日本総研
The Japan Research Institute, Limited

Internet Computerの概要

～クラウドコンピューティング基盤を目指すブロックチェーン～

2022年5月30日

株式会社日本総合研究所
先端技術ラボ

＜本件に関するお問い合わせ＞ 會田 拓海（aita.takumi.m2@jri.co.jp）

序文

一般にブロックチェーンは非中央集権的な仕組みをもち、この特徴を活用することによって**Web3**（分散型Web）の実現を掲げるプロジェクトが興隆している。そのなかでも、**総合的なサービス実行基盤としてクラウドコンピューティング機能の提供**を目指して研究・開発が進められているのが『Internet Computer』と呼ばれる次世代のブロックチェーンである。

従来のブロックチェーンは取引データの保存やスマートコントラクト*1を用いたデータの記録を主な機能としている。これらの機能を利用したアプリケーションは分散型アプリ*2（Decentralized Application：DApp）と呼ばれるが、実際に分散されているのはDAppで利用するデータの一部にすぎない。DApp本体はWebアプリとして実装され、従来と同様にWebサーバー上で実行される。**ブロックチェーンはデータ処理のみを担っているため、DAppの提供には外部のクラウドサービスなどを併用するのが一般**である。しかし、クラウドサービスは大企業による寡占化が進み、運営面で分散しているとはいえない。

Internet Computerは従来のブロックチェーンの機能に加えて、より大きな容量のデータを保存でき、DAppの実行環境も提供している。暗号資産であるICPトークンを用いてInternet Computerのコンピューティングリソースを利用できる。データだけでなく実行環境も複数のノード*3に分散することで、耐改ざん性、耐障害性のある頑健なシステムを構築している。

一方、**Internet Computerで実行するDAppの動作は、一般的なWebサーバーと比べて遅く、ユーザビリティーの観点で劣る**。また、現状では導入コストがクラウドサービスより高価であること、従来のブロックチェーンと同様に法人で暗号資産を取り扱う場合に社内会計が煩雑になることなど、実運用上の課題は多く残る。しかしながら、アプリの実行環境を含めてサービス全体を分散化する試みは近年盛り上がりを見せるWeb3.0を具現化しており、将来のWebサービスの在り方を考える上でInternet Computerの動向は注目に値する。

本レポートでは、Internet Computerの概要や要素技術を概説し、簡易的な技術評価を通じた現時点のユーザーエクスペリエンスや将来展望を考察した。IT動向リサーチの一助としていただければ幸いである。

（*1）ブロックチェーンに指示を与えると、それに応じてブロックチェーン内に記録されているソースコードを呼び出し、実行する機能。
（*2）DAppはWeb3の一要素と考えられる非中央集権性、個々人によるデータ所有などの特徴をもつ。
（*3）ブロックチェーンを維持するプログラムを動かしているコンピューター。

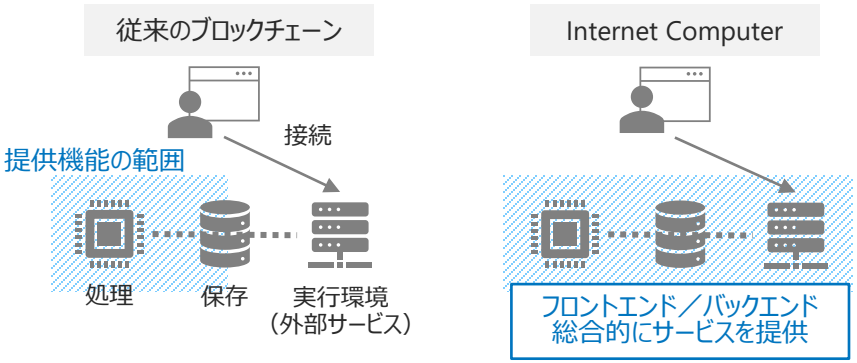
目次

日本総研
The Japan Research Institute, Limited

1.1　Internet Computerの概要

- Internet Computerはデータセンターを世界各国に分散し、**Web3.0（分散型Web）の基盤としてクラウドコンピューティング環境の提供を目的とするブロックチェーン**
- 取引データの蓄積・保存やスマートコントラクトの実行といった、従来のブロックチェーンがもつ機能に加え、**大容量データの保存やアプリケーションのホスト機能をもち、総合的にウェブサービスを提供できる**

概要



- 従来の分散型アプリ（DApp）は外部のクラウドサービスでホスト
- Internet Computerでは**データの保存だけでなく、DAppの実行環境も整えることでWeb3.0の実現を見据えている**
- アプリ開発者が分散型のインフラを設計する必要はなく、PaaS（Platform as a Service）のように扱うことができる

最近の動向

- 2021年5月、一般向けにサービス提供を開始
- SNSやゲームなどのWebアプリが試験的に展開されている
  参考：https://dfinity.org/showcase/

特徴

用途	**クラウドコンピューティング基盤**（**Webサービス提供**に特化）
規模*1 2022年4月時点	ノード提供者 55団体／ノード数 477台 アカウント数 190万以上（2021年5月～）
市場評価	**資金調達額 1.6億米ドル以上***2 米大手VC a16z, Polychain Capitalなど

ブロックチェーンの用途拡大

- 従来は取引データの保存、コントラクトによるデータ処理を提供
- Internet Computerは大容量データの保存やアプリケーション実行環境など**サービスを提供するプラットフォームとして機能を強化**

ユーザエクスペリエンス（UX）の向上

- 従来のDAppの利用には、暗号資産の保有が必要
- Internet Computerはスマートコントラクトに暗号資産を預け入れ、**従来のインターネット同様にシームレスな利用が可能**

ハードウェアレベルでの分散性の確保

- 企業提供のクラウドサービスにノードを構築できる既存ブロックチェーンと異なり、独立したハードウェアでのみ動作
- 物理的に分散・並列化し、**耐改ざん性や耐障害性を図る**

*1 https://dashboard.internetcomputer.org, https://www.icpexplorer.org/#/datacenters
*2 https://www.crunchbase.com/organization/dfinity/company_financials

日本総研
The Japan Research Institute, Limited

1.2　Internet Computerの基本構想

- 地理的に分散するデータセンターのコンピューティングリソースとストレージを統合的に利用できるプラットフォームの構築を目指す
- 業務ロジックにあたるバイトコードの管理に加えて、データの保存領域も備えるスマートコントラクトCanisterを用いたアプリケーションを提供する

ブロックチェーンが果たす役割の変化

Bitcoin	Ethereum	Internet Computer
1st Innovation	2nd Innovation	3rd Innovation
暗号資産の登場	スマートコントラクトの登場	サービス提供基盤
デジタル通貨による決済	分散型金融	ブロックチェーンの用途拡大

- Bitcoinは取引データの保存・共有、Ethereumはそれに加えてスマートコントラクトによるデータ処理機能を提供
- Internet Computerは、主流ブロックチェーンにおける**取引の処理性能やコスト、保存容量の限界に対する課題解決策**を提示
- 取引データの保存やスマートコントラクトによるデータ処理だけでなく、**ストレージ機能**や**Webアプリケーションのホスト機能**を追加し、統合的にサービスを提供する環境の実現を目指す

※第3のブロックチェーンを自称するが、取引の迅速性や拡張性を解決するプロジェクトは他にも存在し、Internet Computerは選択肢の一つ

画像：https://internetcomputer.org/

Internet Computerのアプローチ

分散したリソースを独自プロトコルで運用管理

- Internet Computer Protocolという仕組みを用いて、分散配置されたデータセンターのリソースを統合的に利用
- **数秒のレスポンスタイムでのサービス提供**
- 分散化により、システムの全停止やデータの改ざんを防ぐ

スマートコントラクトの役割の進化

- 業務ロジックを記述した**バイトコード＋データの保存領域**
- データ処理の需要に応じたスケールアウト

レイヤ	役割
Internet Computer (インターネットコンピュータ)	Canisterによるサービス提供
Internet Computer Protocol (ICP)	独自プロトコルによる分散リソースの運用管理
IP	データセンターのノード間を従来のネットワークで相互に接続
データセンター	独立したハードウェアを分散配置

日本総研
The Japan Research Institute, Limited

［参考］研究開発・運営体制

- Internet Computerの開発を主導するDFINITY財団はスイスに本部を置き、米国や上海にも研究拠点を構えている
- Internet Computerはオープンソース化されており、開発者フォーラムで仕様の議論が進む
- Internet Computer Associationはデータセンターやノード提供者の管理などを行う組織
- データセンターやノード提供者の多くがICAに加盟し、自ら管理者の役割も果たしている

DFINITY Foundation

種別	非営利組織
設立	2016年
創業者	Dominic Williams （Founder & Chief Scientist）
本部	スイス　チューリッヒ
研究拠点	• スイス　チューリッヒ • カリフォルニア　パロアルト／サンフランシスコ • 上海 （リモート拠点：独、英、米）
備考	• DFINITYでは暗号技術の研究などを行っている • AppleやGoogle、IBM、Intel、Microsoftなどの大手IT企業の元社員らが参加*1 • DFINITYが開発したInternet Computerは現在オープンソース化されている • 開発者フォーラムではInternet Computerの仕様に関する議論が行われている

Internet Computer Association

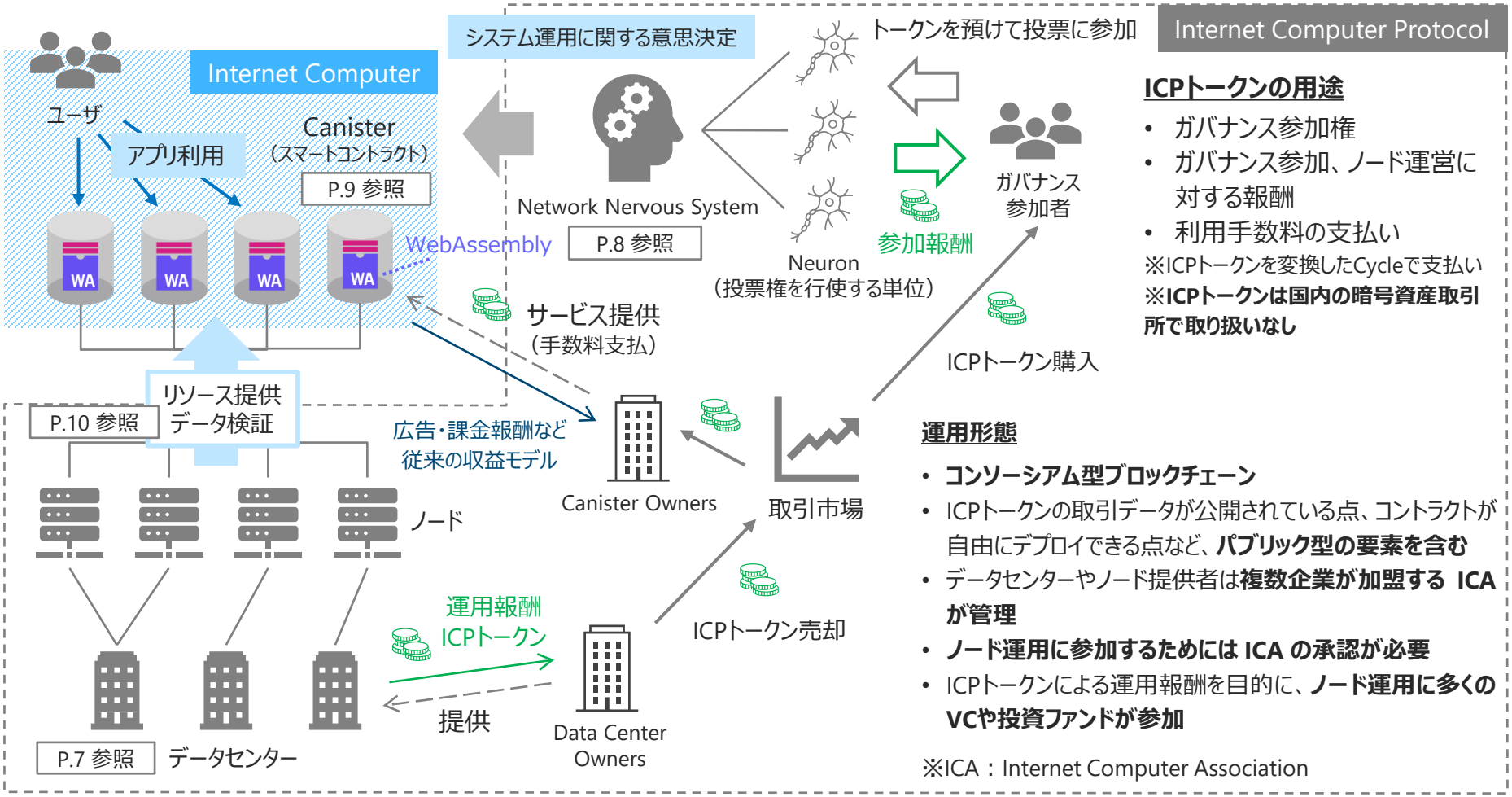
種別	Internet Computerの実質的な運営組織	
参加企業・組織	DFINITY Foundation	スイス
	9Yards Capital	米 / VC
	Electric Capital	米 / VC
	Eterna Capital	英 / VC
	KR1 plc	マン島 / VC
	Warburg Serres Investments	米 / VC
	Polychain Capital	米 / VC
	Scalar Capital	米 / ヘッジファンド
	Archery Fund	ルクセンブルク / 投資会社
	Bity SA	スイス / ブロックチェーン企業
	Bochsler Finance	スイス / デジタル資産専門の資産運用会社
	Sygnum	スイス / デジタル資産銀行
	Decentralized Entities Foundation	スイス
	Origyn Foundation	スイス

*1 参考：https://internetcomputer.org/

日本総研 The Japan Research Institute, Limited

2.1　ICP（Internet Computer Protocol）の全体像

- Internet Computerを運用・管理する仕組みをInternet Computer Protocolと呼ぶ
- 耐障害性やスケーラビリティの向上を目的とした階層アーキテクチャを構成している
- 中央集権的な管理者を置かずにシステムを持続的に運用するため、分散ガバナンスシステムをもつ



ICPトークンの用途

- ガバナンス参加権
- ガバナンス参加、ノード運営に対する報酬
- 利用手数料の支払い

※ICPトークンを変換したCycleで支払い

※ICPトークンは国内の暗号資産取引所で取り扱いなし

運用形態

- **コンソーシアム型ブロックチェーン**
- ICPトークンの取引データが公開されている点、コントラクトが自由にデプロイできる点など、**パブリック型の要素を含む**
- データセンターやノード提供者は**複数企業が加盟する ICA が管理**
- **ノード運用に参加するためには ICA の承認が必要**
- ICPトークンによる運用報酬を目的に、**ノード運用に多くのVCや投資ファンドが参加**

※ICA：Internet Computer Association

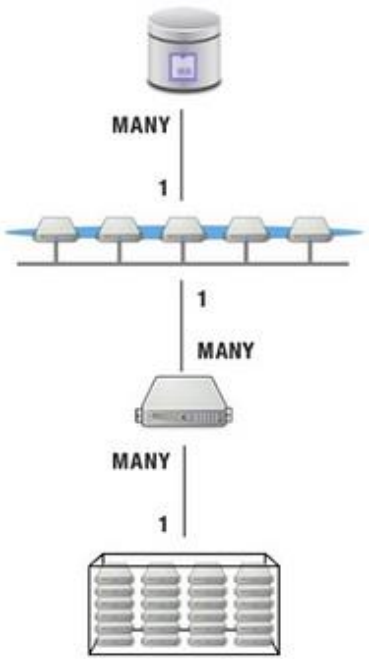
※Ledger Canisterで新規に発行したトークンをデータセンターのオーナーに対して報酬として支払う

日本総研
The Japan Research Institute, Limited

2.2　ICPのアーキテクチャ

- ICPではSubnetと呼ばれるブロックチェーンが複数存在し、分散配置されたデータセンターで動作するノードは各々異なるブロックチェーンを運用
- 異なるデータセンターに属するノード間で一つのSubnetを運用することで、システム全体の耐改ざん性や無停止運用を実現

アーキテクチャの階層構造



Canisters

現況：約4.5万
将来規模[*1]：数十億

Subnets

現況：32本のチェーンが稼働
将来規模[*1]：数十万本

ノード

現況：442台が稼働
将来規模[*1]：数百万台

データセンター

現況：米欧6か国26か所に配置
将来規模[*1]：数千か所

インターネットレベルでのサービス展開を構想し、現在ネットワークを拡大中。日本やイギリス、スペインにもデータセンターが開設される予定。

アーキテクチャの特徴

Subnetによる頑健性

- 主流のパブリック型ブロックチェーンは一本のチェーンに全取引を記録するが、Internet Computerでは複数チェーンを運用
- 異なるデータセンターに属するノード間でデータや計算結果を複製・保存し、耐改ざん性や無停止運用を実現
- BFT[*2]技術やChain Key暗号化技術（後述）を利用

（参考）
一本のブロックチェーンに全てを記録する場合、一般に処理速度の低下が課題として挙げられる。Ethereum 2.0では64本のシャードチェーンを用いてスケーラビリティの向上を計画するなどの流れが見られる。

ハードウェアレベルでの独立性

- ICPは物理的に分散したシステムの構築を目指す
- 既存のブロックチェーンは一般的にクラウド企業が提供する計算資源を利用した運用が可能
- ICPのノードはハードウェア単位で独立したコンピュータでなければ運用できない

画像：https://medium.com/dfinity/a-technical-overview-of-the-internet-computer-f57c62abc20f　*1 10年単位で整備を予定
*2 Byzantium Fault Tolerance（ビザンチン障害耐性）とは複数の情報が伝達されたとしてもただ一つの正しい合意を行うことができるという性質

日本総研
The Japan Research Institute, Limited

2.3　分散ガバナンス Network Nervous System

- Network Nervous System（NNS）は参加者の合意によってInternet Computerのガバナンスを統制
- NNSの参加者をNeuronと呼び、ICPの仕様改善案の提出やその採否に対して投票する
- ガバナンスに参加したいユーザはICPトークンをNeuronに預けることで間接的に投票に参加できる

概要

NNSの目的

- Internet Computerは複数のデータセンターのノードを連携してシステムを構築
- 特定の管理者が存在しない環境下で**複数の参加者の意思決定によってシステムを持続的に運用**[*1]

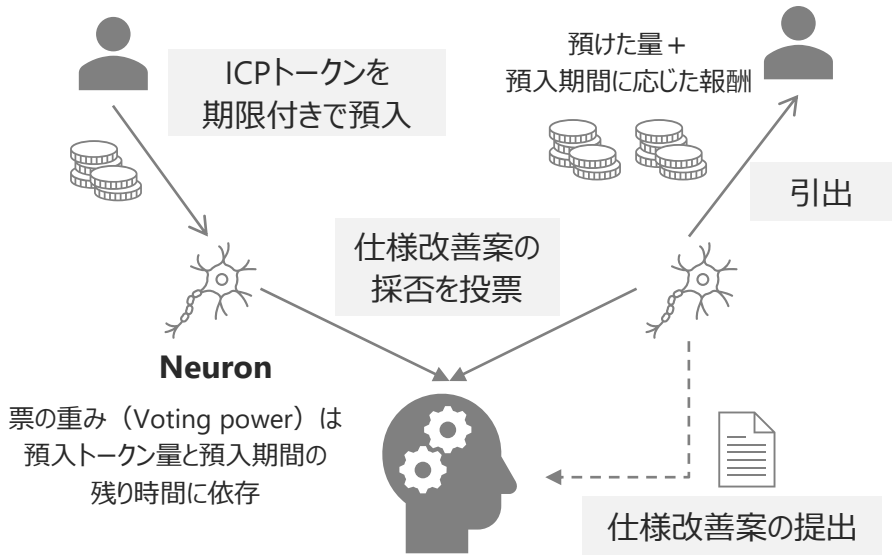
NNSの機能

- **Internet Computerの仕様改善案（Proposal）の管理**
  - ノードのOSやソフトウェアのアップグレード
  - Internet Computerの利用料金の調整
  - ノード提供者に支払う報酬の調整
  - 性能向上を目的としたノードやSubnetの追加　など
- **意思決定に参加できる参加者（Neuron）の管理**
- **各ノードが属するSubnetの管理**

※NNSの機能自体もInternet Computerに組み込まれており、Governance canisterとRegistry canisterによって提供

ガバナンスへの参加

- ユーザはICPトークンの預入（ロック）によってガバナンスに参加
- 預けたトークンが一定期間使用できない代わりに投票権を得る
- 預入期間が終了するとその期間に応じた報酬と合わせてトークンが返却される



Network Nervous System

*1 中央集権的な管理者が不在の環境で複数の参加者同士の合意を以てシステムを自律的に保つことを、ブロックチェーンの世界では自律分散組織（DAO: Decentralized Autonomous Organization）と呼称する

日本総研 The Japan Research Institute, Limited

2.4　スマートコントラクト Canister

- 従来のスマートコントラクトに相当するCanisterはコンピューティング機能に加えて、ストレージ機能を提供
- ICPトークンを変換して得られるCycleを消費することでCanisterのリソース管理を行う
- 負荷が高まったときにはCanister単位で処理を分散し、スケールアウトにて対応できる

Canisterの構成

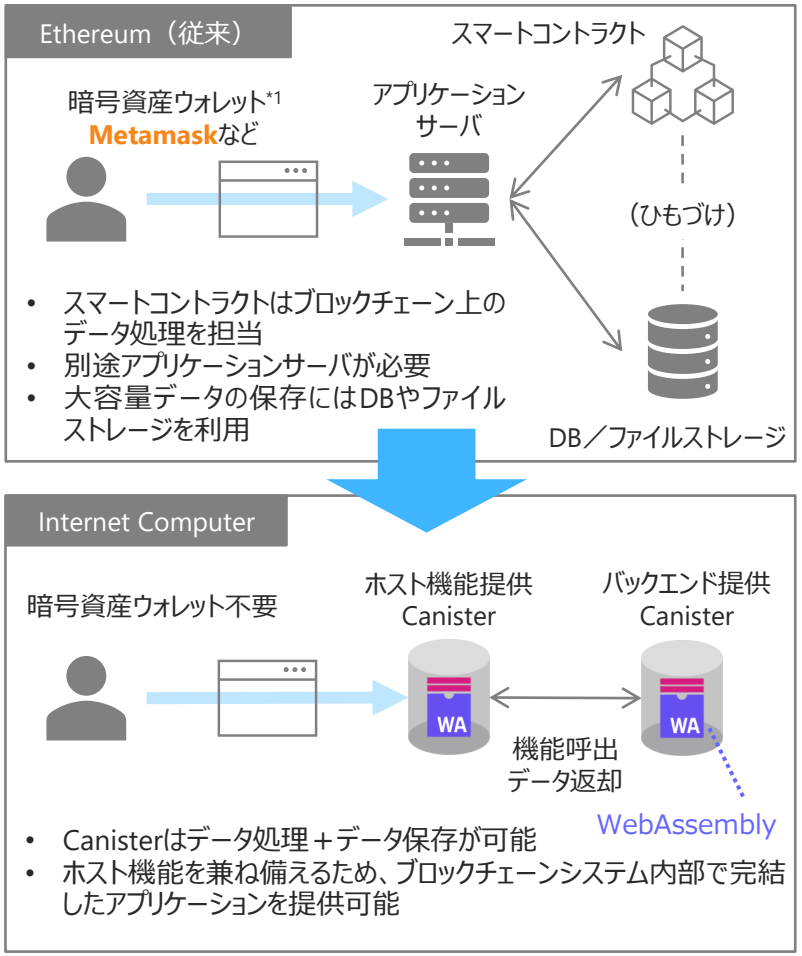
- データ処理内容の記述部（Wasm）＋データ保存領域（最大8GB*2）
- 複数のCanisterに機能分割して実装可能
- Canister IDと利用する関数を指定することで任意の処理を実行
- Webアプリケーションから接続するときはHTTPSリクエストで呼び出し
- Wasmコードはブロックチェーンに載らないため、開発者が任意に書き換えを行うリスクが想定される

Cycleを用いたリソース管理

- Ethereumにおけるガスに相当
- Canisterの処理内容によりCycleの消費量が定められている
- Canisterに予めCycleをデポジットし、ユーザの負担なくサービス提供が可能
- CycleはICPトークンをステーブルコイン*3Cycleに変換して利用

処理内容と速度の関係性

- Update call：内容変更を伴うデータ処理。同一Subnetに属するノードの2/3の賛成が必要で、非同期的にコンセンサスをとるため時間がかかる。
- Query call：内容変更を伴わないデータ取得。ノード単体のデータを参照するため同期的に高速に処理されるが、取得結果の信頼性が担保されない。



*1 画像：https://metamask.io　*2 参考：https://www.dfinitycommunity.com/internet-computer-tech-a-breakdown/#conclusion　*3 安定した価値の提供を前提に発行される暗号資産

日本総研
The Japan Research Institute, Limited

2.5　署名によるデータの保証 Chain Key Cryptography

- Chain Key Cryptographyは閾値署名を利用した技術
- 各ノードがCanisterに対するメッセージに署名することで、データの信頼性を担保する
- ネットワークに新たなノードが参加するとNNSにより自動的に鍵配送が行われ、自律的にスケールアウト

特徴

Canisterとの通信を支える技術

- Canisterの処理実行やデータ参照にはメッセージを介する
- メッセージを介した処理の頑健性、信頼性を保証するためにCanisterは複数ノードに複製されている

データの信頼性を担保する仕組み

- 各Subnetは一つの公開鍵を持ち、Subnetに属するノードはその公開鍵に対応する秘密鍵のシェア（分割した一部）を持つ
- Canisterが送受信するメッセージに対し、同一Subnetに属す複数ノードが閾値署名に参加することで、データの信頼性を担保
- Subnetの公開鍵を用いてメッセージの正しさを検証
- 公開鍵は48バイトと軽量のため、どのようなデバイスでも検証可能
  ※従来のブロックチェーンでは署名検証に多くの情報が必要

ネットワークの拡張

- ノードをSubnetに追加するときや新しいSubnetを構築するときには、NNSが鍵生成技術*1を用いて作成した秘密鍵のシェアを共有
- ネットワークが自律的にスケールアウト

閾値署名による信頼性の担保

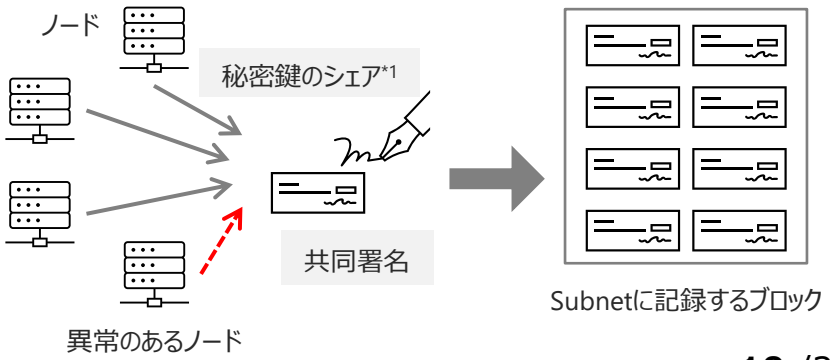
閾値署名の概要

- 署名に複数人が参加するときに用いられる方式
- 閾値以上の秘密鍵のシェアが集まれば、署名可

※秘密鍵をN個に分割したうち、K個揃えば署名用の鍵が復元可能。
同一の秘密鍵を共有しているわけではない。

閾値署名の応用

- 閾値以上のノードが参加しなければ署名できないため、単一ノードで署名するよりも信頼性が高い
- 閾値以上のノードが集まれば、その一部が正常に機能しない場合にも無停止運用が可能（動作異常、不正ノードなど）



*1 NIDKG：Non-interactive distributed key generation and key resharing https://ia.cr/2021/339
*2 N個に分割したうちK個が揃えば署名用の鍵が生成できる。同じ秘密鍵を共有しているわけではない。

日本総研 The Japan Research Institute, Limited

2.6　従来のパブリック型ブロックチェーンとの主な違い

- パブリック型ブロックチェーンは自由にノードとして参加可能だが、同じくパブリック型ブロックチェーンと銘打つInternet Computerは参加申請を出してIDを発行する必要あり
- Internet Computerではトークンの標準規格が定まっていない
- リソース使用の手数料支払いやコスト計算の面からユーザ・開発者が扱いやすい

Ethereum		Internet Computer
自由参加（許認可不要）	ノード参加	ICA (Internet Computer Association)が発行する DataCenter IDが必要（*実質コンソーシアム型）
取引データの蓄積、スマートコントラクトによる処理	機能	既存機能に加えて大容量データの保存が可能
ウェブアプリ経由でブロックチェーン参照・処理実行 （自前ノードや接続サービス“Infura”を利用してブロックチェーンとやり取り）	アプリの形態	インターネットの標準規格を統合し、ブロックチェーン上で直接サービスを展開 （HTTPで接続可能）
ユーザは暗号資産を扱うウォレットを用意 原則として利用時にユーザが手数料を負担（例外あり）	手数料・ 利用方法	事前にインスタンス（Canister）にデポジット ユーザは手数料を支払わずに利用可能（ウォレット不要）
手数料は暗号資産Etherで支払い ネットワーク負荷、Etherの価格変動の影響を強く受ける	手数料支払	ステーブルコインで支払い 価格が安定しコストが計算しやすい
一度デプロイしたコントラクトは更新不可 （異なるコントラクトとしてデプロイする必要がある）	プログラム更新・ 削除	コントラクトの更新が可能 ※コントラクトコードはブロックチェーンに載らない
EVM (Ethereum Virtual Machine)はシングルスレッドで動作	プログラム処理	非同期で並行処理可能なため処理速度向上 非同期処理の結果に齟齬が生じないように工夫が必要
金融領域が主流（ST、NFT、DeFi） トークン標準規格(ERC20, ERC721)策定済み DeFiが活発化する一方、非金融は少ない	活用の方向性	非金融領域が主流 トークン標準規格の策定はコミュニティで議論中 大容量コンテンツが扱えることからSocialFi, GameFiにも

3.1　開発・実装方法

- フロントエンドの開発は既存のWeb技術を利用可能。バックエンドの開発はWasmコンパイラが提供されている言語を用いる。開発パッケージを利用し、Canisterのデプロイや動作確認を行う。
- 多くのブロックチェーンではコントラクトのテスト環境としてパブリック／ローカルテストネットを提供しているが、Internet Computerではパブリックメインネットのみ提供。

DApp開発に必要なスキル

- CanisterはNode.js環境で開発
- フロントエンド／バックエンドはCanisterとして実装

フロントエンド（ユーザが利用するアプリケーション）
- DAppは基本的なWeb開発技術を用いて作成可
- Canisterでのホストには React.js導入が推奨される

バックエンド（業務ロジックの処理）
- 独自言語 Motoko 或いは汎用言語 Rust, Javaで記述
- Wasmへのコンパイラが提供される汎用言語であれば対応可能
- 開発パッケージDFINITY Canister SDKで動作確認、デプロイ
- ブラウザで動作確認するCandid UIもSDKに付属

開発環境
- Visual Studio Code（IDE：統合開発環境）
  - Motokoの開発環境は整備中
  - 文法ハイライト機能を公式に提供（Syntax highlight）
  - 自動整形機能の提供なし（Auto Formatter）
  - 自動補完機能の提供なし（IntelliSense）

動作環境

パブリックネットワーク
- 多くのブロックチェーンでは公開ネットワークでプログラムの動作確認を行うためのテストネットが存在
- Internet Computerでは現在テストネットの運用はなく、本番環境であるメインネットのみ利用可能

ローカル環境
- Ethereumではローカルテスト用ブロックチェーンGanacheを利用
- Internet Computerではローカルテストネットの提供なし

※ローカルノードでCanisterの動作確認は可能

名称	パブリックメインネット	ローカルノード
環境	インターネット	ローカルPC
費用	トークンが必要	不要
備考	• 接続するときは事前にウォレットの設定・登録作業が必要 • 初回は20USD分のCycleを無料付与	• PC上にCanisterを構築し、ローカルホストで起動 • Canisterの動作確認が主な機能

日本総研
The Japan Research Institute, Limited

独自言語Motoko／DFINITY Canister SDKの概要

- Canisterの実装には汎用言語Rust或いはDFINITY開発の専用言語Motokoを用いる
- アプリケーションのビルドやInternet Computerへのデプロイの機能はDFINITY Canister SDKにより提供
- SDK同梱のツールを用いてCanisterとの通信を行う

独自言語Motoko

概要

- DFINITYが開発したCanisterをサポートする専用言語
- SDK同梱のコンパイラを用いてWebAssemblyにコンパイル
- 学習環境Mokoto Playgroundを公式提供
（Motoko Playground自体もCanister上で実行されている）

特徴

- オブジェクト指向プログラミングに対応
- 静的型付けが可能
- Actorモデルを利用した非同期処理
  - ステート（オブジェクトの状態を保持する変数）と非同期処理を実行する関数をカプセル化
  - オブジェクト同士の非同期通信に対応し、リクエストの返却を自動で待機
- JavaScriptに類似する文法を持ち、習得コストが比較的低い

DFINITY Canister SDK

概要

- 主な提供機能
  - Internet Computer上のCanisterとの通信
  - 専用言語Motokoのコンパイラ
  - Canisterのローカル実行環境
- アプリケーションのビルドやデプロイなどの主要機能はdfxコマンドを利用して実行可能
- 2022年4月現在、macOS/Linuxでのみ利用可能

特徴

- SDKはコマンド実行により容易に導入可能
- ルートディレクトリに配置された設定ファイルdfx.jsonを用いてDAppの設定を管理
  - フロントエンド／バックエンドのパス
  - ローカル／パブリックネットワークのIPアドレス
  - SDKのバージョン　など

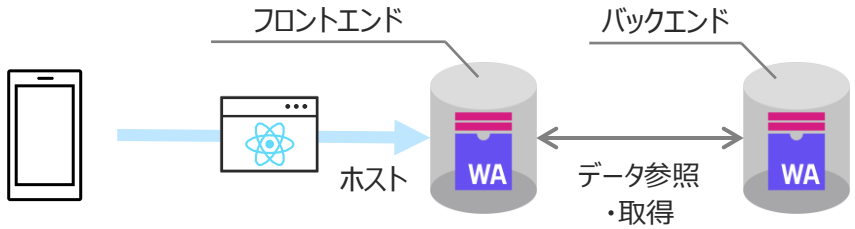
関連資料：https://smartcontracts.org/docs/language-guide/motoko.html（閲覧日：2022.4.1）

日本総研 The Japan Research Institute, Limited

検証用アプリの構成・データフローイメージ

- 検証用アプリとしてCanisterを利用するDAppと、クラウド環境にデプロイするWebアプリケーションを用意
- データの作成／読込／更新／削除を伴うユースケースとして収支を記録する家計簿を想定して実装

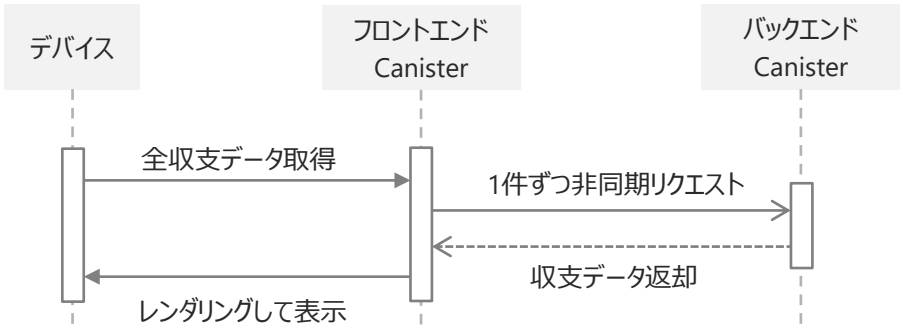
DApp構成



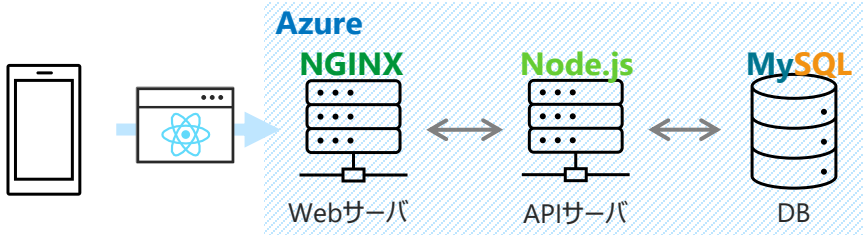
- フロントエンドを提供するCanisterでReact.jsをレンダリング
- バックエンドを提供するCanisterはフロントエンドのHTTPSリクエストに応じ、データの作成／読込／更新／削除結果を返却
- Canisterに割り当てられるFQDN*1はすべて異なるため、CORS*2による遅延が発生

［検証用アプリのソースコード］https://github.com/4ita/ic_demo
［解説記事］https://qiita.com/4ita/items/6c835e849aee5d6798b1

（トップページ接続時の処理概要）

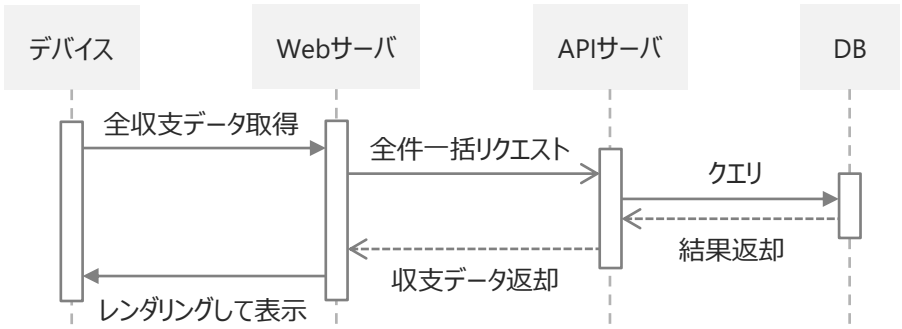


Webアプリケーション構成



- NginxでReact.jsをレンダリング
- Fetch APIを用いてExpressサーバにHTTPリクエストを送信
- ExpressからMySQLにアクセスしてデータの作成／読込／更新／削除を行い、その結果をNginxに返却
- Azure VM上の同一インスタンスにすべてのサービスを展開

［検証用Webアプリのソースコード］https://github.com/4ita/react_demo
［検証用APIサーバのソースコード］https://github.com/4ita/db_api
［サンプルアプリ］https://3cjnt-oiaaa-aaaai-qgmia-cai.ic0.app



*1 Fully Qualified Domain Name：ホスト名を含むドメイン名　*2 Cross Origin Resource Sharing：異なるFQDN間で通信を行うための仕組み

日本総研
The Japan Research Institute, Limited

3.2　技術検証の概要

- Internet Computerを利用する場合、既存のクラウド環境と比較してどの程度快適にサービスを利用できるのか、ユーザエクスペリエンスの観点から比較検証を行う
- サーバ応答やページ描画、データ処理に要する時間を実測し、Internet Computerの性能を評価する

概要

- Internet ComputerはWebアプリケーションを用いてサービスを提供する
- Internet Computerを用いて既存技術と同様のアプリケーションを提供したとき、期待するユーザエクスペリエンスを得られるか実測値を基に評価する
- Apache HTTP Serverのテストツール ApacheBench[*1]及びChrome User Experience Report[*2]をもとに検証項目を設定した

方法（構成詳細は付録を参照）

- 既存技術（Node.js, MySQL, Reactなど）を用いたWebアプリケーションをMicrosoft Azure Virtual Machine上に構築する
- 実装したCanisterをInternet Computerにデプロイし、分散型アプリケーション DApp（Webアプリケーション）を構築する
- Azure VMとInternet Computer上に構築されたアプリケーションのそれぞれについて、下記の各検証項目にしたがって実測する（測定にはApache JMeter及びChrome開発者機能を利用）

検証項目	詳細
サーバ応答	TCPやSSL/TLSの接続を確立するための初期接続の所要時間（Initial connection）
	ブラウザからHTTPリクエストが送信されてからデータ受信開始までの待機時間（TTFB: Time To First Byte）
ページ描画	サーバやブロックチェーンに接続してからページリソースやデータを取得し、DOMを描画するまでの所要時間
データ処理	DBやブロックチェーンに対し、データの作成／参照／更新／削除処理（CRUD）を実行したときの所要時間

※各項目について10回測定したときの平均を算出

*1 https://httpd.apache.org/docs/2.4/programs/ab.html　*2 GoogleがUXの評価に用いる指標 https://developers.google.com/web/tools/chrome-user-experience-report

日本総研
The Japan Research Institute, Limited

3.3　検証結果

- Internet Computerの現在の性能を処理時間で評価した場合、既存のクラウドコンピューティング環境と比較して倍以上の時間を要した
- Canisterとの通信時間の長さがアプリケーションの性能全体に強く影響している

サーバ応答時間

- TCP接続の確立やSSLネゴシエーション／ハンドシェイク*1にかかる時間（Initial connection）、接続してからデータを受け取るまでの待機時間（TTFB: Time To First Byte）が長い

ページ描画時間

- ページリソースのロード時間自体は短いが、TTFBが非常に長いためにGETリクエスト送信からリソースのロードが終了するまでに時間を要した
- 異なるドメイン間で通信を行うため、データを取得するときにPreflight requests*2が発生し、通信時間が増大する一因となっている

データ処理時間

- データの作成／更新／削除にはノードの合意を必要とするため、時間を要した
- データの読込には合意を必要としないため、比較的早く処理が終了した
- ブラウザからデータ処理のリクエストを送受信する場合はPreflight requestsが発生し、Internet Computerの性能以上の時間を要した
- 処理時間の分散が大きく、時折10秒を超えるデータ処理時間（作成／更新／削除）が生じるケースが見られた

	平均サーバ応答時間		平均ページ描画時間		平均データ処理時間*3			
	初期接続	データ取得待機	データ読込なし	データ読込あり	作成	読込	更新	削除
Internet Computer	500ms	957ms	3,196ms	4,811ms	4,124ms	1,193ms	4,186ms	4,232ms

※単位 ミリ秒 [ms]
※上記は筆者作成のDAppに対してGoogle Chromeブラウザを用いて接続した際の実測値であり、参考値として示す

*1 SSL/TLS接続を確立するための認証・共通鍵交換などの処理　*2 異なるドメイン間でリソースを共有するCORS（Cross Origin Resource Sharing）の仕組みを利用するため、リクエスト送信前に通信先の確認が行われる　*3 **測定値はPreflight requestsの影響を受けるため、Internet Computer自体のデータ処理時間は測定値より短い**と予想される

日本総研
The Japan Research Institute, Limited

3.4　技術検証の所見

- DApp開発においては従来のWeb技術を応用できる面が多いため、実装コストは低い
- 非同期実行のため、実行したノードによって結果に差が生じないようアプリケーション設計が必要
- Webアプリケーション基盤としての役割が強く、より広範な目的に利用しやすい

開発者視点

レスポンスの遅さ

- 動作の遅延はノードとの接続時間やTTFB[*1]が長いことに起因する
- レスポンス向上のためにはICPのノードに対応を求めざるを得ない

開発プロセス

- フロントエンドは広く使われているReact.jsによる開発が推奨される
- バックエンドは専用言語や一部の汎用言語を用いて開発する
- 専用言語はJS/TSの文法に類似するため、習得コストは高くない
- ドキュメントが一部未整備のため、ソースコードの参照が必要

既存技術との親和性

- 外部APIとの連携部分を書き換えれば、既存アプリにも適用できる
- デフォルトでSSL/TLSに対応し、暗号化したデータ通信が可能

実装上の留意点

- データ作成／更新／削除処理は**非同期実行のため、データ書換で齟齬が生じないようアプリ側で原子性を保証**[*2]する
- ビルド時にプログラムを32bitに最適化するため、メモリ上限が4GB

※いずれも改善案が提出されており、将来的に解消される可能性

ユーザ視点

従来のDAppとの相違点

- 暗号資産ウォレットの準備や暗号資産の保有を必要としないため利用しやすい
- Webアプリケーション提供基盤としての役割が大きく、分散型金融の機能提供はまだ少ない
- 暗号資産の取引や運用を目的とした用途から、SNSやゲームなどの一般的な用途に拡大

利用上の懸念点

- ユーザエクスペリエンス（UX）向上のため、**データ取得時間の長さを感じさせない画面遷移などの工夫**が必要
- Internet Computer上で動作させるためには常にCycleがデポジットされている必要があり、枯渇した場合にユーザが利用できない

*1 接続してからデータを受け取るまでの待機時間（TTFB: Time To First Byte）　*2 非同期でトランザクションが並行処理されるとき、データの書込結果に齟齬が生じないよう処理の順番を制御する

日本総研 The Japan Research Institute, Limited

3.5　既存クラウドソリューションとのコスト比較

- 現在主流のクラウド事業者と比較してコンピューティングコストは同等程度、ストレージコストは高い
- データの読み書き要求に対するコストは低いが、通信容量に対するコストが非常に高い
- ストレージ用途を想定しない従来のブロックチェーンと比較し、コスト優位性を保つ

- Internet ComputerはICPトークンを変換して得られるトークンCycleを消費してサービスを提供
- クラウドサービスではデータベースとストレージは異なるサービスとして展開されるが、Internet Computerではどちらも Canisterの機能として実装可能なため、今回はファイルストレージとのコストを比較
- 現状ではストレージコストの面でクラウドと競合する可能性は低いが、他のブロックチェーンと比較してコスト優位（Ethereumでデータ1GBを保存する場合、理論上は数百億米ドル*7レベルのコストが発生し、現実的ではない）

※サービスごとに課金体系は異なるため、概算値として示す

24H稼働時のランニングコスト 月額（米ドル）		Computation	Storage (per 1TiB)	Write (1M requests)	Read (1M requests)	Transfer (per 1TiB)
Google Cloud Platform Compute Engine E2*1 / Cloud Storage [in Tokyo]		62.75	23.55	5.0	0.4	122.88
Amazon Web Services EC2*1 *2 / S3*3 [in Tokyo]		39.71	25.6	4.7	0.37+α	116.74
Microsoft Azure [in Japan East]	仮想マシン D2 v3*1	94.17		0.05		79.6
	Blob Storage 汎用 v2	—	20.0	5.0	0.4	（アクセス層で異なる）
Internet Computer*4 *6		36.29*5	460.86	2.51	0.26	上り 2,800 下り 1,400

*1 vCPUs 2, メモリ8GiB構成　*2 Elastic Compute Cloudの略称　*3 Simple Storage Serviceの略称
*4 1XDR=$1.4として算出（レート：https://imf.org/external/np/fin/data/rms_sdrv.aspx, 閲覧日：2022.1.24）*5 Canisterに事前に確保されたリソース（メモリ4GiB）を100%使用した場合
*6 https://smartcontracts.org/docs/developers-guide/computation-and-storage-costs.html（閲覧日：2022.3.24）
*7（参考）https://ethereum.github.io/yellowpaper/paper.pdf

日本総研 The Japan Research Institute, Limited

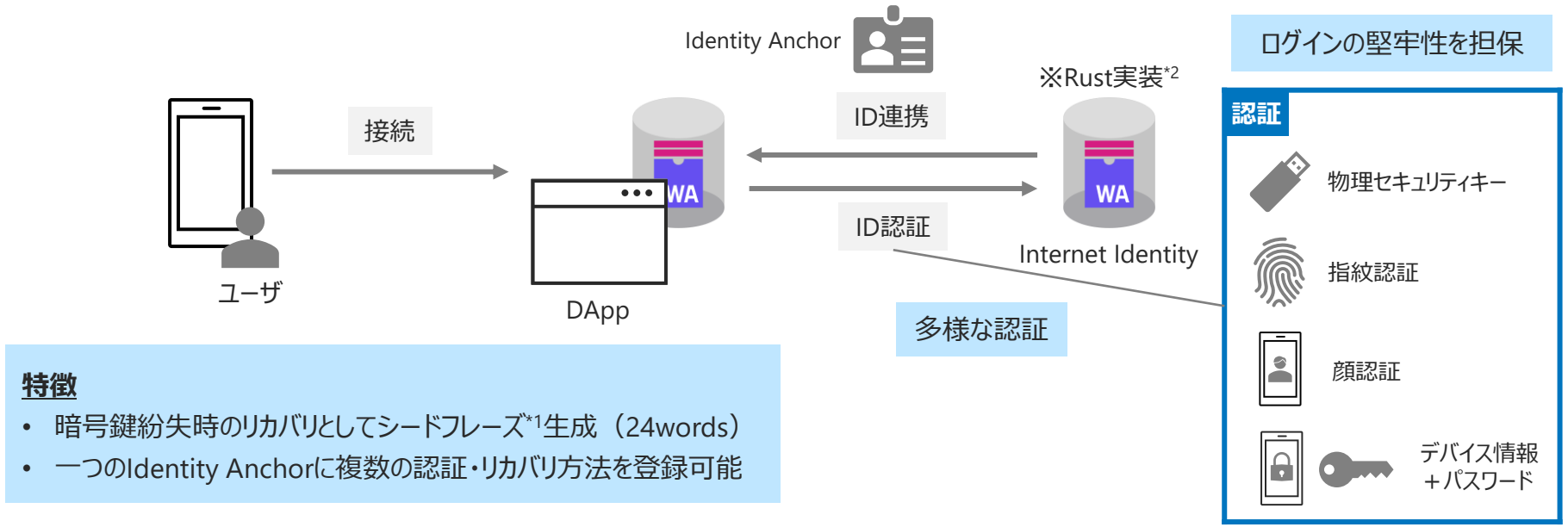
4.1　ID認証プラットフォーム – Internet Identity

URL：https://identity.ic0.app

- Internet Computer上のDApp利用に必要なID作成・認証機能を提供する
- IDはInternet Identity Anchorと呼ばれ、サービス共通のアカウントとして利用できる
- ID認証時に生体情報や物理セキュリティキーを用い、堅牢なログインシステムを構築している

概要

- Internet Computerには中央集権的に管理するユーザプリンシパル（ユーザ識別符号）が存在しない
- Internet Identityはユーザプリンシパルとして数字7桁で構成されるIdentity Anchorを発行
- Canisterで公開鍵／秘密鍵を管理し、**DApp利用時の共通認証プラットフォームとして機能**する
- **Identity Anchorにひもづく電子署名と公開鍵を用いてユーザ認証**し、Canisterに送信



特徴

- 暗号鍵紛失時のリカバリとしてシードフレーズ*1生成（24words）
- 一つのIdentity Anchorに複数の認証・リカバリ方法を登録可能

*1 秘密鍵の復元に用いる複数単語の組み合わせ　*2 DFINITYが開発。ソースコード：https://github.com/dfinity/internet-identity

日本総研
The Japan Research Institute, Limited

4.2　投稿型ソーシャルメディア – DSCVR

URL：https://dscvr.one

- Canisterのストレージ／ホスト機能を利用したWebアプリケーション
- 複数のCanister間でリクエストを送受信し、ページリソースを取得する
- CanisterでWebアプリケーションをホストする場合も、CDNなど一部のWebサービスと連携が可能

概要

- Internet Computer上のCanisterでホスティングされており、表示中のページやテキストなどのコンテンツ（UGC: User Generated Contents）もCanisterに保存
- ユーザアカウントにはIdentity Anchorを利用 P.19 参照
- ユーザの投稿記事に対し、他のユーザが自由にコメント（類似する既存サービス：Reddit）

特徴

- Canisterのホスト機能を利用する場合にも既存のWeb技術を用いて実装を行うため、従来のCDNなどサービスと連携可能（リンクで埋め込まれた画像などは外部サーバを参照）
- Internet Computer上で発行されたNFTをアイコン画像として利用するユーザが散見される
- NFTは各々異なるCanisterに保存されており、**複数のCanisterにリクエストを送信**して得られたリソースをページに表示

（参考）
- 当サービスではコミュニティに参加しているユーザに対し、トークンの発行を検討
- コミュニティへの積極的な参加などで得られるトークンは投機対象となることがあり、利益を目的としたソーシャル活動が行われるケースが存在
- 上記のように、利益獲得を目的とした利用形態やその市場を**SocialFi**と呼称することがある

日本総研
The Japan Research Institute, Limited

4.3　クラウドストレージ – IC Drive

URL：https://icdrive.co

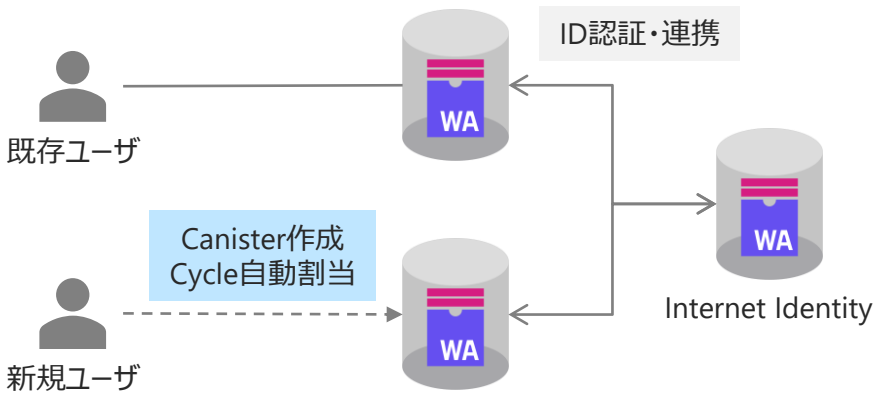
- Canisterのストレージ／ホスト機能を利用したWebアプリケーション
- ユーザの登録時に新規Canisterの生成とCycleが自動で割り当てられる
- ユーザごとCanisterが割り当てられるため、スケーラビリティに優れる

概要

- デモ版として公開されている個人向けクラウドストレージ
- Canister上のストレージにファイルを保存（上限4GB）
- Identity　Anchorをひもづけ、**ユーザが専有利用するCanisterを自動で生成**
- 利用手数料として支払う**Cycleも自動付与**

※手数料負担はCanister発行者（≠ユーザ）

（例）画像ファイル2.5MBをアップロードしたときのコスト
- 自動割当分：6,000億Cycles（$0.84）
- アップロードによる消費：約50億Cycles消費（$0.007）



特徴

- Canister間通信にCycleを消費するため、**アップロードやファイル取得に利用手数料が発生**
- 新規ユーザ登録のたびに新規Canisterを生成するため、ユーザは仮想的なリソースを占有できる
- デモ版のため、簡易実装に留まる
  - 可能な操作：アップロード／ダウンロード／ファイル削除
  - リネーム／ファイル移動／ファイルプレビューは不可

ソースコード：https://github.com/IC-Drive/ic-drive

日本総研 The Japan Research Institute, Limited

5.1　開発・サービス提供における課題

- 開発したアプリケーションをパブリックネットワークを利用してテストできない
- サービスを提供するときにWebアプリケーションのドメイン設定に強い制約がある
- 現在の法律下ではアプリケーション実行に付随する暗号資産取引に関し、課税対象の所得とみなされる可能性が高い

開発における課題

本番環境とテスト環境の分離

- Ethereumなど他ブロックチェーンと異なり、テストネットが存在しない
- ハードウェアを調達し、独自テストネットの構築は可能

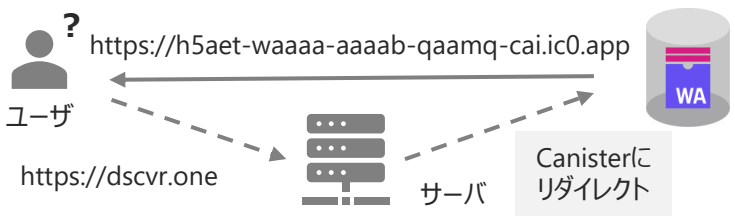
サービス提供における課題

利用手数料の高さ

- 処理内容、使用するストレージに応じて手数料が決まっている
- ICPトークンを変換したCycleで手数料を支払うため、投機的取引の影響を受けてICPトークンの価値が高騰すると、手数料も上昇
- 将来的にノードが増加し、処理内容やストレージ当たりのコストが低下すれば手数料の低減が期待できる

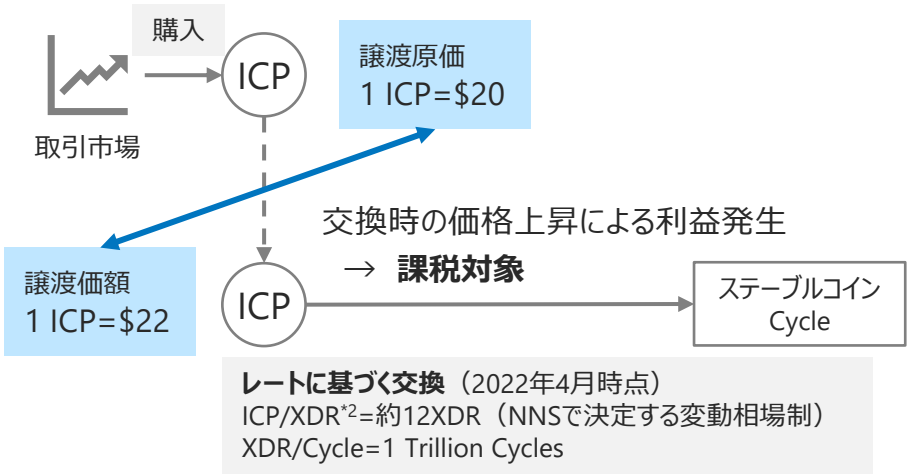
Webアプリケーションのドメイン設定

- ドメイン名やホスト名を任意に設定できず、スパムと誤認する恐れ
- 現在は初回接続時のみCanisterのURLへリダイレクト
- **将来的にInternet Computer向けDNSが提供される予定**



意図せずに発生した利益に関する税務上の取り扱い

- 市場取引されるICPトークンをステーブルコインCycleに変換
- 利用手数料として、CanisterにCycleをデポジットする必要がある
- 国税庁は**暗号資産同士の交換するとき、譲渡価額と譲渡原価の差額を所得金額と定める**
- 企業利用の場合、**約定した事業年度において譲渡損益の計上が求められる可能性**があり、日本企業で導入するハードルは高い
- 個人利用の場合、暗号資産取引により生じた利益は所得税の課税対象となり、原則雑所得に区分



*1 出典：「暗号資産に関する税務上の取扱いについて(情報)」（国税庁 令和3年12月）　*2 XDR（SDR: Special Drawing Rights）：特別引出権（国際通貨基金創設の国際準備資産）

日本総研
The Japan Research Institute, Limited

5.2　今後の展望

- Internet Computerは今後も主要機能の追加が予定され、DFINITYやコミュニティを中心に議論や実装が進む
- 技術面では既存ブロックチェーンやWebサービスとの連携により、**DApp提供基盤としての役割を強化する**
- ビジネス面ではクラウドサービスなど**既存技術を即時に代替する可能性は低い**が、**BitcoinやEthereumブロックチェーン向けのトランザクションを直接生成する機能などが実装されることで利用拡大の可能性あり**

技術的展望

Bitcoin/Ethereumとの相互運用性

- CanisterでBitcoinの残高管理やTX[*1]送受信を行う
- Internet Computer上でEthereumコントラクトを実行する[*2]
- **従来のブロックチェーン間の処理や資産移動は第三者を仲介するため、仲介者を信頼する必要**がある
- **CanisterがTX生成機能やコントラクト実行機能をもつことで仲介者が不要になる**

※ただし、**Canisterのコード監査は従来通り求められる**

Service Nervous Systemの提供

- トークンを用いたサービスの自律制御（トークンガバナンス）
- ※トークン保有者の投票でサービスの仕様変更などを決定し、**中央集権的な管理者を置かない持続的なサービス提供の実現**を目的とする

ビジネス的展望

パブリックブロックチェーンの利用拡大

- ICPトークンの市場規模（流通総額）は約50億米ドルで暗号資産市場36位（2022.3.31現在）
- 最大の市場規模を持つBitcoinやDeFiプラットフォームとして地位を確立するEthereumとの連携機能の実装により、**パブリックブロックチェーン活用において利用拡大の可能性**

既存技術の代替可能性

- 従来のクラウドサービスと比べ、**利用コストは依然として高いため短期間で代替される可能性は低い**
- 分散型Web（Web3.0）を実現するプラットフォームとして引き続き注目すべき

2022 Q1	2022 Q2	2022 Q3	2022 Q4	2023+
• Bitcoinシステムの統合 • 閾値署名のECDSA導入 • CanisterからHTTPリクエスト送信 • カスタムサブドメイン	• Service Nervous Systemの構築	• Ethereumシステムの統合 • Canisterの連携強化	• 他ブロックチェーンとの統合 • 認証局、DNSの分散 • ストレージ用Subnet構築	• ポスト量子暗号導入 • Canisterに対する秘密計算の導入（MPC[*2]）

※プロポーザルの一部を抜粋。具体的検討が進む案はNNSに提出され、Proposal IDが付与されている。青字は開発段階、黒字は議論段階。

*1 トランザクションの略　*2 Ethereumのスマートコントラクト実行環境（EVM：Ethereum Virtual Machine）の実装を計画している
*3 Multi-Party Computation：複数主体が連携してデータを暗号化したまま処理する技術の一つ

日本総研
The Japan Research Institute, Limited

5.3　まとめ

Internet Computerは、DFINITY Foundationが開発を主導する、**分散コンピューティングプラットフォームの実現を目的としたコンソーシアム型ブロックチェーン**である。従来のブロックチェーンにみられるトランザクションデータの保存機能やプログラムの実行機能に加えて、**大容量のデータを保存するストレージ機能とアプリケーションを動作させるホスト機能**をもつ。

現在主流のブロックチェーンは、データの読み書きに時間を要する点や、Webアプリケーションを外部サーバでホスティングする必要がある点など、実運用上の課題や制限が多い。Internet Computerは、従来のWebシステムと同等に、**数秒のレスポンスタイムで処理が可能なフロントエンド／バックエンドのサービスを提供する目標**を掲げている。この実現のため、Internet Computer Protocolと呼ばれる独自アーキテクチャを構築している。

Internet Computer上でWebアプリケーションを試作した結果、Webアプリケーションをホスティングできる環境が整っていることを確認した。処理時間は、一般的なWebアプリケーションと比べて、**ページ遷移は約2.2倍、データ処理は約4.8～16.5倍となり、ユーザエクスペリエンスは劣化**した。

Internet Computerは今後、BitcoinやEthereumとの連携機能を追加する計画がある。これが実現すると、Internet Computer上で**BitcoinやEthereumのトランザクションを発行できるなど、ブロックチェーンの相互運用性が高まり、パブリックブロックチェーンの用途拡大**につながる。