日本総研
The Japan Research Institute, Limited

モジュラーブロックチェーンの概要と潮流
～ブロックチェーンを性能向上させる技術～

株式会社日本総合研究所 先端技術ラボ

2024年9月2日

本レポートに関するお問い合わせ 先端技術ラボ 會田 拓海（aita.takumi.m2@jri.co.jp）

エグゼクティブサマリ

本レポートでは、ブロックチェーンの課題である性能(スケーラビリティ)向上に対し、解決策の1つとされるモジュラーブロックチェーンを概説する。

◆ 現状

ブロックチェーンでは処理性能の低さが課題とされ、改善に向けてレイヤー2[*1]やブロックチェーン間の連携が提案された。

システム開発領域でモノリス/モジュール(マイクロサービス)の議論があるように、ブロックチェーン領域では「モジュラーブロックチェーン」という概念が注目されている。

メインのブロックチェーン(レイヤー1)は主に検証を担当し、データ処理や保存を切り出すことを想定している。

◆ 展望

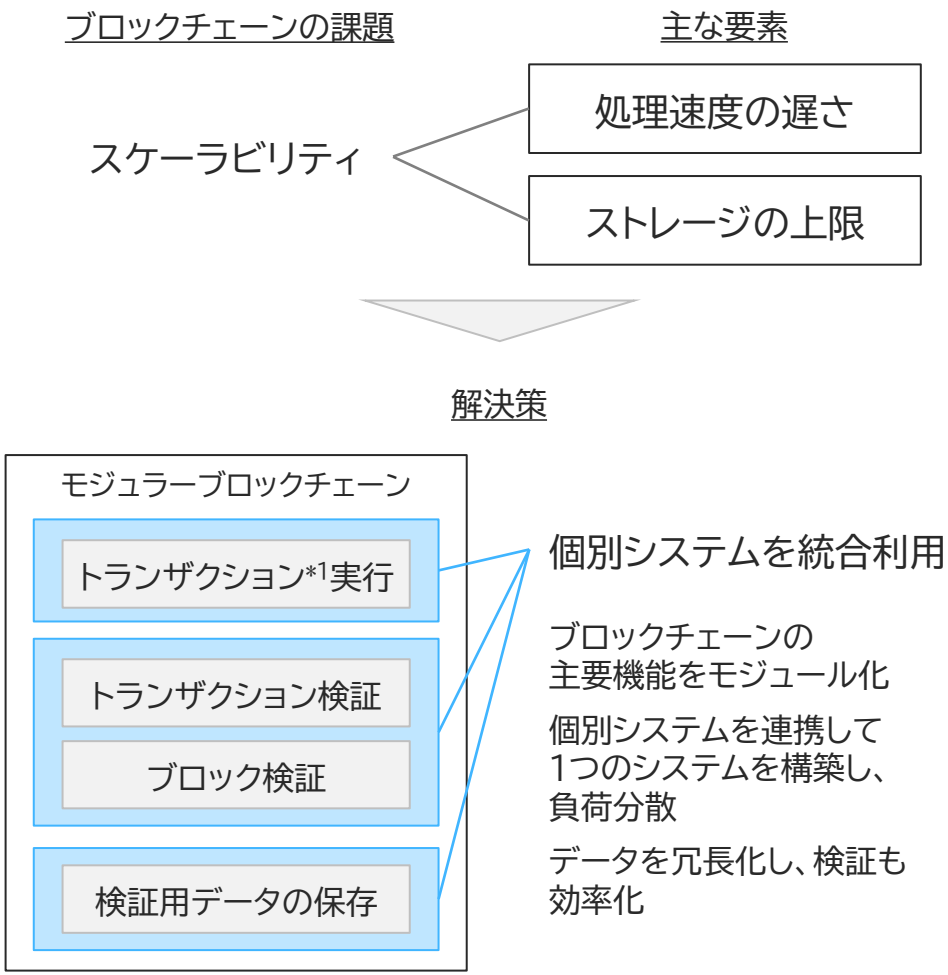
データ処理と検証・保存機能を分離して性能を向上させる取り組みは既に多く、ブロックチェーンの機能の切り出しは有用な手段になる。

複数のブロックチェーンの連携も高まっていくと予想される。

◆ 提言

機能の切り出し方法として、Rollupは事例が増えており、障壁も下がっていることから優良な候補となる。

ブロックチェーンの連携が高まればネットワーク効果も期待されるため、相互接続性の低い技術は導入を避けるべき。



1. レイヤー2とは、メインのブロックチェーン(レイヤー1)の外でトランザクションを実行し、その結果を保存することで性能を向上させる技術。
2. ブロックチェーンにおいて、データ本体やデータの処理内容を示すメッセージ。

目次

パブリックブロックチェーンが抱える課題については、先端技術ラボ発行レポート「パブリックブロックチェーンの技術動向 ～企業活用に向けた技術課題と現状～」を参照。

1.ブロックチェーンの性能向上に対する
モジュラーブロックチェーンのアプローチ

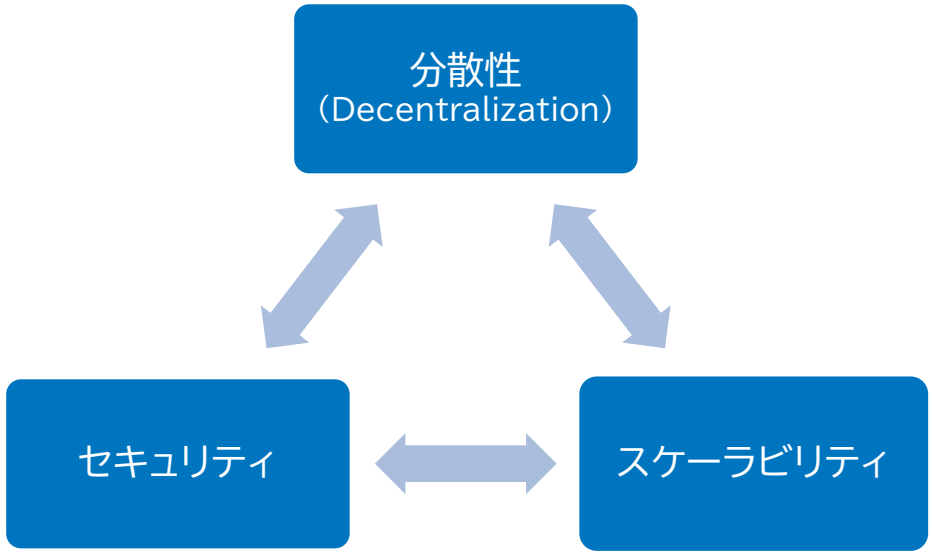
1.1 ブロックチェーンが抱える課題と運用形式

- ブロックチェーンの3つの性質「分散性・セキュリティ・スケーラビリティ（処理性能）」を同時に高めることは難しい。
- パブリック/プライベートブロックチェーンは、いずれかの性質を失う代わりに性能を高めている。

ブロックチェーンのトリレンマ

ブロックチェーンのトリレンマとは、次の3つの性質を同時に高めることは難しいという概念。

- 分散性：特定のノードに権限が集中していない
- セキュリティ：ブロックチェーンへの攻撃に抵抗できる
- スケーラビリティ：トランザクションの増大に耐えうる



ブロックチェーンの運用形式による性質の違い

パブリックブロックチェーン

不特定多数の運用者がノードを運用する。多数のノードが、安全なコンセンサスアルゴリズムを用いることで、分散性とセキュリティを保証する。

多数のノードでデータを処理・保存するため、データの同期に時間を要する。

プライベート/コンソーシアムブロックチェーン

特定少数の運用者が、少数ノードを集権的に運用することで、サイバー攻撃への耐性とデータ処理性能を向上させる。

ノード数が少数、かつ運営者を信頼できるため、データの同期に要する時間が短い。

性質＼ブロックチェーン	パブリック	プライベート/コンソーシアム
分散性	高	低
セキュリティ	高	高
スケーラビリティ	低	高

1.2 スケーラビリティ改善に向けたこれまでの取り組み

- ブロックチェーンのスケーラビリティ改善に向けて、コンセンサスアルゴリズムの改良やブロックチェーン群による水平スケーリング、レイヤー2の活用が提案されてきた。
- 近年では、複数のブロックチェーンやシステムを連携し、システム全体で性能を高める動きが強まっている。

単一のブロックチェーンの仕様変更で解決

コンセンサスアルゴリズムの改良（垂直スケーリング）

近年のブロックチェーンは、最初期と比べてコンセンサスが高速化。特に高TPS[*1]を掲げるブロックチェーンは、投票などで選ばれた検証者がデータ検証・承認を行うことで、処理性能を改善しつつ分散性を保つ。

一方、検証者を中心に合意形成を行うアルゴリズムは、システムの総負荷が運営者の処理性能に依存するため、システム全体の性能向上に限界があるという指摘もある。[*2]

複数のブロックチェーンやシステムを連携して解決

ブロックチェーン群による水平スケーリング

同様の機能をもつ複数のブロックチェーンを相互に運用することで、システム全体の処理性能を向上させる。

多数のアプリケーションが接続する汎用のブロックチェーンではなく、特定用途の基盤となるブロックチェーン（Appchain）という概念がある。

レイヤー2の活用

独立したシステムがトランザクションを処理した結果をメインのブロックチェーンに記録することで、処理を分散しつつセキュリティを保証する。

ブロックチェーンのトランザクション処理を切り出す方式として、サイドチェーン[*3]やRollup[*4]がある。

1. Transactions per second
2. Cohen, S., Goren, G., Kokoris-Kogias, L., Sonnino, A., & Spiegelman, A. (2023, May). Proof of Availability and Retrieval in a Modular Blockchain Architecture. In *International Conference on Financial Cryptography and Data Security* (pp. 36-53). Cham: Springer Nature Switzerland.
3. 独自のノードを用いて運営され、メインチェーン（レイヤー1）と双方向に接続されたブロックチェーン。
4. メインチェーン外（オフチェーン）でトランザクションを処理し、その結果をメインチェーンに書き込むシステム。

1.3 ブロックチェーンにおけるデータ処理の流れ

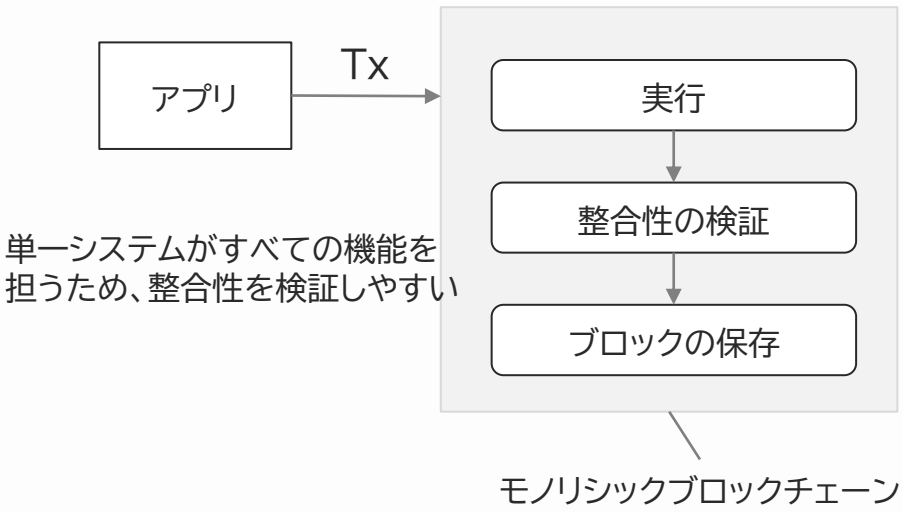
- 単一のブロックチェーンがすべての機能を提供する形態をモノリシックブロックチェーンと呼ぶ。
- ソフトウェア開発において、独立したサービスを連携し1つのサービスを提供する形態をマイクロサービスアーキテクチャと呼ぶ。ブロックチェーン領域では類似する概念をモジュラーブロックチェーンと呼ぶ。*1

(従来型)モノリシックブロックチェーン

モノリシックブロックチェーンとは、単一ブロックチェーンがデータの処理・検証・保存などすべての機能を提供するアーキテクチャ。

ソフトウェア開発におけるモノリシックアーキテクチャに類似する。

構成例とトランザクション(Tx)の流れ

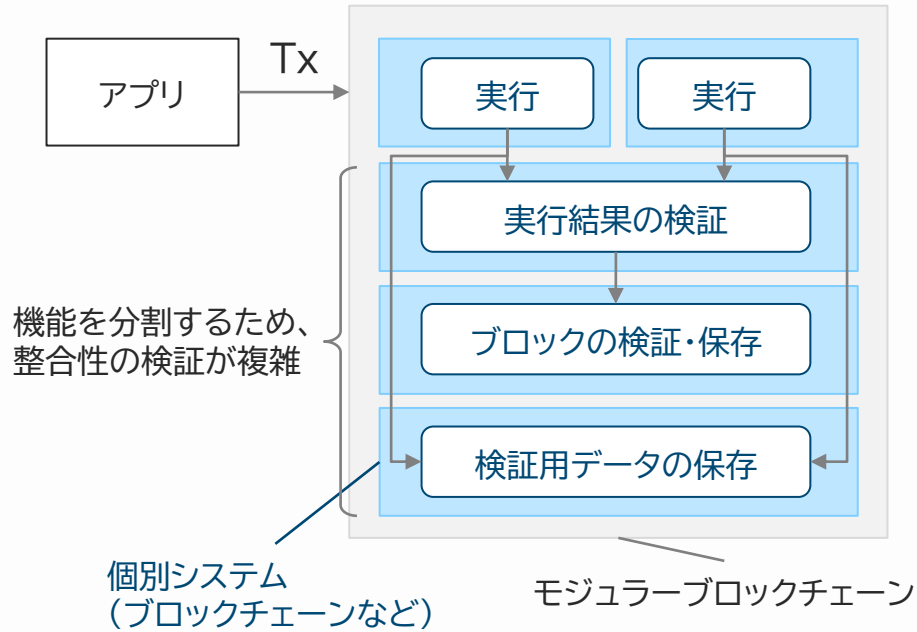


モジュラーブロックチェーン

モジュラーブロックチェーンとは、単一ブロックチェーンが担う主要機能を分割し、複数のブロックチェーンやシステムを連携して1つのシステムを構築するアーキテクチャ。

ソフトウェア開発におけるマイクロサービスアーキテクチャに類似する。

構成例とトランザクション(Tx)の流れ



1. モジュラーブロックチェーンにおける各機能は、単体で動作する。コードの再利用を目的とするのではなく、ブロックチェーンの性能向上、機能拡張を目的とする。

1.4 モジュラーブロックチェーンのアーキテクチャ概要

- モジュラーブロックチェーンとは、ブロックチェーンの主要機能を分割し、役割が異なる複数のシステムを連携して一つのシステムを構築するという概念。
- 従来のモノリシックブロックチェーンが担ってきた役割は、トランザクション(Tx)実行、実行結果の正しさに対する検証、Txの整合性の保証、検証データへのアクセスの保証に分けられる。

機能名称(レイヤー)	機能概要	機能詳細
Execution	Txを実行	署名済みTxを受け取って実行し、状態(state)を変更する。 スマートコントラクトの実行環境を提供する。
Settlement*1	Txの検証とブリッジを提供	Executionレイヤーから送信されるProof*2を検証する。 Executionレイヤーで動作するシステム(Rollupなど)間を繋ぐ。
Consensus	Txの整合性を保証	ノード間でTxの整合性(順序付け)に関する合意形成を行い、その結果についてファイナリティを提供する。
Data Availability	検証用データへのアクセスを保証	整合性の検証に用いるTxデータを常にダウンロードできる環境を整備し、検証者(ノード)に対して公開する。

(参考)"Monolithic vs. modular blockchain". Visa. https://usa.visa.com/solutions/crypto/monolithic-vs-modular-blockchain.html, (参照 2024-04-03), "What Are Modular Blockchains?". Binance Academy. https://academy.binance.com/en/articles/what-are-modular-blockchains, (参照 2024-04-04), "Settlement in the modular stack". Celestia. https://celestia.org/learn/modular-settlement-layers/settlement-in-the-modular-stack/, (参照 2024-04-04)

1. Settlementレイヤーは他のレイヤーに内包される場合もあるため、必須ではない。また、Settlementレイヤーを単体で提供するシステムは現状存在しない。
2. Txの計算結果の正しさを証明するデータ。

2. モジュラーブロックチェーンの構成要素

2.1 モジュラーブロックチェーンの構成例

- モノリシックブロックチェーンは、レイヤー1がすべての機能を提供するが、モジュラーブロックチェーンは複数のブロックチェーンやシステムを組み合わせて統合的に機能を提供する。
- RollupがTxの実行機能を担うほか、データ可用性をオフチェーンに切り出す構成もみられる。

機能名称(レイヤー)	機能概要	モノリシックブロックチェーン	モジュラーブロックチェーンの構成例			
			一般的なRollup	Validium	Optimium	Sovereign Rollup
Execution pp.10-11	Txを実行	レイヤー1[*1] Ethereumなど	Rollup (zkRollup/ Optimistic Rollup)	zkRollup	Optimistic Rollup	Rollup
Settlement p.12	Txの検証とブリッジを提供		レイヤー1	レイヤー1	レイヤー1	
Consensus p.13	Txの整合性を保証					レイヤー1/ DAレイヤー
Data Availability pp.14-18	検証用データへのアクセスを保証			DAレイヤー (オフチェーン)	DAレイヤー (オフチェーン)	

1. メインチェーンとも呼ばれる。なかでも、Ethereumがモジュラーブロックチェーンにおいてレイヤー1に利用されるケースが多い。

(参考)"Monolithic vs. modular blockchain". Visa. https://usa.visa.com/solutions/crypto/monolithic-vs-modular-blockchain.html, (参照 2024-04-03), "What Are Modular Blockchains?". Binance Academy. https://academy.binance.com/en/articles/what-are-modular-blockchains, (参照 2024-04-04), "Settlement in the modular stack". Celestia. https://celestia.org/learn/modular-settlement-layers/settlement-in-the-modular-stack/, (参照 2024-04-04)

2.2.1 Executionレイヤーの役割

- Executionレイヤーは、トランザクション(Tx)を実行してデータを書き換える役割をもつ。
- 代表的なレイヤー2であるRollupは、Txを処理した結果を集約してレイヤー1に送信することで、ブロックチェーンを用いたシステム全体のスケーラビリティを向上させる技術。

モジュラーブロックチェーンにおけるExecutionレイヤーは、レイヤー1の代わりにTxを実行してデータを書き換える役割をもつ。

Executionレイヤーでは、サイドチェーンやRollupが利用される。なかでも、Rollupの利用が進んでいる。

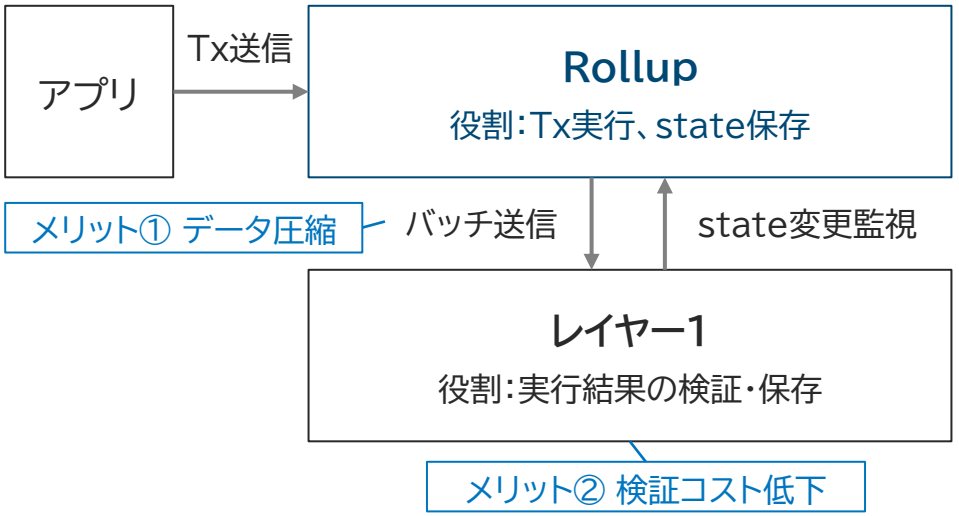
Rollupとは、レイヤー2の一種。Txの実行とデータ(state)の保存機能を提供し、レイヤー1の負荷低減を目的とする。

複数のTxをオフチェーンでまとめ、処理した結果(バッチ)をレイヤー1に送信し、合意形成を行う。

Txをまとめることでレイヤー1に保存するデータを圧縮して容量を削減する他、レイヤー1のノードが負担する検証コストが減少するメリットがある。

RollupがTxを高速に処理しつつ、レイヤー1が実行結果の完全性を保証する*1。

Rollupの構造



RollupのTVL*2(取り扱うトークンの合計価値)は、1年間で約4倍に高まり、技術開発とともに利用拡大が進んでいる。

2023年5月末		2024年5月末
約108億米ドル*3	約4倍	約424億米ドル*3

詳細は、先端技術ラボ発行レポート「パブリックブロックチェーンの技術動向」(https://www.jri.co.jp/page.jsp?id=106307)を参照。

1. レイヤー1のコンセンサスが処理結果を保証することを「レイヤー1のセキュリティを借りる/用いる」と表現することがある。
2. Total Value Locked.
3. “L2BEAT”. https://l2beat.com/scaling/summary?#active, (参照 2024-05-31)

2.2.2 代表的なRollupの方式

- Txの正当性を保証する方法の違いから、Optimistic Rollup/zkRollupに分かれて開発されている。
- Optimistic Rollupは正当性を事後確認するのに対し、zkRollupは正当性を確認したことをゼロ知識証明で保証する。

Optimistic Rollup

オフチェーンで受け取ったTxが有効であるものと仮定して計算し、レイヤー1の検証者(ノード)にバッチを送信する。

バッチを送信してから一定期間(7日間)はチャレンジ期間と呼ばれ、期間中は異議申し立てができる。

これは、検証者がFraud proof(不正証明)でTxの不正を確認した場合に行われる。不正があればTxを再度計算し、不正なTxをブロックに追加したノードにペナルティを課す。

(例)計算結果に誤りがある。ガス代が不足している。

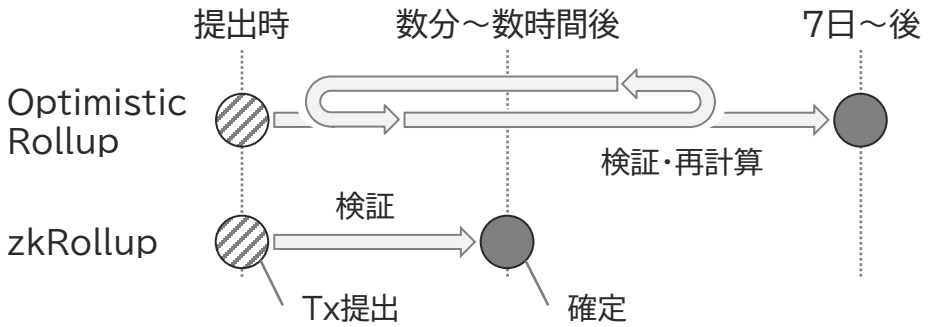
例:Arbitrum One, OP Mainnet(旧:Optimism), Base

Zero-Knowledge Rollup(zkRollup)

ゼロ知識証明を用いたValidity proof(有効性証明)によって、バッチによる状態遷移の結果と、バッチ内のTxの実行結果が一致することを保証する。

Txの有効性が保証されるため、Txの再計算が発生しない。Optimistic RollupはTxの状態が確定するまでに時間を要す一方、zkRollupは比較的短時間で確定する。

実装難易度が高く、現在はOptimistic Rollupの利用が進んでいる。



例:Linea, Starknet, zkSync Era

詳細は、先端技術ラボ発行レポート「パブリックブロックチェーンの技術動向」(https://www.jri.co.jp/page.jsp?id=106307)を参照。

(参考)" Optimistic Rollups | ethereum.org". https://ethereum.org/en/developers/docs/scaling/optimistic-rollups/, (参照 2024-05-30), "Zero-knowledge rollups | ethereum.org". https://ethereum.org/en/developers/docs/scaling/zk-rollups/, (参照 2024-05-30)

2.3 Settlementレイヤーの役割

- Settlementレイヤーは、Txを検証する役割と、Executionレイヤー同士を連携する役割をもつ。
- Settlementレイヤーの機能は他のレイヤーとともに提供され、単体として提供されることはない。スマートコントラクトが実行できるレイヤー1を利用することが多い。

Settlementレイヤーは、Executionレイヤーの実行結果が正当であるかを検証する役割をもつ。

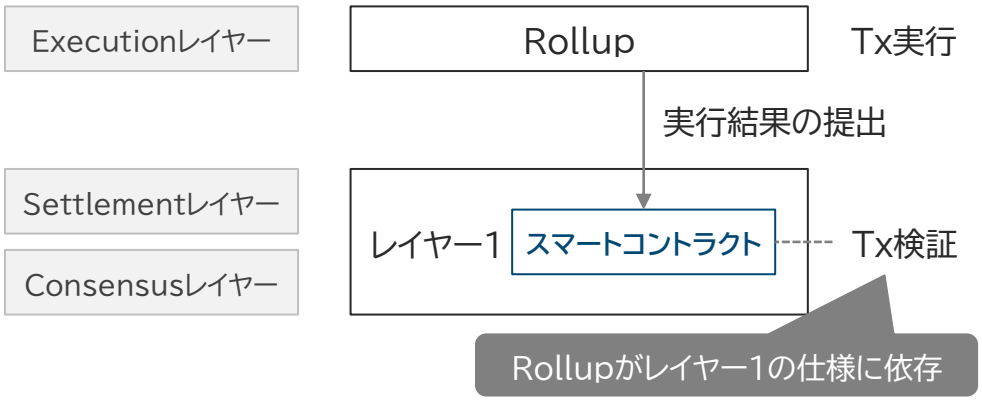
特定の用途に沿って構築したブロックチェーンを利用するAppchainの運用や、並列運用による処理性能向上を目的として、複数のExecutionレイヤーを接続する場合には、Executionレイヤー間のメッセージングを担うものと整理されている。

レイヤー1上で動くスマートコントラクトがTxの実行結果を検証するため、大抵はレイヤー1がSettlementレイヤーの役割を果たし*1、Settlementレイヤーとして個別システムを構築・提供する事例はない。

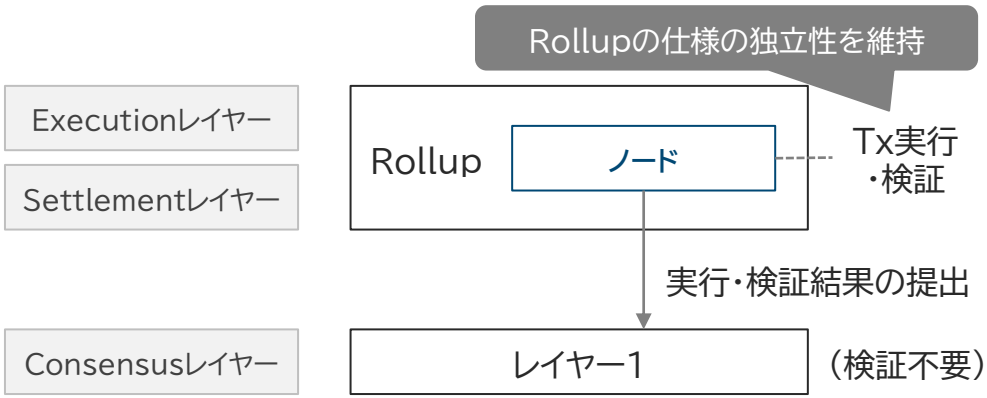
レイヤー1のスマートコントラクトに代わり、Rollup（Executionレイヤー）のノードが実行結果を検証する構成も提案されている。

レイヤー1を利用することで生じる制約をなくし、Rollupの主権性を保証できることから、Sovereign Rollupと整理されている*2。

Settlementレイヤー（レイヤー1）によるTxの検証



Sovereign RollupによるTxの検証



1. スマートコントラクトが利用できるLayer1の代表例として、Ethereumが挙げられる。
2. DAレイヤーを提供するCelestiaが提案する概念。

2.4 Consensusレイヤーの役割

- Consensusレイヤーは、ブロックチェーンに保存する内容（State）に対する合意をとり、Txの整合性を保証する役割をもつ。
- 基本的にレイヤー1の合意形成を利用することが多い。

Consensusレイヤーは、Tx自体やその実行結果に対して整合性を保証する役割をもつ。

ブロックチェーンでは、複数のノードにそれぞれのタイミングでTxが到着する。各々が異なる順番でTxを実行してしまうと、最終結果が異なってしまうため、すべてのノードで同じデータを共有し、整合性をとることができなくなる。

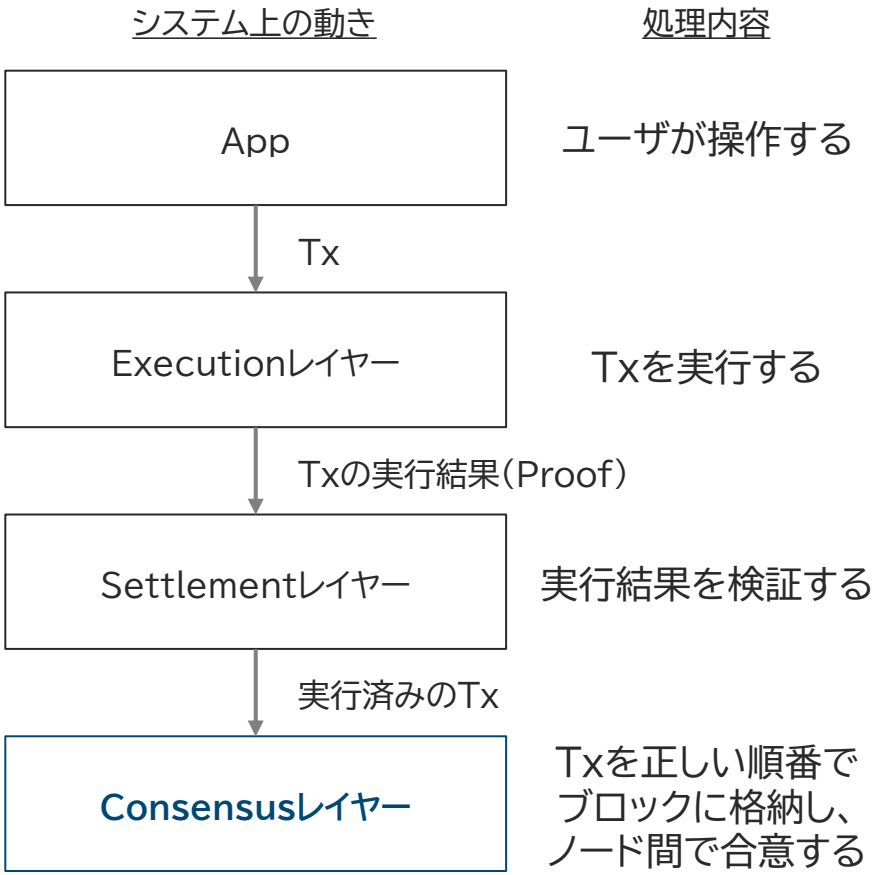
そのため、保存する内容に誤りがないことや、正しい順番でTxを実行しているかを検証し、問題がなければノード間で合意して保存するという仕組みをとっている。

多数のノードが参加するネットワークにおいて合意形成する仕組みは、従来からブロックチェーンの主たる特長の一つとして知られる。

モジュラーブロックチェーンを構成する場合も、レイヤー1がConsensusレイヤーに利用されることが多い。

Settlementレイヤーでは、RollupでTxを実行した結果の証明（Proof）をスマートコントラクトが検証する一方、Consensusレイヤーでは、Txが含まれるブロックの内容が正しいかをノードが検証する。

Txがブロックに格納されるまでの流れ



2.5.1 Data Availabilityレイヤーの役割

- Data Availabilityレイヤーは、ノードに対してデータ可用性を保証する役割をもつ。
- ブロックチェーンにおいて、データ可用性とは「検証に必要なデータを常にネットワークに公開する能力」。
- 複数のシステムからなるモジュラーブロックチェーンは、効率的にデータ可用性を保証する基盤が必要。

データ可用性(Data Availability: DA)とは

可用性という用語は、情報システムの分野では「システムが継続して稼働できる能力」、情報セキュリティの分野では、「認可された利用者が、必要なときに情報にアクセスできることを確実にすること」[*1]を指す。

一方、ブロックチェーン分野におけるデータ可用性とは、「検証に必要なデータを常にネットワークに公開する能力」。

データの整合性を常に検証できるため、ブロックチェーンは改ざん耐性をもち、データの完全性を保証する。この特性を実現するには、データ可用性の維持が重要となる。

データ可用性の実現方法の種別

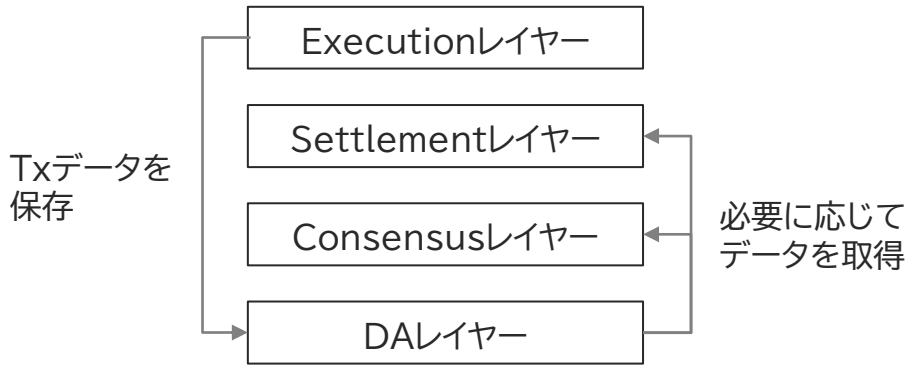
オンチェーンDA	すべてのTxデータをブロックチェーンに保存 長所: 攻撃耐性、改ざん耐性が高い 短所: コストが高く、スケールしにくい
オフチェーンDA	Txデータをブロックチェーン外に保存し、そのハッシュをブロックチェーンに保存 長所: オンチェーンと比べてコストが低い 短所: データ可用性は外部システムに依存

モジュラーブロックチェーンとデータ可用性

既存のモノリシックブロックチェーンでは、フルノード[*2]がデータを保存して整合性を検証し、データの完全性を保証してきた。

しかし、負荷が集中すると処理性能が低下するため、複数システム(レイヤー2など)との連携を高めてきた。

データの保存先が分散され、検証の効率が低下するため、検証に用いるデータをノードに公開するDAレイヤーが必要になっている。



1. "国民のためのサイバーセキュリティサイト". 総務省. https://www.soumu.go.jp/main_sosiki/cybersecurity/kokumin/glossary/ja_02/, (参照 2024-05-27)

2. ブロックチェーンのノードのうち、すべてのトランザクションデータを保存するノード。

2.5.2 データ可用性を維持する仕組み

- 高い信頼性で効率的にデータ可用性を維持する仕組みとして、Data Availability SamplingやData Availability Committeeといった手法が提案されている。
- データ冗長化の技術や信頼できるノードの集合を利用し、全データが取得できることを保証する。

Data Availability Sampling(DAS)

各ノードは、全データのうちランダムに選択されたデータの集合をダウンロードする。このとき、要求通りのデータが取得できれば、すべてのデータが利用できるとみなす手法。

Erasure Codingという技術を用いて、ダウンロードしたデータ以外のデータも取得できることを保証する。

レイヤー1はノードの複製によって冗長性を実現してきたが、DASによって効率的にデータ可用性を維持する。扱うデータが少なく性能が低いノードで検証できる。

EthereumやCosmos系ブロックチェーンを用いて実装される事例がある。

Erasure Coding(消失訂正符号)

データを冗長化し保護する手法。

欠落したデータを補えるため、複製による冗長化と比べて少ないデータ容量で実装でき、同等の耐障害性をもつとされる。

一部の分散ファイルシステム(例:Hadoop)で既に利用されている。

(参考)"Data Availability". Ethereum.org. https://ethereum.org/en/developers/docs/data-availability/, (参照 2024-05-22), "HDFS Erasure Coding". Apache Hadoop. https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HDFSErasureCoding.html, (参照 2024-05-28).

Data Availability Committee(DAC)

信頼できるノードの集合を用意し、オフチェーンでデータを保存する。ネットワークからデータを取得できない場合は、DACのノードがデータを提供するという手法。

DASと組み合わせて利用、あるいはDASを代替する。

オンチェーンのデータを取得する場合と比べてガス代効率が良いとされる一方、データ可用性の維持がDACの運営状況に依存するセキュリティリスクが指摘されている*1。

DAレイヤーを構築しているEigenLayer, StarkEx, zkPorterなどのシステムで導入がみられる。

例:StarkEx

DACメンバーはブロックチェーン関連企業が担っている。

メンバーとなっている企業*2:
Consensys/Infura/Nethermind/Cephalopod/Iqlusion/StarkWare

1. Tas, E. N., & Boneh, D. (2023, May). Cryptoeconomic security for data availability committees. In *International Conference on Financial Cryptography and Data Security* (pp. 310-326). Cham: Springer Nature Switzerland.
2. "StarkEx | An L2 Scalability Engine, Live on Ethereum Mainnet". StarkWare. https://starkware.co/starkex/, (参照 2024-05-28)

［参考］Avail｜DAレイヤー

- Availは、RollupやAppchainごとに専用IDを割り当てて効率的なデータ取得を図っている。
- 数学的性質を用いてデータの整合性・完全性の検証を切り出すことでスケーラビリティを向上させる。

◆ プロダクトの特徴

Availは、Avail DAを中心に、複数のブロックチェーンを連携したシステムを構築できるデータ可用性基盤を目指す。

現在は、公開テストネットを運用中。

◆ 技術的な特性

Ethereumをスケーリングするdankshardingの仕様に対応し、Blob形式のデータを扱う。

データ可用性を保証する機能とそれを支援する機能で構成される。

コンポーネント	機能
Avail DA	Substrateをベースに構築されたDAレイヤーで、データ可用性を保証する。 バリデータがTxデータを検証し、公開する。 RollupやAppchainに専用のappIDを割り当て、関連するTxのみを参照できる。
Avail Nexus	Avail DA上で動作するzkRollupで、Proofの集約などにより、複数のRollupを連携する。
Avail Fusion	外部発行トークンでステーキングできるようにし、NPoS[*1]におけるセキュリティを強化する。

データ可用性を保証する仕組み

Tx受信	RollupからTxデータとappIDを受信
データ冗長化	ノードはブロックデータを k 個に分割、$2k$ 個に拡張し、完全性を保証するためのproofを作成
コミットメント作成	ブロックごとにKZGコミットメント[*2]とProofを作成
合意形成とブロック伝播	バリデータはコミットメントとProofでデータの整合性を検証し、2/3以上の合意でネットワークにブロックを伝播
整合性の確認	ライトクライアント（LC）[*3]はDASを用いてデータの整合性を検証し、データ可用性を確認
完全性の確認	LCはproofを検証し、データの完全性を確認

(参考)" What is Avail? - Avail Developer Docs". https://docs.availproject.org/docs/introduction-to-avail/avail-da, (参照 2024-05-29), "GitHub - availproject/avail-light". https://github.com/availproject/avail-light/tree/main, (参照 2024-05-30), "Avail's Vision: The unification layer for web3.". https://blog.availproject.org/the-avail-vision-accelerating-the-unification-of-web3/, (参照 2024-05-30)

1. Nominated Proof of Stake. Delegated PoSと類似し、暗号資産をもつノミネータの投票でバリデータが決まる。
2. 暗号学における多項式コミットメントスキームの一つ。暗号学的に算出された公開パラメータを用いて、元データから作成した多項式（コミットメント）を表現する。入力データや公開パラメータから算出される値とコミットメントと比較すると、入力データが元データに含まれるか証明できる（包含証明）。ハッシュ木と同様に、全量より少ないデータで包含証明できる。
3. Txデータを自身に保存せず、必要に応じてフルノードに照会しながら動作するクライアント。

［参考］Celestia｜Consensus+DAレイヤー

- Celestiaは、データ冗長性を効率的に保証するため、データ拡張方法を工夫している。
- 空間を区切って管理できるハッシュツリーを採用することで、個々のRollupなどに紐づいたデータのみを取得して検証できる。

◆ プロダクトの特徴

Celestiaは、従来よりも容易にブロックチェーンを用いたシステムを構築できるよう、バリデータを自前で用意せずにデータ可用性を保証する。

Cosmos SDKを用いて構築されている。

◆ 技術的な特性

① 効率的なデータ冗長性保証

DASに必要となるデータの冗長性を確保するため、2D Reed-Solomon（RS）encoding*1という手法を用いてデータを拡張し、一部データの欠損時に復元する仕組みをもつ。

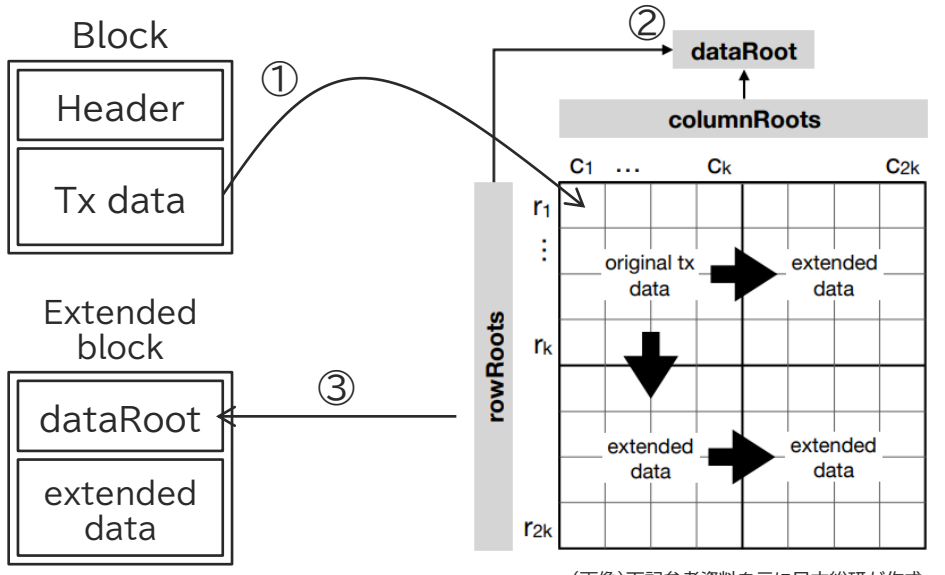
ブロック作成者による不正なデータ拡張を防ぐため、ライトノードが不正な拡張データを検知するための仕組み*2が用意されている。

② ハッシュツリーの工夫

名前空間ごとにデータを整列するNamespaced Merkle Treeという仕組みを用いてハッシュを効率的に管理する。

これによって、個々のアプリケーション（Rollupなど）において検証に必要なデータのみを取得できる。

2D Reed-Solomon（RS）
Encoded Merkle Tree Construction



（画像）下記参考資料を元に日本総研が作成。

以下の手順でデータの冗長性を保証する。

① ブロック内のTxデータを $k \times k$ に分割
② データを $2k \times 2k$ に拡張し、各行列のマークルルート（columnRoots/rowRoots）のマークルルート（dataRoot）を算出
③ dataRootと拡張したデータをブロックに格納

(参考)Al-Bassam, M., Sonnino, A., & Buterin, V. (2018). Fraud and data availability proofs: Maximising light client security and scaling blockchains with dishonest majorities. *arXiv preprint arXiv:1809.09044*. Al-Bassam, M. (2019). Lazyledger: A distributed data availability ledger with client-side smart contracts. *arXiv preprint arXiv:1905.09274*.
1. Reed-Solomon code（リード・ソロモン符号）とは、誤り訂正符号の一種。
2. Fraud Proofs of Incorrectly Generated Extended Dataと呼ぶ。全データをもつフルノードは、自身のデータと復元データを比較すれば不正な拡張データを検知できるため、Fraud proofとして公開しライトノードに検証を求める。

［参考］Eigen DA｜DAレイヤー

- Eigen DAは、データ可用性を保証するノードと、データ冗長化や整合性を検証するproofの生成を実行するサービスを分離することで、スケーラビリティを向上させている。
- ノードの署名がデータの整合性を保証するため、ノード以外は信頼する必要がない。

◆ プロダクトの特徴

EigenDAは、EigenLayer*1を構成する仕組みの一つ。Rollup上でのファイナリティが得られるまでの間、Txデータを保存・提供し、データ可用性を保証する。

2024年4月にメインネットで稼働を開始した*2。

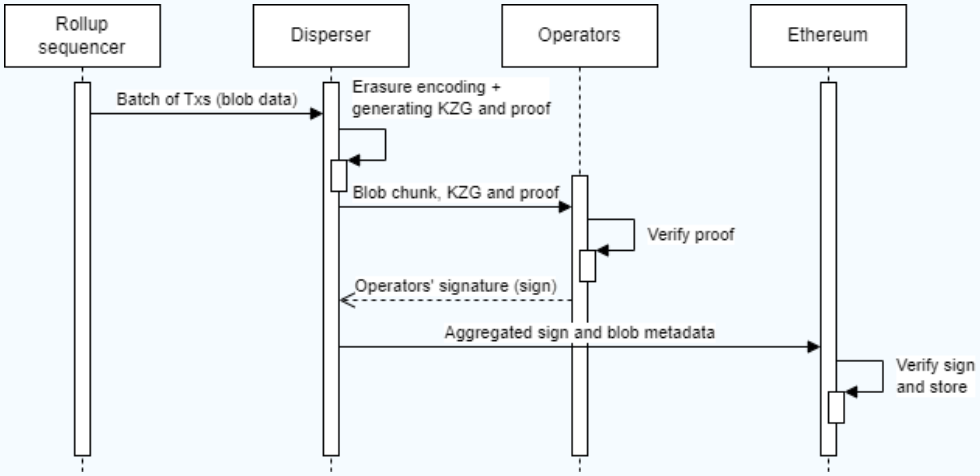
◆ 技術的な特性

EthereumをスケーリングするDankshardingを見据えた設計であり、Blob形式のデータを分割してデータ可用性を保証する。

OperatorがKZGを用いて整合性を検証し、その結果に署名するため、DisperserやRetrieverを信頼する必要はない。

コンポーネント	概要
Operator	Blob形式のデータを保存するEigenDAのノード
Disperser	EigenDAのクライアント、Operator、コントラクト間のやり取りを行うサービス
Retriever	Operatorからデータの欠片（Chunk）を受信し、正しさを検証してデータを再構成するサービス

データを検証・保存する流れ



データ受信	RollupからDisperserへデータ（Blob）を送信
データ冗長化	Disperserはデータを冗長化し、KZGコミットメントとProofを作成
包含検証	Operatorは受信したデータを検証し、データの整合性を確認できれば署名・返信
オンチェーンへの登録	Ethereumに署名とデータを送信し、保存

(参考)"EigenDA Overview". https://docs.eigenlayer.xyz/eigenda/overview, (参照 2024-06-13)
1. LST(Liquid Staking Token)を他のプロジェクトにステーク可能にするRestakingという仕組みを用いて、DAppsが利用するシステム全体のセキュリティを向上させることを目的とするプロジェクト。
2. "Mainnet Launch Announcement: EigenLayer ∞ EigenDA". https://www.blog.eigenlayer.xyz/mainnet-launch-eigenlayer-eigenda/, (参照 2024-06-10)

3. 展望・提言

3.1 今後の展望・考察

①ブロックチェーンの機能の切り出しは性能向上の有用な手段に

システムの構成要素をモジュール化し、必要に応じて組み合わせて一つのサービスを構成する手法は、ソフトウェア開発の領域でもかつてよりみられてきた。ブロックチェーンの領域では、Txの実行機能を切り出したレイヤー2という考え方が浸透している。

モジュラーブロックチェーンはこの概念の延長線上にあり、ブロックチェーンが登場した当初と比べて、複数のシステムを連携してブロックチェーンの弱点を補うという考え方に変化している。

スケーラビリティ向上のため、今後もSettlement/Consensusレイヤーを担うレイヤー1とExecutionレイヤーの連携が進むと予想される。DAレイヤーは多数のシステムが連携するようになって初めて重要になるため、社会実装に向けた開発に取り組んでいる現段階では、一般企業の利用という観点で議論が本格化しにくい。

②複数のブロックチェーンの連携を前提としたプロジェクトが増加

スケーラビリティの向上を目的に、並列する複数のブロックチェーンの相互運用、機能単位で役割分担するモジュラーブロックチェーンという概念が登場してきた。

新たなアルゴリズムを導入したブロックチェーンや、既存ブロックチェーンのコアプログラム変更は減少傾向。エンタープライズ利用でも運用実績を増やしているEthereumやAvalancheなど有力なブロックチェーンのエコシステムを中心に、周辺機能の研究・開発が進んでいる。

異なるブロックチェーンやシステムの連携を強め、一体的に利用する流れは今後も続く。

3.2 ブロックチェーンの利用に向けたエンタープライズへの提言

①ブロックチェーンが担う範囲の見極めが重要

モジュラーブロックチェーンでは、Rollupによる処理の高速化、DAレイヤーによる複数のRollupの連携、データ可用性の保証など、さまざまな恩恵が得られる。一方で、システムのモジュール化は、アーキテクチャが複雑になるほど不具合が生じた際にエラーの切り分けや対処が難しくなる。

ブロックチェーンの導入を検討する際には、ブロックチェーンとそれ以外のシステムがそれぞれ担当すべき範囲を見極めることが重要。エンタープライズでのシステム構築時、ブロックチェーンに期待する特性の一つは「データの完全性の保証」と考えられる。

より円滑なサービスを提供するため、レイヤー1を用いる際には完全性を保証することのみを期待し、データの処理・実行は外に切り出すことが現実的である。レイヤー2の一つであるRollupの利用は、ゲームや暗号資産関連サービスを中心にCoinbaseなど大手企業でも事例がみられ、RaaS[*1]の登場で導入の難易度も低下しているため、切り出す先のシステムとして優良な候補になりうる。

個別システムとしてのDAレイヤーはまだ発展途上にあり、ブロックチェーンそのものの利用が拡大した後に検討するほうがよい。

②相互接続性の低い技術の導入は避ける

ブロックチェーンが利用され始めた当初、エンタープライズではプライベートブロックチェーンの採用が多くみられた。近年は、既存の課題が解決されつつあり、パブリックブロックチェーンの利用も増えている。また、ブロックチェーンやシステム同士を連携する動きが強まっている。

主要ブロックチェーンの連携が深まることでネットワーク効果が期待できるため、相互接続性が低い技術の導入は避けるべき。

1. Rollup as a Service. Rollupの構築から稼働環境までを提供するサービス。