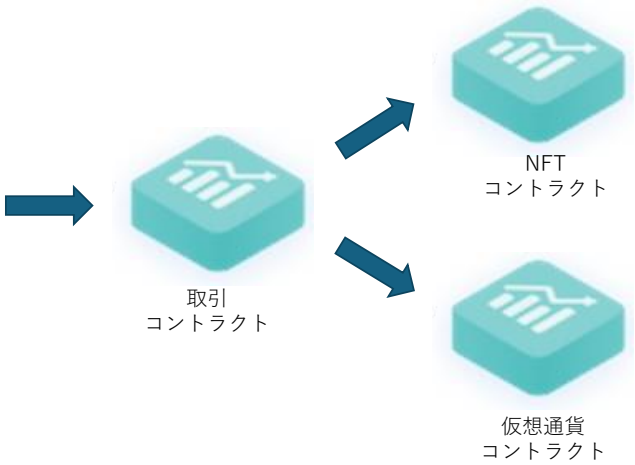
これらの一連の処理はすべてオンチェーンで行われ、アトミックに処理されます。したがって、残高チェックから送信の執行の間に、横から残高が引き出されるという心配はありませんし、送信の執行の最中に障害停止したとしても、一連の処理がすべて巻き戻されるため、中途半端な状態で完了するという心配もありません。

スマートコントラクトから別のスマートコントラクトを呼び出すこともできます。これによりより複雑な自動執行が可能になります。例として NFT のオーナーの移転と仮想通貨の支払いを同時に行う取引を考えます。

NFT と仮想通貨は、それぞれ別々にスマートコントラクトで実現されています。そして、取引を自動執行するスマートコントラクトもそれらとは別にあります。この 3 つのスマートコントラクトが連動して取引が自動的に執行されます。

取引時の処理の概略を簡単に説明します。まず取引コントラクトは入力の整合性チェックを行います。整合性が確認されると取引の執行を始めます。取引コントラクトは NFT コントラクトを呼び出し、NFT のオーナーの移転を指示します。また、取引コントラクトは仮想通貨コントラクトを呼び出し、仮想通貨の送信を指示します。この 2 つの呼び出しが成功した後、取引コントラクトは処理を成功として完了します。(事前に取引コントラクトへの権限の委譲(approve)が必要であるなどの細部を省略して説明しています)

スマートコントラクトの呼び出しを含むこれらの一連の処理は、オンチェーンでアトミックに実行されます。つまり、一連の処理の中で一か所でもエラーが起きれば、全体の処理がすべて巻き戻されます。そのため、複雑な処理であっても中途半端な状態で完了するという心配がなく、正確に執行されることが保証されます。



ステージ４：他アプリケーションとの連携を含んだエコシステム

最後のステージとして、ブロックチェーンが最大限に活用された理想の姿を示します。ブロックチェーンの目標の一つは、信頼関係が薄く利害が異なる複数の組織であっても、それらをブロックチェーンを通じて結び付けることです。多くの組織をブロックチェーンを介して緩く結合し、いままでにはなかったエコシステムをブロックチェーン上に形成することができれば理想的です。

ブロックチェーンは透明性のあるスマートコントラクトによって統治される無人の自動執行の世界です。ブロックチェーンを介することにより、組織間のコミュニケーションや取引のあり方が変革されます。従来は契約を通じて行われていた、人による組織間の取引をデジタルトランスフォーメーションしたものと解釈できます。また、ブロックチェーン上に記録された組織間の取引履歴は改ざん耐性を持ち、客観的に信頼できる証拠として活用できます。さらに、ブロックチェーン上に構築された他のアプリケーションとの連携をすすめていけば、より複雑で多層的なエコシステムを形成し、そこにあらたなビジネスが生まれることも想像できます。

事例：分散型自立組織(DAO)

DAO とは、分散型自立組織（Decentralized Autonomous Organization）の略で、ブロックチェーン技術を用いて運営される組織のことです。DAO は、中央集権的な管理者や仲介者が存在せず、参加者が自律的に協働することで、目的や価値観に沿った活動を行います。DAO の特徴は以下の通りです。

- ブロックチェーン上にスマートコントラクトを配置し、そのプログラムが組織のルールやガバナンスを定義します。
- 参加者は、ブロックチェーン上の NFT のオーナーとなることで、DAO の一員となり意思決定に参加します。
- 参加者の投票や提案によって、DAO の方針などを決めることができます。スマートコントラクトによって、投票結果は透明性と信頼性をもって自動執行されます。
- DAO は、ブロックチェーンの不変性によって、外部からの干渉に対して耐性を持ちます。

DAO は、従来の法人組織とは異なる新しい組織形態であり、国境や地域を超えて、共通の目標やビジョンを持つ人々が集まって事業やプロジェクトを展開することを目標とします。

このような特徴を持つ DAO ですが、2024 年のガートナーのハイプサイクル[1]によれば、黎明期をぬけピーク期に差し掛かったところで、主流の採用まで 5〜10 年を要するとされています。まだ発展途上のテクノロジーであり、注意すべき点は多くあります。

まず、組織の規則をスマートコントラクトで記述することが困難です。これは、言語の翻訳の難しさということではなく、対象がデジタル化されていなければ、規則だけをデジタル化することができないというところに起因する難しさです。つまり、組織全体のデジタル化が同時に必要です。

次に、「外部からの干渉に対して耐性を持つ」という点が困難です。スマートコントラクトの設計に欠陥があれば、無人で自動執行されるブロックチェーンは悪意ある者にかえって利用されやすく、容易に組織が乗っ取られる可能性があります。そのため、スマートコントラクトの自動執行の設計には、念には念を入れた慎重さが求められます。

それでは、DAO を具体例として設計の進め方について詳しく説明していきます。なお、ここで述べる内容はあくまで例であり、必ずしもこの通りに進めなければいけないわけではなく、このガイドが成功を保証するものでもありません。
ここからは、DAO を段階的にブロックチェーン上に構築していきます。

---

[1] Gartner、「日本における未来志向型インフラ・テクノロジのハイプ・サイクル：2024 年」を発表

最初の段階として、DAO で意思決定した事項を記録するためのスマートコントラクトを用意します。意思決定する内容の種類に応じて、スマートコントラクトを別々に作成するのが良いでしょう。たとえば、DAO のメンバー管理（追加・削除や役割変更など）に対応するスマートコントラクトと、収益分配のパラメーターを管理するスマートコントラクトを別々に作成します。この段階では、単に決定事項をブロックチェーンに記録するだけの構成ですが、DAO としての意思決定の種類を設計するという点において重要です。なお、最初からすべてを網羅するのは難しいでしょうから、あとから意思決定のスマートコントラクトを追加できるアーキテクチャとしておくのがよいでしょう。

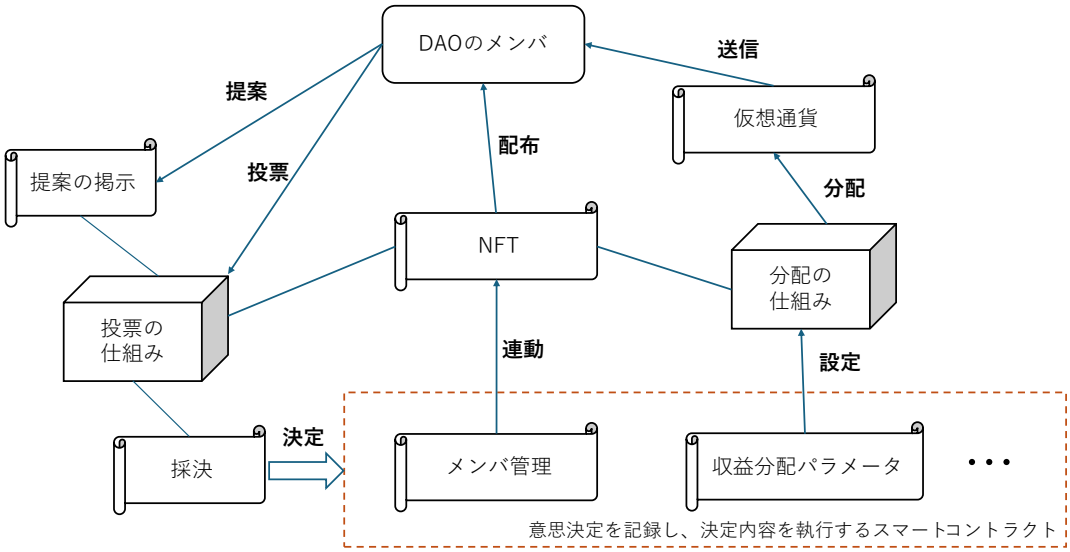
次に、DAO のメンバーの役割を表す NFT のスマートコントラクトを作成し、現在のメンバーに配布します。そして、メンバー管理のスマートコントラクトと NFT スマートコントラクトが連動し、メンバーの変更に自動追随するようプログラムします。たとえば、DAO が新しいメンバーの追加を決定し、その情報をメンバー管理のスマートコントラクトに入力すると、その新規メンバーに NFT が自動的に配布されるようにします。

つづいて、投票システムの段階に進みます。DAO における投票システムとは、提案に対して投票権を持つメンバーが投票を行い、一定の賛成票を得た場合、その提案が DAO の決定として承認される仕組みのことです。前の段階で導入された NFT のオーナーが投票権をもち、それぞれの立場で投票を行います。採決の結果、提案が可決されれば、それが意思決定のスマートコントラクトに自動的に入力され、ブロックチェーンに記録されます。

投票システムの導入によって、DAO の意思決定が自律的に行われるようになりますが、提案から採決までの時間を十分長くとり、悪意のある提案を中心メンバーが拒否できるようにしておくなど、中央集権的な要素をいくぶん残しつつ、慎重に導入を進めるといった工夫も必要でしょう。

最終的な段階としては、意思決定のスマートコントラクトによる決定内容の自動執行があります。例えば、収益分配のパラメーターを変更するケースを考えます。これをブロックチェーン上で自動的に実行するためには、仮想通貨を利用した収益の分配をスマートコントラクトに組み込んでおくことが事前に必要です。こうすることで、パラメーターが変更された場合、その変更が収益分配のスマートコントラクトに通知され、反映されます。この例からもわかるように、この段階では、組織の仕組み自体がブロックチェーン上で具現化されていることが必要になります。

これらの段階を経て DAO が完成します。実際には、この中間段階でも、オンチェーン処理とオフチェーン処理を適切に組み合わせることで、ＤＡＯとして十分に機能します。すべてをオンチェーンにする必要があるわけではありません。



オンチェーンの処理時間の軽減

オフチェーン処理からオンチェーン処理へ移行していった結果、オンチェーン処理に時間がかかりすぎるという問題が発生することがあります。これは、オフチェーン処理に比べてオンチェーン処理の性能が低いために起きます。

一つ目の対策は、複数のトランザクションを並行して要求することでスループットを向上させる方法です。これは、並行したトランザクションが一つのブロックとして処理されやすくするためのものです。

二つ目の対策は、複数の処理を一つのトランザクションに集約する方法です。処理の統合によって、処理量を減少させる効果が期待できます。

三つ目の対策は、一連の処理をオフチェーンで行い、定期的に結果をまとめてオンチェーンに反映させるというものです。一部の処理をオフチェーンにオフロードすることによる性能向上が期待できます。

このように、オンチェーンとオフチェーンの適切な組み合わせによって、ブロックチェーンのメリットを享受しながら、そのデメリットを軽減することができるのです。

BC+へのオフチェーン・データの保存

※この章はBC+の添付ファイル機能について説明しています。

一般的にブロックチェーンでは、オフチェーン・データはその名の通りチェーンの外部に保管されます。そして、データのハッシュ値をオンチェーンに記録することで、改ざん防止が保証されることを先述しました。この場合、オフチェーン・データを保管する場所がブロックチェーンの外に必要になりますが、簡易的な用途ではそのようなストレージを用意するコストが問題になる場合があります。

[BC+]

そこで、BC+では、オフチェーン・データをブロックチェーン・ノードに格納できる仕組みを用意しました。具体的には、トランザクションの添付ファイルという名目で、オフチェーン・データを後からノードにアップロードすることができます。トランザクション自体にはデータのハッシュ値しか含まれていませんので、オンチェーンに記録されるわけではありませんが、ノードにアップロードすれば格納できるという仕組みです。

なお、添付ファイルのアップロード時にはトランザクションに含まれるハッシュ値と整合しているかがチェックされるので、異なるデータをアップロードすることはできず、改ざん耐性があります。また、添付ファイルのダウンロードにはアクセス制限があります。これは、対応するトランザクションへのアクセス制限と一致する仕様となっています。

まとめ

ブロックチェーンを利用するアプリケーションの設計ガイドとして、オンチェーンとオフチェーンの特徴、メリット・デメリット、使い分けのステージを説明しました。事例を通して、効果的なブロックチェーンの活用例を紹介しました。