大規模な都市データを支える技術と視点

テクノロジードリブンなシステム開発

CEO of Eukarya
Kenya Tamura

本日の発表内容

1. 会社紹介
2. 都市データの現状と課題
3. 最新Web技術の動向
4. 最新Web技術のデジタルツインへの応用
5. 最新Web技術の積極的な活用による副次的効果
6. 最新Web技術の積極的な活用に重要な視点

Eukarya

会社紹介

Eukarya

Speaker

Kenya Tamura

1989年広島県出身。専門は地理学・GIS。

東京大学大学院在籍中にデータベースや可視化ツールの開発を行う企業を設立し、次世代データベース「APLLO」開発で国内クラウドファンディング最高額を達成。WebGISシステム「Re:Earth（リアース）」は、国土交通省が進めるデジタルツインプラットフォームであるPLATEAU（プラトー）の可視化基盤にも採用。

学歴
2012年　奈良大学文学部地理学科卒業
2014年　奈良大学大学院文学研究科地理学専攻修士課程修了
2015年　名古屋大学大学院環境学研究科社会環境学専攻研究生
2018年　東京都立大学大学院ｼｽﾃﾑﾃﾞｻﾞｲﾝ研究科ｲﾝﾀﾞｽﾄﾘｱﾙｱｰﾄ学域博士課程単位取得満期退学
2023年　東京大学大学院学際情報学府文化・人間情報学コース博士課程単位取得満期退学

現職
株式会社Eukarya 代表取締役CEO
東京大学大学院情報学環客員研究員
一般社団法人 Welcome Japan 理事
一般社団法人 MIEF 監事

Eukarya

About Eukarya

「すべての情報を保存することと、保存したい世の中にする」を価値観に、2017年に東京大学渡邉英徳研究室のメンバーによって創業した研究開発型スタートアップである。
次世代データベース「apllodb」の研究開発しつつ、その成果を活かした大規模かつ複雑な都市データを扱うことができるWebGISプラットフォーム「Re:Earth」を開発している。
2021年からサービス提供をはじめた「Re:Earth」は、国土交通省の「Project PLATEAU」に採用され、約130都市のデータを管理・運用し、自治体での普及や市民・民間企業での利用促進を目指している。

Webですべての業務が完結するGISプラットフォーム

従来のWebGISが抱える問題に対し、最新のWeb技術を用いて、**大規模化する都市データの高負荷処理が軽快に実行でき**、汎用的かつ柔軟なシステムの開発が可能になってきている。

R:Earth

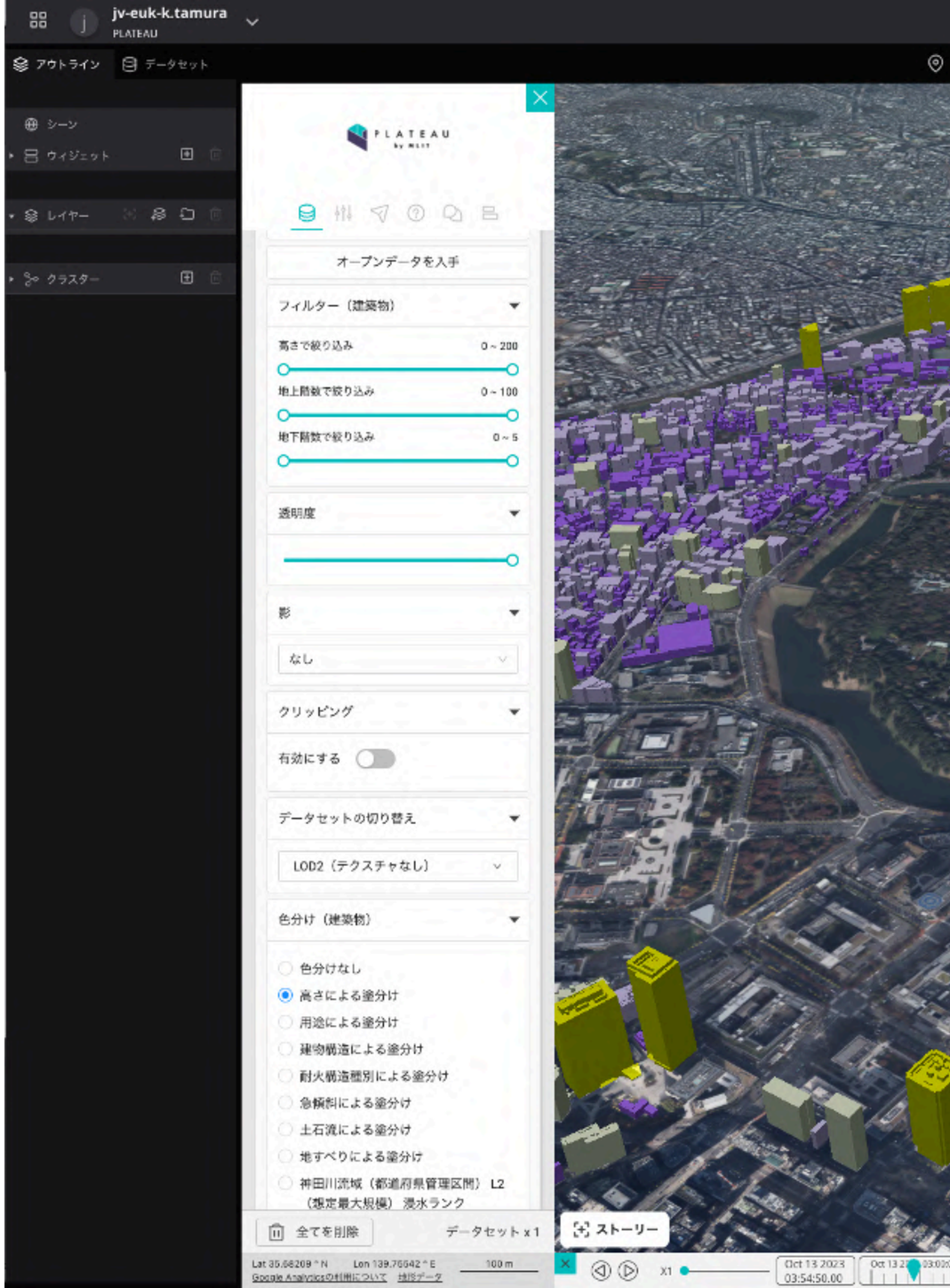
Eukarya

Re:Earth Visualizer

ノーコード可視化ツール

- ノーコードでデータビューアを作成・編集
- プラグイン機能により、柔軟なカスタマイズ性能
- クリックひとつで、Webアプリケーションとして公開

最新Web技術である「WebAssembly」を採用し、世界で初めてWebGISにプラグインシステムを実装した。それにより、ユーザー自身による機能拡張が可能になり、高品質かつ高速なWebアプリを、低コストで開発し、素早く公開することが可能になった。

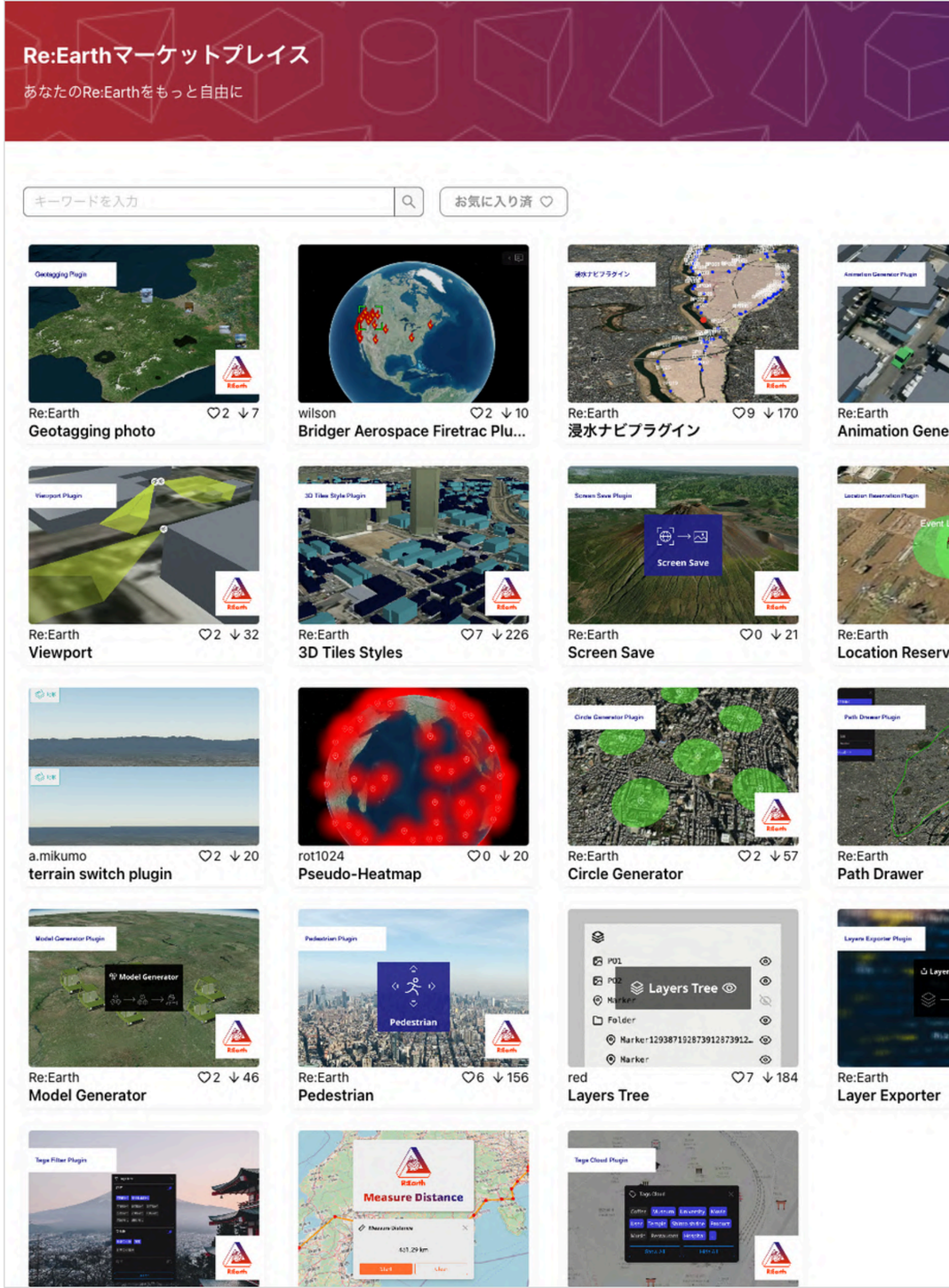


Re:Earth Marketplace

プラグイン共有ツール

- エンジニアが開発したプラグインをアップロード
- ノンエンジニアがプラグインによって自由に機能を拡張
- 同じ機能を開発しなくて良いので、コストカットに貢献

Re:Earth Marketplaceは、世界ではじめてWebGISに実装したプラグインシステムを、エコシステムとしてプラグイン共有を可能にした。将来的には、課金システムも導入し、ユーザーのプラグイン開発のインセンティブを高める。

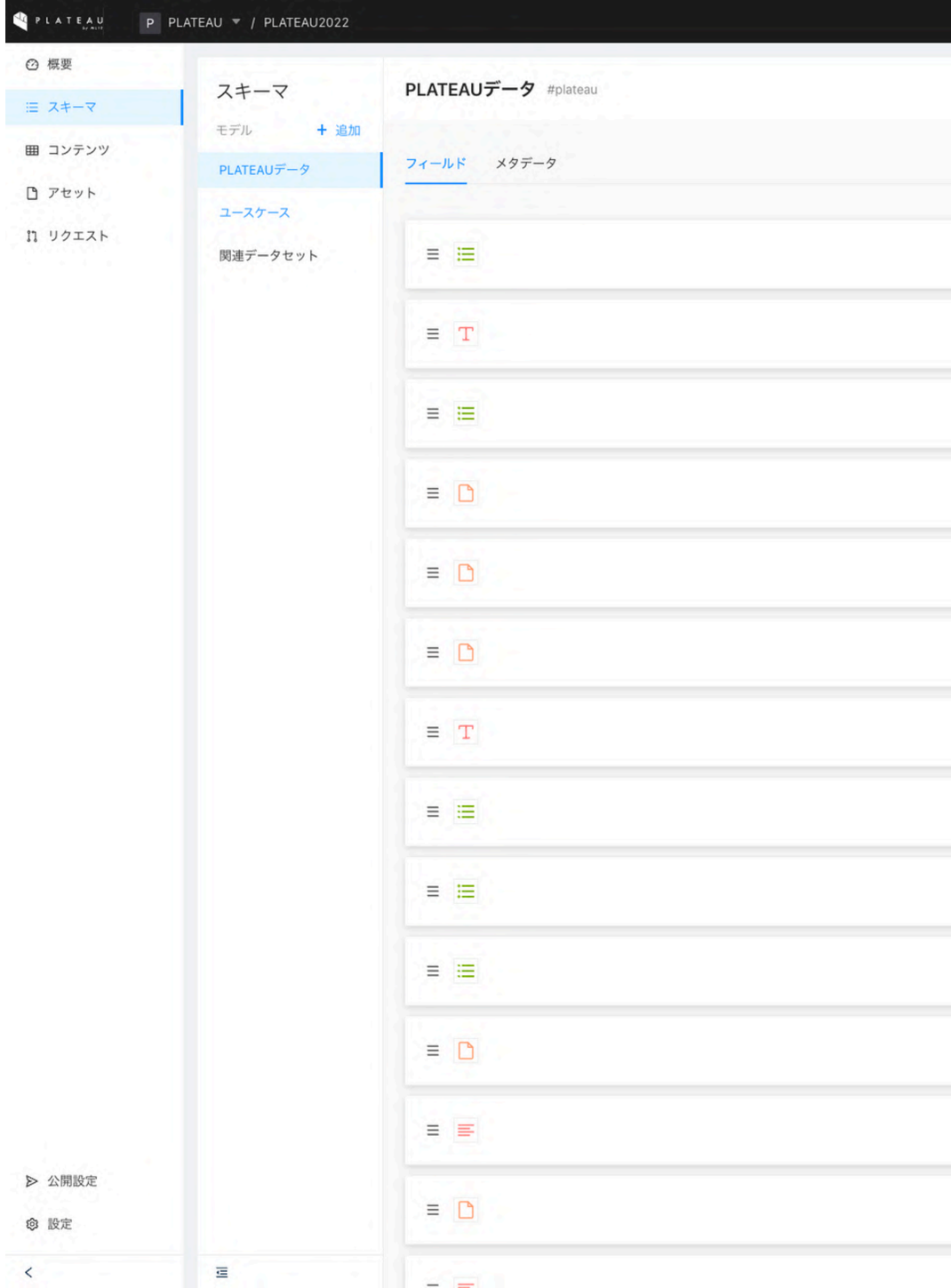


Re:Earth CMS

ノーコードデータ管理・配信ツール

- ノーコードでデータ登録・管理・API配信
- Gitのようなバージョン管理
- APIインテグレーションによって、業務の自動化

Eukaryaのデータベース研究開発の知見を活かし、大規模かつ複雑化する都市データを効率よく管理・配信する「ヘッドレスCMS」を開発した。他のWebGISの競合製品に比べて、データ管理やデータ配信の柔軟性が高く、外部アプリケーションとの高度な連携も可能にした。
現在、全ての3D都市モデルは、Re:Earth CMSによって管理・配信されている。

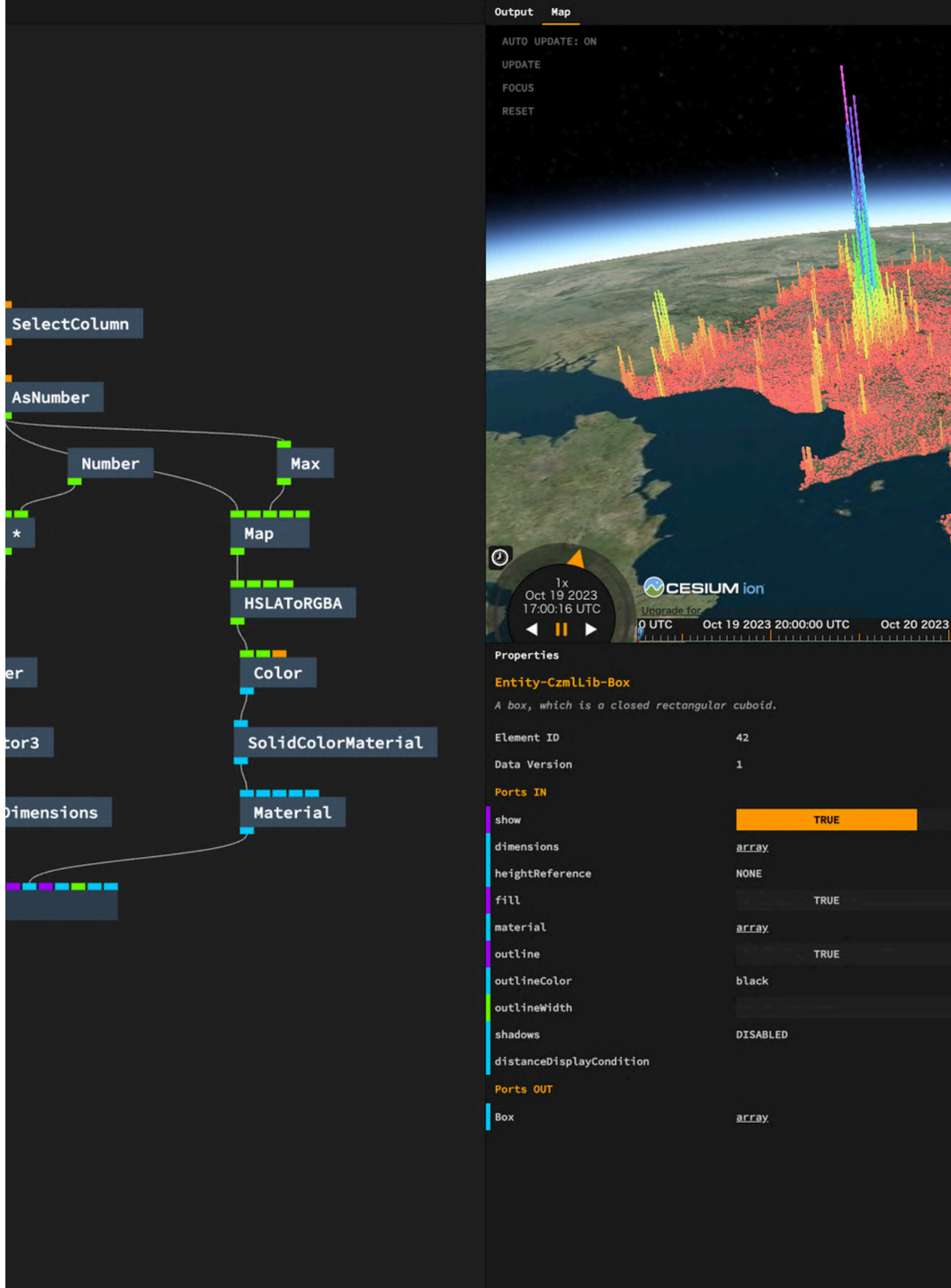


Re:Earth Flow （開発中）

ノーコードデータ変換・解析ツール

- ノーコードでデータ変換・解析
- Re:Earth Visualizer/CMSと連携
- WebGISにバックエンドプラグインシステムを構築予定

Re:Earth Visualizer/CMSと連携し、データ変換や解析をノードベースもしくはブロックで操作ができるWebGISツールを開発する。既にWebGISにデータ変換や解析はあったが、簡易な処理しかできなかった。Eukaryaの持つWeb技術によって、これまでWebで実現不可能だったGIS処理を、Re:Earth Flowで実現する。



Re:Earth AI （開発中）

ノーコードデータ生成・構造化ツール

- **生成AIを用いて、非構造データから構造データを生成**
- **Word・Excel・PDFなどの資料をCSVやJSONに変換**
- **アドレスマッチングや空間結合・テーブル結合などの処理**

生成AIを用いて、Word・Excel・PDFなどの「非構造」な資料を分析や可視化を可能にする機械判読可能な「構造的」なデータを生成するシステム。
Re:Earth AIは、データ化作業を大幅に効率化し、EBPMやBPRなど組織の意思決定プロセスをより迅速かつ効果的にサポートすることができます。

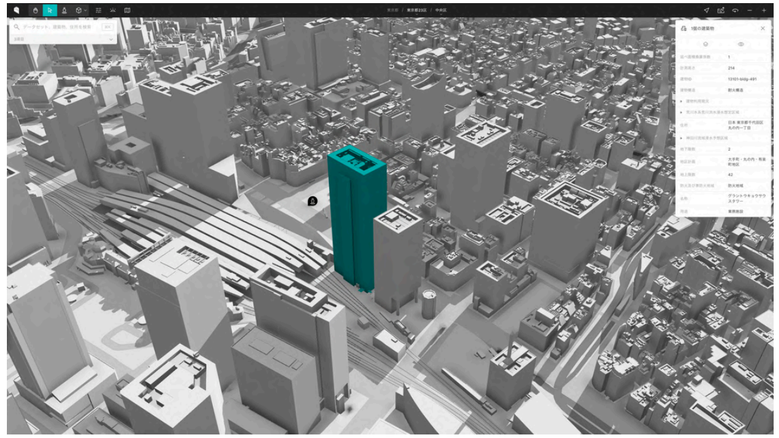
次世代マップエンジン （開発中）

ヘッドレスな次世代マップエンジン

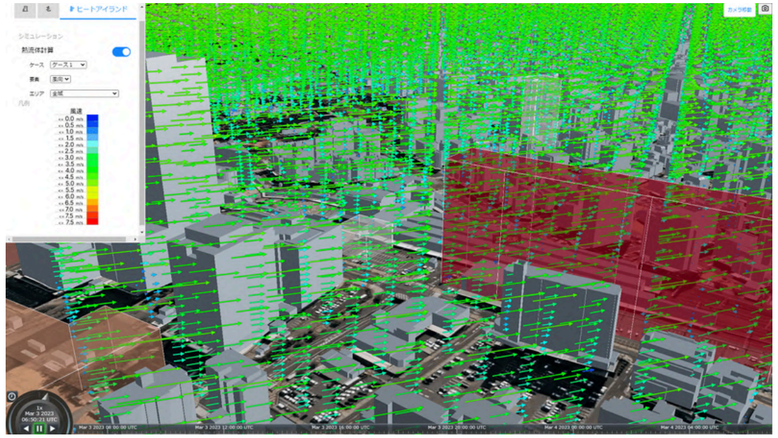
- **高速なGISデータ処理能力**
- **高品質なレンダリングによるマップ表示**
- **マルチプラットフォーム対応**

国土交通省中小企業イノベーション創出推進事業（ＳＢＩＲフェーズ３）に採択され、5ヶ年計画で開発を開始している。現在のマップエンジンの技術的な制約を、全く新しい「ヘッドレスなマップエンジン」という設計によって、GISデータ処理エンジンとレンダリングエンジンが独立しており、それぞれの領域でのパフォーマンスを最大限に引き出せる理想のWebGISエンジンを開発する。

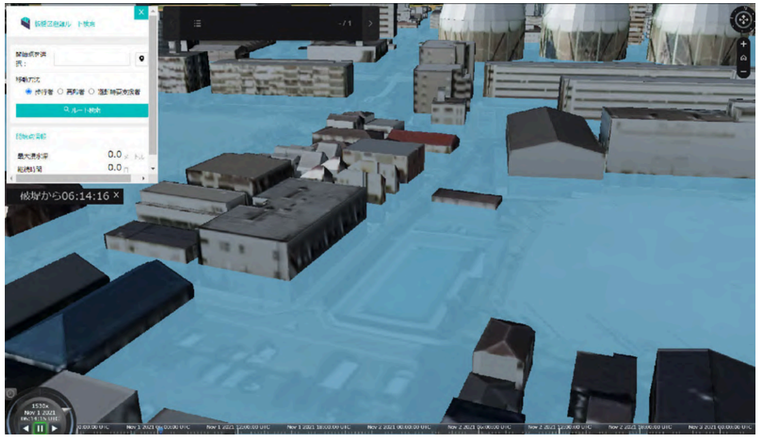
Re:Earthのユースケース



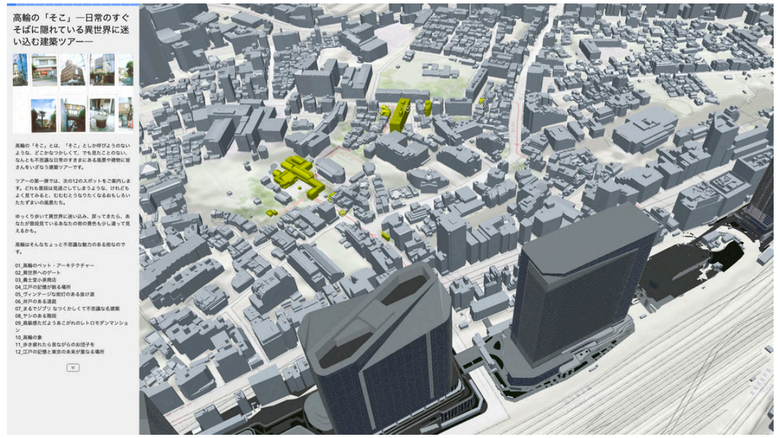
PLATEAU VIEW



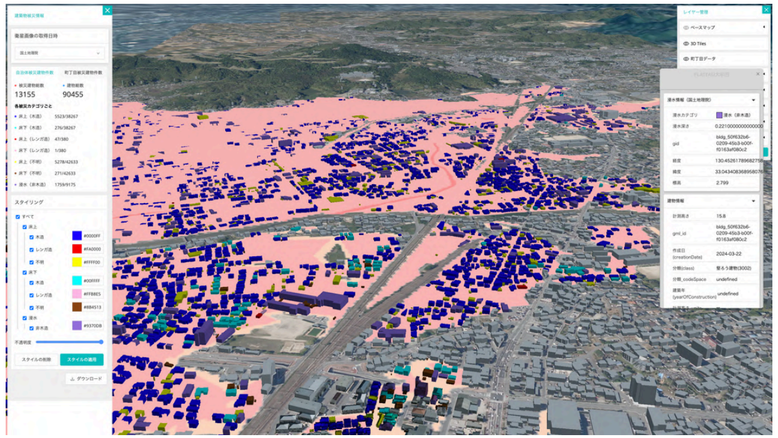
リアルタイムヒートアイランド解析(環境)



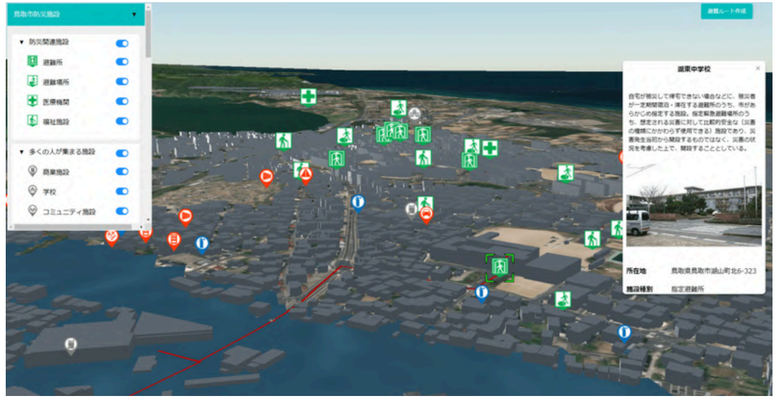
時系列浸水避難シミュレーション(防災)



スクロール型ストーリーテリング(まちづくり)



罹災証明発行支援システム(防災)



地域防災支援プラグイン(防災)

都市データの現状と課題

Eukarya

都市データの膨大なデータ量と多様性

データの規模

東京都のような大都市では、日々膨大な量のデータが生成されている。地理情報、人口、交通、エネルギー、環境など、異なる分野で多様なデータが収集されており、これらを効果的に管理することが求められている。

例: 東京都3D地図のデータや、センサーデータ（気象、交通流量など）

データの複雑性

多種多様なデータフォーマットや、リアルタイムでの更新が必要なデータなど、取り扱いが複雑であるため、これらを一貫して管理するのが困難。

- **1つのファイルで500GB以上**
- **3次元化される地図データ**
- **求められるリアルタイム性**
- **BIM/CIM/GISで異なるデータ形式**

Eukarya

ベンダーロックインの問題

古い技術遺産

都市データ管理のシステムやインフラは、過去に導入されたベンダー製品に依存している場合が多く、これが技術更新を妨げる要因となっている。特定のベンダーに依存したデータフォーマットやシステムが存在すると、新しい技術を導入する際に互換性や移行コストが大きな障害となる。

- **影響**

  新しい技術やオープンソースソリューションの導入が難しくなり、結果としてシステムの老朽化やメンテナンスコストが増大する。
- **コストの増大**

  古い技術を維持するために、ベンダーに支払うライセンス料やサポートコストが増加し、結果として自治体や都市運営の予算を圧迫する要因となる。**同時に民間企業も維持・管理、研究開発費のコスト増大の問題を抱えている。**

Eukarya

データが一部の専門家にしか扱えない

技術的な壁

都市データの一部は高度に専門的であり、特定のGISツールや特殊なフォーマットを使う必要があるため、専門知識がないと操作できない。結果として、都市データの管理や運用が限られた技術者や専門家に依存してしまう。

- **影響**

  これにより、データを効果的に活用したい場合でも、技術的スキルがないとアクセスできないため、プロジェクトが生まれない、生まれても進行が遅れることがある。また、専門家不足が運用上のボトルネックになることも。
- **コストの増大**

  古いGISソフトウェアや特定のベンダー製品に依存していると、誰もが簡単に操作できないため、データ活用が停滞するリスクが高まる。

Eukarya

結局のところ、、、

レガシーシステム脱却・システムモダン化

レガシーシステムの定義とシステム刷新（再掲）

レガシーシステムに陥る主要因

システム観点

①技術面の老朽化

古い要素技術やパッケージでシステムが構成されており、H/W等が故障すると代替がきかない状況。または、古い要素技術に対応できる技術者の確保が難しい状況。

②システムの肥大化・複雑化

システムが複雑で昨日の追加・変更が困難となり、現行業務の遂行や改善に支障がある状況。システムの変更が難しい耐え、外部に補完機能が増えたり、人が運用をカバーしなくてはいけない状況。

③ブラックボックス化

ドキュメントなどが整備されておらず、属人的な運用・保守状態にあり、障害が発生しても原因がすぐにわからない状況。または、再構築のために現行システムの仕様が再現できない状況。

経営観点

① 障害が発生しても対応できない（ブラックボックス化）
- 実態がわからないので何か起きても対応できない

② 投資されていない
- 経営者が投資する気がないシステムは、問題が起きても応急措置のパッチを当てるだけなどのその場しのぎの対応になり、その結果複雑化してしまう

③ 古い制度やしがらみに縛られている
- 昔ながらの古い業務プロセス（ハンコなど）や制度のままのシステム
（システムだけモダン化しても意味がない）

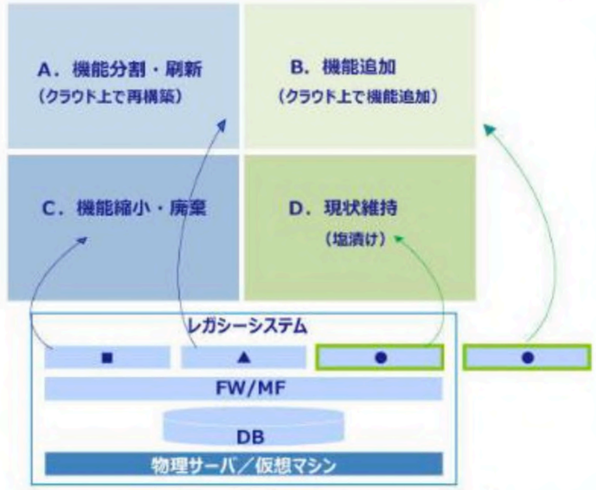
＝可視化

＝経営層からシステムへのエンゲージメント

＝現場の業務至上主義

出典）DXレポート ～ITシステム「2025年の崖」の克服とDXの本格的な展開～（簡易版）

レガシーシステム解消のアプローチ



機能毎に四象限で評価し、システム再構築を計画
- A：頻繁に変更が発生する機能はクラウド上で再構築
- B：変更されたり、新たに必要な機能は適宜クラウドへ追加
- C：肥大化したシステムの中に不要な機能があれば廃棄
- D：あまり更新が発生しない機能は塩漬け

経済産業省 商務情報政策局 情報産業課「レガシーシステムモダン化委員会について」
https://www.ipa.go.jp/disc/committee/begoj90000002xuk-att/legacy-system-modernization-comittee-20240912-overview.pdf

Eukarya

最新Web技術の動向

Eukarya

都市データの膨大なデータ量と多様性

データの規模

東京都のような大都市では、日々膨大な量のデータが生成されている。地理情報、人口、交通、エネルギー、環境など、異なる分野で多様なデータが収集されており、これらを効果的に管理することが求められている。

例: 東京都3D地図のデータや、センサーデータ（気象、交通流量など）

データの複雑性

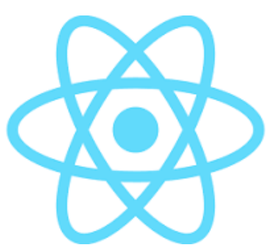
多種多様なデータフォーマットや、リアルタイムでの更新が必要なデータなど、取り扱いが複雑であるため、これらを一貫して管理するのが困難。

- **1つのファイルで500GB以上**
- **3次元化される地図データ**
- **求められるリアルタイム性**
- **BIM/CIM/GISで異なるデータ形式**

Eukarya

都市データの課題を克服するための最新技術

都市データの規模や複雑さが増す中で、従来の技術では対応が難しい場面が増えてきた。これを解決するために、**最新のWeb技術を活用した効率的なデータ管理と配信が鍵となる。**



Eukarya

2015年以降のWeb技術の飛躍（フロントエンド）

バグ発生の最小化

エラーの発見が難しく、大規模なプロジェクトでは特にバグの発生リスクが高かった。

- **JavaScriptに「型」を導入し、大規模開発でも安全性を向上。**
- **モジュール化と互換性が強化。**
- **ライブラリやモジュールの管理が一元化され、簡単に使えるように。**

複雑なUIの実装

特に複雑なUIを扱う場合、パフォーマンスの低下が問題となることが多かった

- **再利用可能なコンポーネントを使って、効率的にUIを開発できるようになった。**
- **DOMの効率的な更新が可能になり、パフォーマンスが向上。**

Webページの軽量化

ユーザーが操作するたびにサーバーにリクエストが送られ、ページ全体がリロードされて、UXが大幅に損なわれていた。

- **ページ遷移を伴わないアプリケーションで、ユーザー体験が向上。**
- **サーバーサイドレンダリングによりSEOや初回表示のパフォーマンスを改善。**

関連する技術

TypeScriptの普及, webpackやBabelの普及, npmによるエコシステムの形成, ES2015(ES6)の策定, Reactとコンポーネント指向, 仮想DOMの導入, SPA(Single Page Application), Next.jsによるSSRの普及

Eukarya

2015年以降のWeb技術の飛躍（バックエンド）

インフラの互換性

サーバーごとに仮想マシンを起動し、リソースを分割していたが、これは重く、リソースの無駄が多い問題があった。

- **コンテナ技術の登場により、従来に比べて、軽量かつ高速な仮想化が可能に**
- **インフラの一貫性と信頼性が向上し、変更や更新による問題が減少**

サーバー管理からの開放

従来のインフラ管理は、多くを手動で行う必要があった。これには、リソースの管理やサーバー障害への対応が伴っていた。

- **サーバーレスアーキテクチャにより、インフラ管理を意識しないでアプリケーションを動かせる環境が普及した。**
- **開発者はインフラの管理から解放されるようになる。**

サーバー処理の軽量化

従来のBE言語は、ランタイムが大きく、多くのリソースを消費し、パフォーマンスや可搬性の面で課題があった。

- **Go言語はコンテナ時代に適した言語として、シンプルで効率的なバックエンド開発が可能に。**
- **シングルバイナリのコンパイルが可能で、軽量かつ高速な実行環境を提供する。**

関連する技術

Dockerとコンテナ技術の普及、サーバーレスアーキテクチャ（CaaS, PaaS, FaaSなど）、Go言語によるコンテナ時代のバックエンド開発、イクロサービスとKubernetesの普及、クラウド管理のコード化（Infrastructure as Code: IaC）とDevOpsの普及

Eukarya

2015年以降のWeb技術の飛躍（その他）

データベースの進化

データを一貫して管理するためにRDBに依存しており、性能面でのスケーラビリティの問題があった。

- **NoSQLは、非構造化データや大規模なデータセットを効率的に扱えるようになった。**
- **NewSQLは、NoSQLのスケーラビリティとRDBの一貫性を両立させた新しいデータベース。**

認証認可技術の進化

各サービス間での認証や認可の処理は一元化されておらず、手動での実装が必要だった。

- **JWT: 改ざん防止の署名付きトークンで、スケーラブルな認証に適す。**
- **OAuth: サービス間での認可を提供し、トークンベースで安全なアクセス権限の移譲が可能に。**

通信プロトコルの進化

効率的な通信やストリーミングには限界があり、特に複数リソースの読み込み時に遅延が発生しやすかった。

- **HTTP/2: ページの読み込み速度が飛躍的に向上し、効率的なデータ転送が可能になる。**
- **リアルタイム性の向上を目指し、よりインタラクティブなWebアプリケーションが開発へ**

関連する技術

NoSQL, NewSQLの登場、JWTとOAuthの普及、OpenID Connectの拡張、HTTP/2, HTTP/3の策定、WebSocketやWebRTCの規格化

Eukarya

2015年以降のWeb技術の飛躍（その他）

APIの効率化

従来のAPIは、クライアント側のニーズに合わない大量のデータを送信することが多く、非効率なデータ転送が問題だった。

- **クライアントが必要なデータだけを取得できるため、API呼び出しの効率が向上し、モバイルアプリやシングルページアプリケーション（SPA）で特に有効となった。**

API開発の柔軟性

APIの設計や変更が煩雑で、開発者の手作業によるエラーが発生しやすい状況だった。

- **OpenAPI仕様により、開発の初期段階からAPI仕様を確定させることで、APIの設計変更が容易になり、フロントエンド・バックエンドの連携がスムーズに。**

関連する技術

GraphQLの登場、スキーマ駆動開発とOpenAPIの普及

Eukarya

それでもまだあるWebの弱点

- **基本的にインターネット接続が前提**

  オフラインだけで作業を完結させるのは難しい。特にクラウドアプリケーションの場合、常時多数の通信が発生するため、ユーザー側がどのような通信がどれだけ発生するかを予測・コントロールするのは困難。

- **開発言語がJavaScriptに限定される**

  CやC++などの言語を自由に選んで開発するのは難しく、クラウドサーバーモデルではJavaScriptが主に使用される。JavaScriptは事前コンパイルが難しく、字句解析や構文解析に伴うオーバーヘッドが発生するため、ネイティブコードと比較してパフォーマンス面で不利。

- **ハードウェアの性能を引き出しにくい**

  OSやハードウェアに直接アクセスできないため、安全性とマルチプラットフォーム対応が確保される一方で、パフォーマンスを最大限に引き出すことが難しい。OSやハードウェアの違いを吸収するために、APIは最大公約数的なものになりがち。

Eukarya

2020年以降のWeb技術の飛躍（WebAssembly）

JavaScriptの制約

従来のWebブラウザ上で動作するアプリケーションは、HTML, CSS, JavaScriptに依存していた。
そのため、フロントエンド開発は、JavaScriptに限定され、JavaScriptの制約で、安全性や高速動作に課題があった。

WebAssemblyの登場

高速に動作

OSやCPUに依存せず、オーバーヘッドが少なく、ネイティブに近い高速な実行が可能。

安全な実行

サンドボックス環境で動作し、第三者のコードでも安全に実行できる。

様々な開発言語が利用可能

C/C++やRustなどの他の言語をWebブラウザで動かすことが可能になる。

WEBASSEMBLY

WebGISの発展可能性

1. ブラウザ上でのネイティブレベルのパフォーマンス
2. クライアントサイドでの大規模データ処理
3. 3D GISとリアルタイムレンダリングの向上
4. クロスプラットフォーム対応
5. プラグイン開発とサードパーティ拡張性
6. オフラインでの高度な処理

Eukarya

従来のWeb技術によるWebGISの課題

GIS業務で求められるパフォーマンスをWebでは実現できない

課題01 高度な解析ができない

- デスクトップGISのような重い解析は、JavaScriptによる実装ではパフォーマンスが不十分だった。
- 高度な解析や大規模データセットを扱う場合、サーバーサイドでの処理しか方法がなく、レスポンスの遅延が多かった。

課題02 大規模データが処理できない

- Webブラウザで扱えるデータサイズは限られており、大規模データ処理においてパフォーマンスの限界、メモリ管理の非効率、シングルスレッド制約などの課題がある。

課題03 地図表示が重い

- ブラウザ内での3D GISは、JavaScriptのパフォーマンスに依存していた。
- 非常に大規模なデータセットや複雑な3Dモデルの表示が難しく、動作が遅くなることが一般的だった。

課題04 アプリ互換性ない

- WebGISを異なるプラットフォーム(デスクトップ、モバイル、AR/VRなど)に展開するには、それぞれのに合わせたアプリ開発が必要になる。
- 異なるOSやデバイスに対応するためのメンテナンスコストが高い。

課題05 サードパーティ拡張性がない

- ブラウザ上で動作するWebGISの拡張性は制限があり、サードパーティが提供する高度な機能を組み込むには難しい。
- QGISにあるプラグインシステムを実現できず、すべての機能が無駄に詰め込まれたWebGISアプリケーションになっていた。

課題06 オフラインで処理ができない

- WebGISを異なるプラットフォーム（デスクトップ、モバイル、AR/VRなど）に展開するには、それぞれのに合わせたアプリ開発が必要になる。
- 異なるOSやデバイスに対応するためのメンテナンスコストが高い。

Eukarya

WebAssemblyによって実現する新しいWebGIS

WebAssemblyの登場により解決可能性がでてきた

課題01 高度な解析ができない

- デスクトップGISのような重い解析は、

ブラウザ上でのネイティブレベルのパフォーマンス

合、サーバーサイドでの処理しか方法がなく、レスポンスの遅延が多かった。

課題02 大規模データが処理できない

- Webブラウザで扱えるデータサイズは限ら

マルチスレッド対応など大規模データ処理

送や処理待ち時間がかかり、ユーザー体験が悪化する。

課題03 地図表示が重い

- ブラウザ内での3D GISは、JavaScriptのパ

レンダリング品質の向上

が一般的だった。

課題04 アプリ互換性ない

- WebGISを異なるプラットフォーム(デスク

クロスプラットフォーム対応

テナンスコストが高い。

課題05 サードパーティ拡張性がない

- ブラウザ上で動作するWebGISの拡張性は

プラグインの開発とサードパーティ拡張性

WebGISアプリケーションになっていた。

課題06 オフラインで処理ができない

- WebGISを異なるプラットフォーム（デス

オフラインでの高度な処理

テナンスコストが高い。

Eukarya

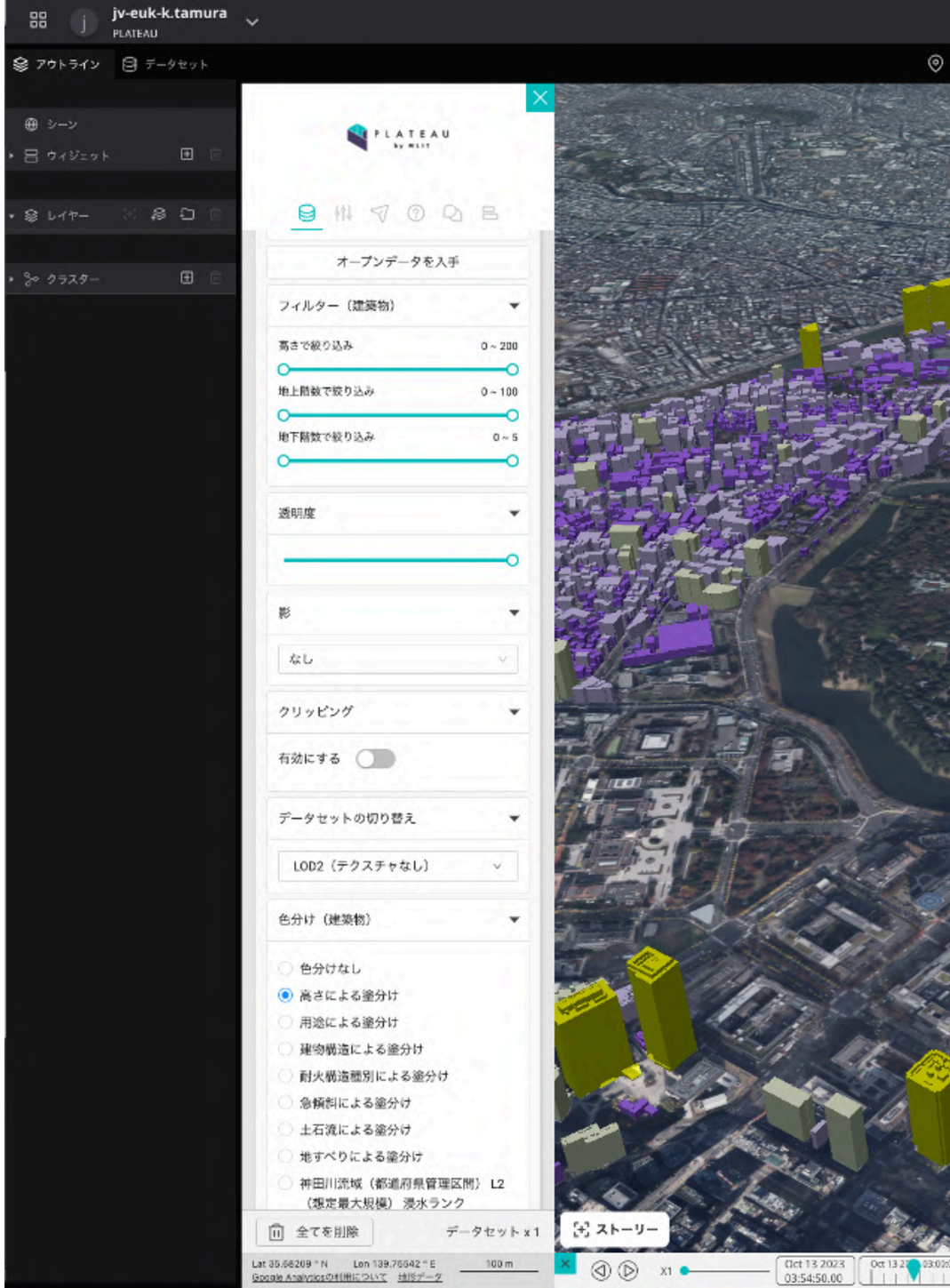
最新Web技術のデジタルツインへの応用

Eukarya

Re:Earth Visualizer

GIS版Figma·Netlify

- React、TypeScript、Go、GraphQL、Cloud Runなどの技術を採用し、Figmaレベルの複雑なUIを実現
- WASM、QuickJS、quickjs-emscripten {,-sync}を活用した、安全なプラグインシステムをWebGISに構築
- Google Cloud Certificate Managerを使用し、カスタムドメイン向けにTLS証明書の発行機能も提供
- 将来的な構想
  - リアルタイム共同編集機能やホットリロード機能
  - Figma Widget APIにインスパイアされた新しいUI APIなどのより開発が容易になるプラグインAPIや



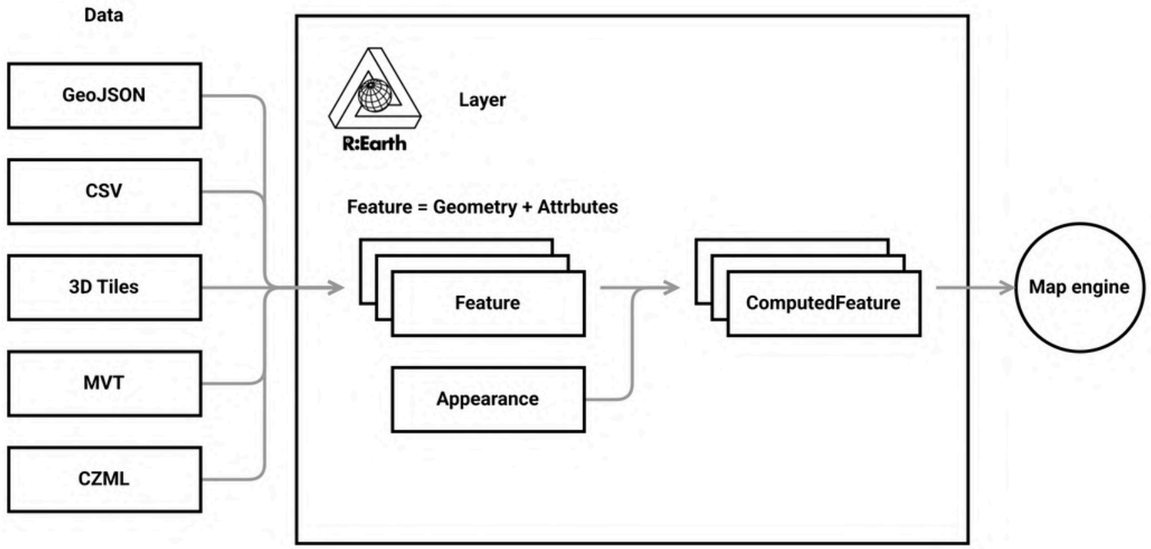
関連する技術

React、TypeScript、Go、GraphQL、MongoDB、Cloud Run、Docker、OpenID Connect、WASM、QuickJS、quickjs-emscripten {,-sync}、CRDT (Conflict-free Replicted Data)

Re:Earth Core

既存のマップエンジンをラップして、
シンプルな宣言的APIを提供するJSライブラリ

- 従来のGISのレイヤー概念をWebに適用することに無理があり、WebGISは大量のデータの可視化が貧弱
  - 属性に応じた細かいスタイルやインフォボックスの変更ができない
  - データ件数が多いとパフォーマンスが大きく低下
  - 様々なデータフォーマットを扱いながら、動的にレイヤーのスタイリングをする機能が貧弱
- 対応するどのデータフォーマットを読み込んでも、同じスタイルを適用可能
- スタイルに式を記述し、属性に応じた色の塗り分けなど、動的なスタイリングも可能。
- 現在はCesiumJSのみをサポート。将来的にはレイヤーやスタイルなどを維持したまま、Mapbox GL JS等の他のマップエンジンを後から切り替えることも可能に



関連する技術

React、TypeScript、Go、GraphQL、MongoDB、Cloud Run、Docker、OpenID Connect、WASM、QuickJS、quickjs-emscripten {,-sync}、CRDT (Conflict-free Replicted Data)

Eukarya

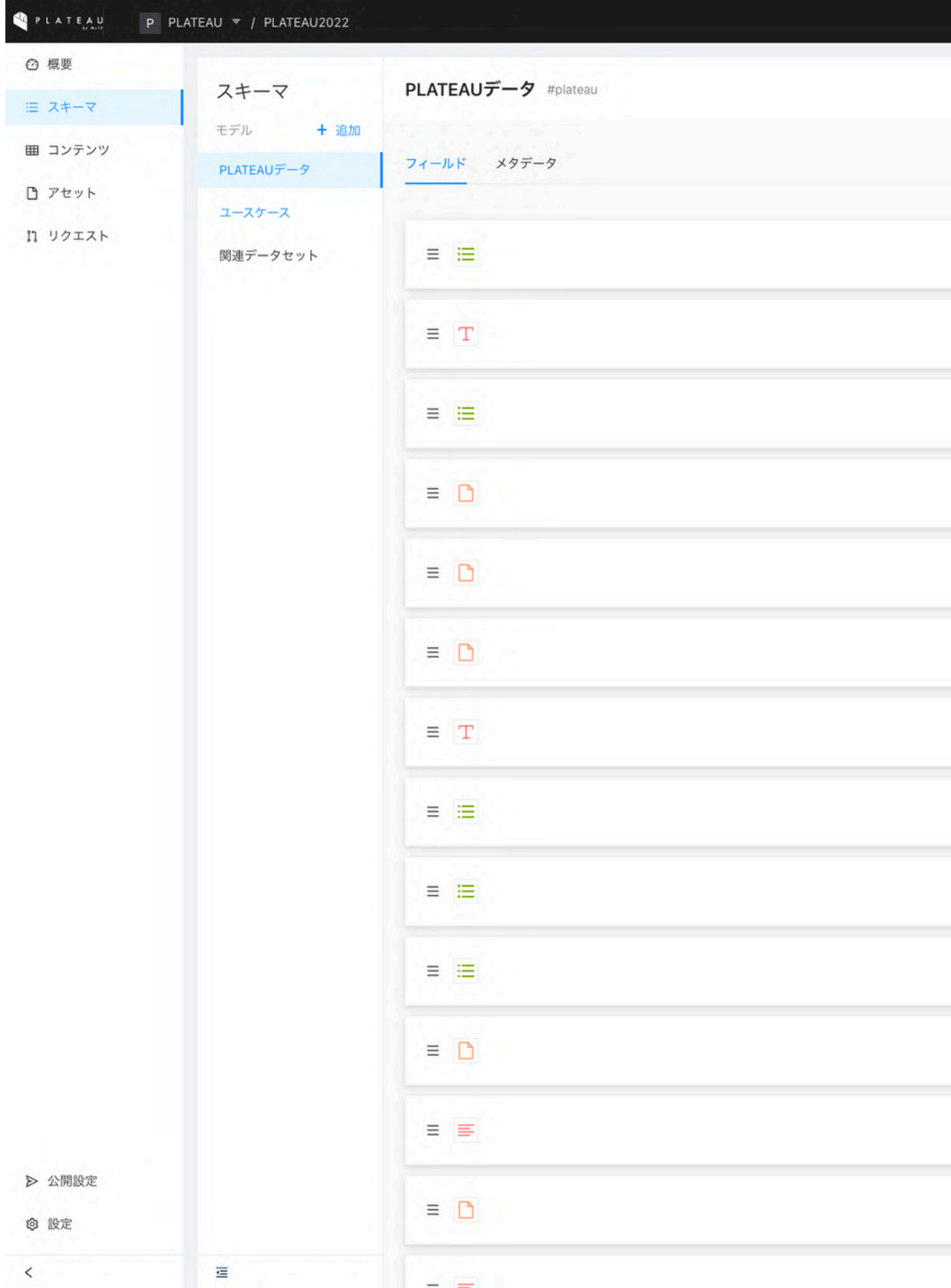
Re:Earth CMS

GIS版Headless CMS

- スキーマレスなMongoDBを使用し、ユーザーが自由にスキーマを作成・編集できる。
- Cloud Buildや各種最適化を行い、数100GB規模のzipファイルをクラウド上で解凍する機能を実現。
- MongoDB上にApIloDBやGitにインスパイアされたバージョン管理システムを実装し、すべてのアイテムの変更履歴を管理可能に
- GeoJSONの地物データをそのままアイテムとしてインポート・エクスポートできる機能も実装中。
- 将来的には、XMLスキーマやCityGMLレベルのスキーマを表現できる機能の導入も検討している。

関連する技術

React、TypeScript、Go、GraphQL、MongoDB、Cloud Run、Docker、OpenID Connect



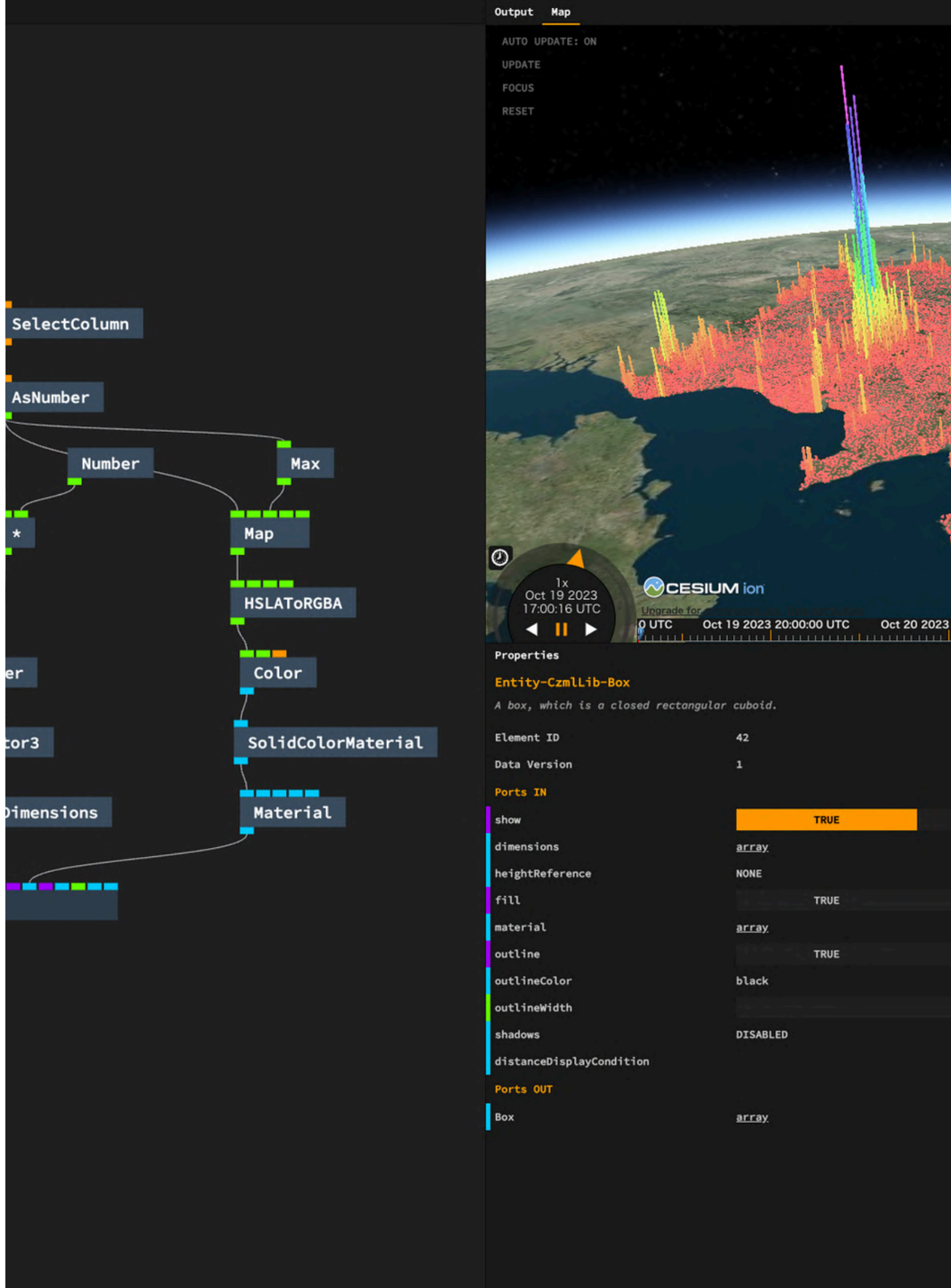
Re:Earth Flow （開発中）

データ品質検査・データ変換・高度な解析

- **サーバーレスな実行基盤**：Rustを用いて高速な処理を実現し、Cloud Batchを用いてサーバーレスでの実行を行う。
- **リアルタイム共同編集機能**：Figmaの仕組みを参考に、CRDT、y.js、WebSocket、Cloud Runを用いて実装を計画中。
- **Python等の安全な実行**：ワーカー内でWASMランタイムを起動し、その中でPythonインタプリタを動作させることで、安全にコードを実行できる。また、WASMにコンパイルされた言語処理系が存在すれば、どの言語でも実行可能。

関連する技術

React、TypeScript、Rust、GraphQL、MongoDB、Cloud Run、Cloud Batch、Docker、OpenID Connect、WASM、CRDT (Conflict-free Replicted Data)、y.js、WebSocket



次世代マップエンジン （開発中）

ヘッドレスな次世代マップエンジン

高速処理が可能なRust・WebAssembly・WebGPUを採用し、マルチスレッドにも対応することで、CPUやGPUなどのハードウェアを最大限活用しながら、データ処理やレンダリングのパフォーマンスを最適化する。

- **高速なGISデータ処理能力：**GISデータ処理を最適化し、3D都市モデルなどの大量のデータを効率的に処理する。
- **高品質な地図表示：**レンダリングの品質を向上させ高品質な地図表示を可能にする。
- **マルチプラットフォーム対応：**異なるプラットフォームに対応するため、プラットフォーム依存の部分を抽象化し、共通のインターフェースを提供する。

関連する技術

React、TypeScript、Three.JS、Rust、WASM、WebGPU, wgpuライブラリ
ECS(Entity Component System)

最新Web技術の
積極的な活用に重要な視点

Eukarya

ベンダーロックインの問題

古い技術遺産

都市データ管理のシステムやインフラは、過去に導入されたベンダー製品に依存している場合が多く、これが技術更新を妨げる要因となっている。特定のベンダーに依存したデータフォーマットやシステムが存在すると、新しい技術を導入する際に互換性や移行コストが大きな障害となる。

- **影響**

  新しい技術やオープンソースソリューションの導入が難しくなり、結果としてシステムの老朽化やメンテナンスコストが増大する。
- **コストの増大**

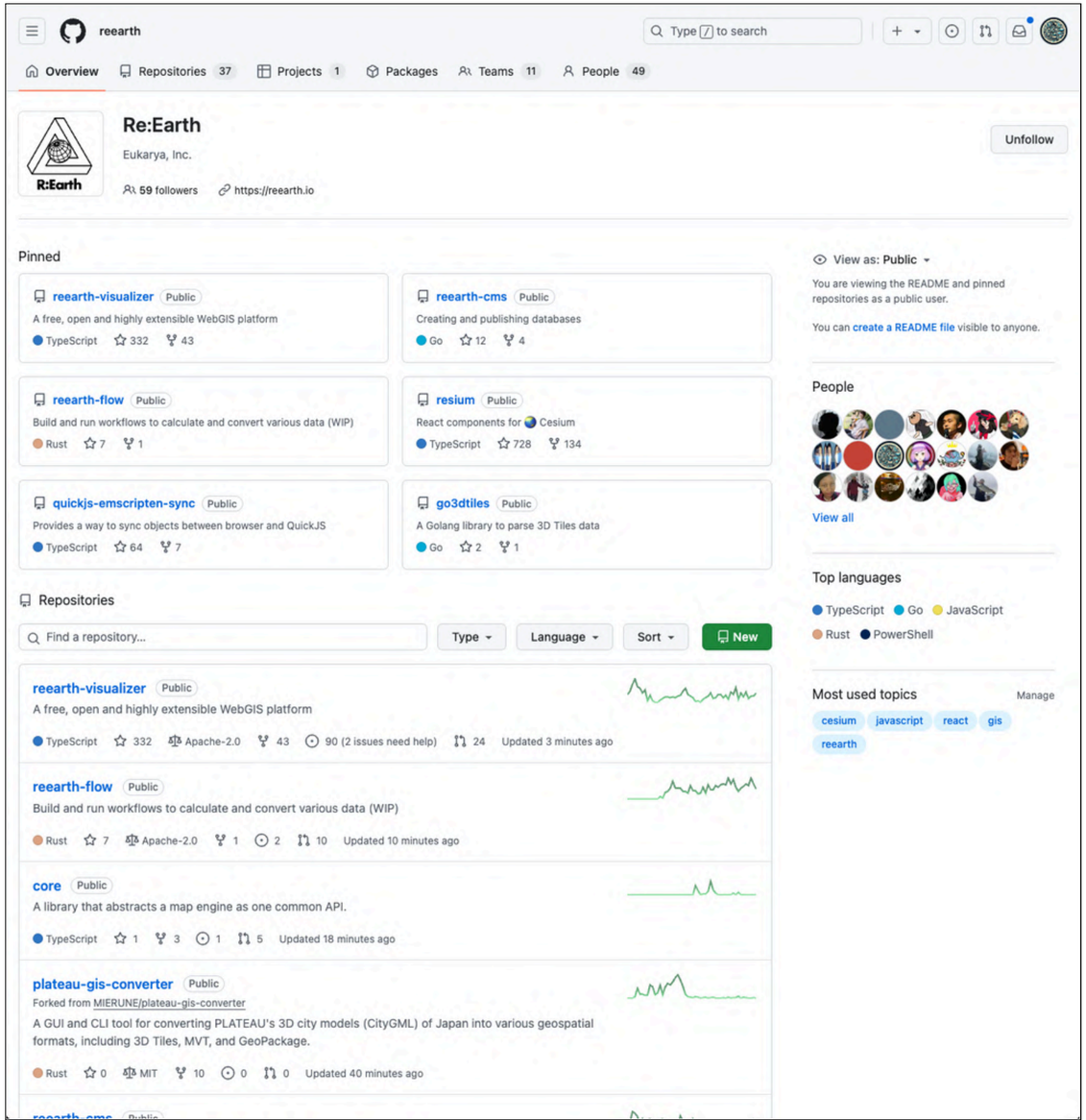
  古い技術を維持するために、ベンダーに支払うライセンス料やサポートコストが増加し、結果として自治体や都市運営の予算を圧迫する要因となる。**同時に民間企業も維持・管理、研究開発費のコスト増大の問題を抱えている。**

Eukarya

Re:EarthはOSSで全てのソースコードを公開

1. **世界中のユーザーからの開発貢献**
    a. グローバルな改善・機能追加
    b. バグ修正やセキュリティの強化
    c. コミュニティによるサポートの充実
    d. 多言語・多文化の対応
2. **優秀なエンジニアの獲得**
    a. エンジニアの興味を引きつける
    b. エンジニアの評価材料としてのOSS
    c. 企業ブランドの向上
    d. 貢献者が将来の社員候補に

最先端テクノロジーの研究開発には、莫大な研究開発投資が必要というのが前提である。
しかし、オープンソース戦略によって、技術に対する探究心と情熱が高いエンジニアや、自発的に新しいアイデアを提案するクリエイター型人材、コミュニティや協調作業を大切にする人材、稼ぐことよりも世の中の役に立ちたい社会貢献型人材など、多様な人材を獲得できる。



Re:EarthはOSSで全てのソースコードを公開

ベンダーロックイン

- 技術革新が起こりにくい構造で、**古い技術遺産が蓄積し、運用・保守コストが増加**する。
- 特定のプロプライエタリなに依存したデータフォーマットやシステムだと、**新しい技術への互換性や移行コストが大きな障害**になる。
- 例え、研究開発・新規開発をやりたくても、クローズドな環境なので、**自前でコストを捻出**しなければならない。

オープンソース

- **ベンダーロックインからの解放**
- **最新技術の知識・技術の獲得**
- **行政も民間も、コストの削減を実現**
- **IT人材不足の日本において、優秀なエンジニア獲得**

Eukarya

Digital Public Goods

デジタルな公共財という視点

Digital Public Goods (DPGs) は、デジタル技術を用いた公共財を指し、誰でも自由にアクセス・使用・改変できるソフトウェア、データ、AIモデル、基準、コンテンツ等で提供されるもの。
SDGsの達成や社会全体の福祉向上を目指し、政府や組織、個人がオープンで協力的に利用できるよう設計されている。

DPGsの特徴

オープンソースまたはオープンアクセス

オープンソースライセンスで公開され、誰もがアクセスし、自由に利用・改変できる。

プライバシー保護とセキュリティ

プライバシーやセキュリティが尊重され、安全な使用が確保されている。これにより、特定の人々や企業に依存しない公平な利用が可能。

持続可能な開発目標（SDGs）への貢献

国連が掲げるSDGsに貢献することを目的としている。例えば、教育、保健、環境、ガバナンスなどの分野で使用されるケースが多くある。

国際的な協力とイノベーション

国境を越えて協力するための基盤を提供し、技術的な格差を縮める役割を果たす。デジタル公共財を活用することで、地域に関係なく平等な機会を提供することを目指す。

DIGITAL PUBLIC GOODS

United Nations

THE UNITED NATIONS SECRETARY-GENERAL'S ROADMAP FOR DIGITAL COOPERATION

PROMOTING DIGITAL PUBLIC GOODS

To unlocking a more equitable world, a global effort is needed to encourage and invest in the creation of digital public goods: open source software, open data, open artificial intelligence models, open standards and open content. This is key to achieving the Sustainable Development Goals. Digital public goods should adhere to privacy and other applicable laws, standards and best practices, do no harm. These should also and help attain the Sustainable Development Goals.

THE WAY FORWARD

1. **DIGITAL PUBLIC GOODS PLATFORMS,** which share digital public goods, engage talent, and pool data sets
    a. Digital Public Goods Alliance, which responds directly to the High-Level Panel's recommendations
2. **PROMOTE ROBUST HUMAN RIGHTS AND GOVERNANCE FRAMEWORKS AND INCLUSION FOR SUCH "GOODS".**
3. **PROMOTE DIGITAL PUBLIC GOODS,** including through greater investment, amplified efforts and strengthen coordination
4. **ALL TO DEPLOY DIGITAL PUBLIC GOODS AS PART OF THEIR IMMEDIATE RESPONSES** and in the future, approaches to achieve the SDGs

LEARN MORE AT: UN.ORG/DIGITAL-ROADMAP

OSSではなく、Digital Public Goodsという視点

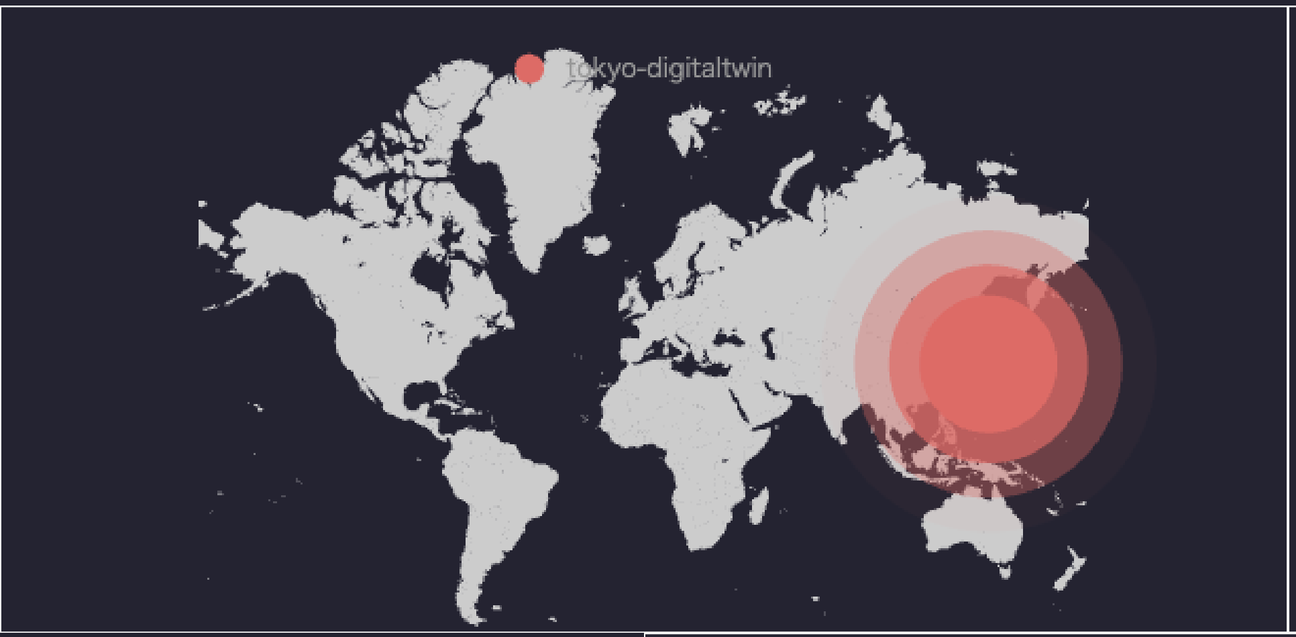
リモートワークは生きる希望を与えてくれる。

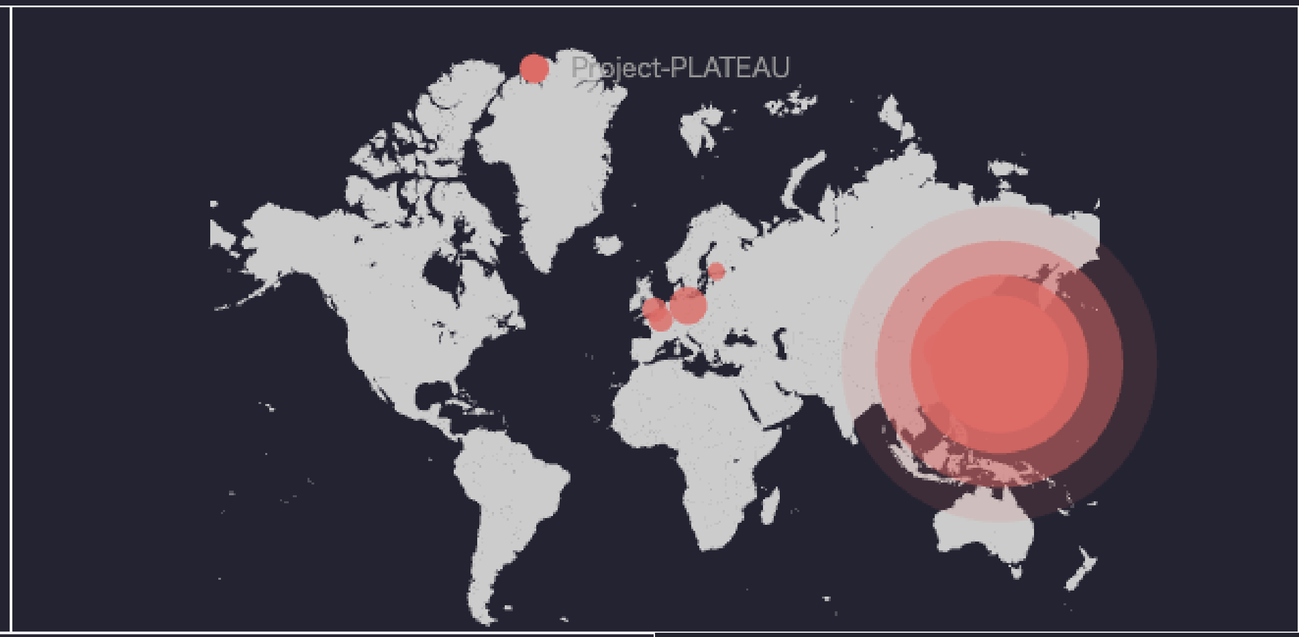
「ユーカリヤと協力することで、経済的にも教育的にも助けられました。国内情勢はいつか改善されると信じています。」

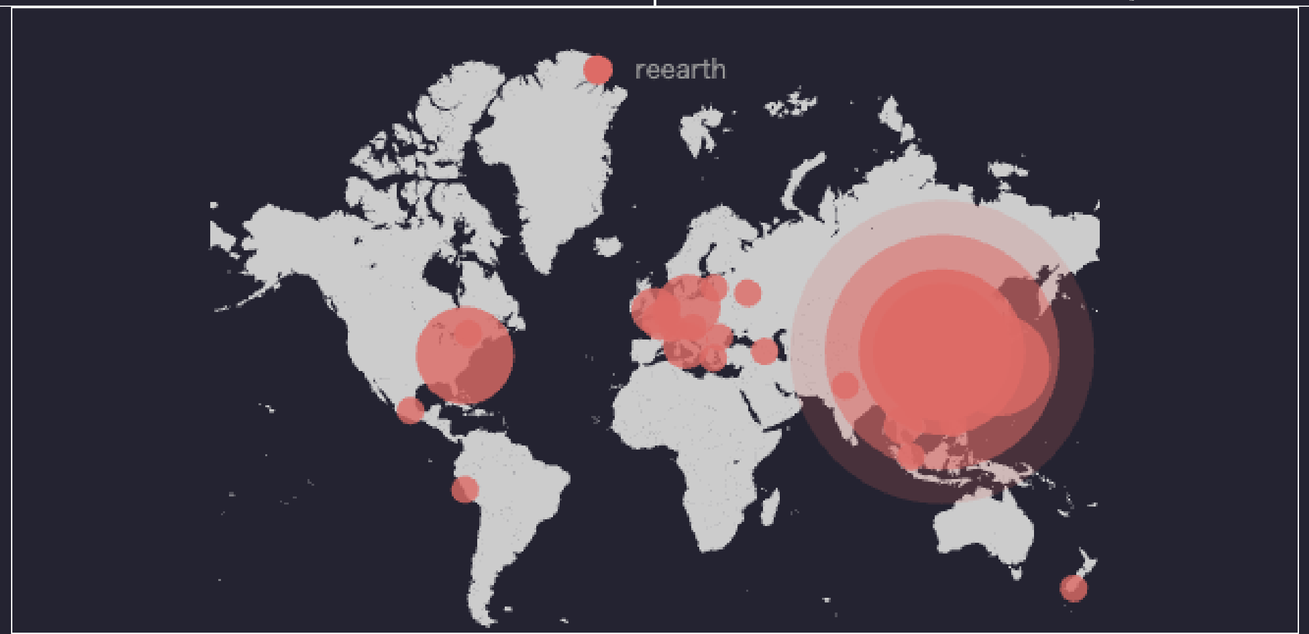
中日新聞「ダマスカスの雪、あるいは『難民』と『デジタル』を巡る私の旅」
https://www.chunichi.co.jp/article/662622/

Eukarya

日本の行政プロジェクトの多くが国際化していない







OSS Insightを用いたGithubスターの分布
左上：東京都デジタルツイン実現プロジェクト
右上：国土交通省Project PLATEAU
下中央：Re:Earth

データが一部の専門家にしか扱えない

技術的な壁

都市データの一部は高度に専門的であり、特定のGISツールや特殊なフォーマットを使う必要があるため、専門知識がないと操作できない。結果として、都市データの管理や運用が限られた技術者や専門家に依存してしまう。

- **影響**

  これにより、データを効果的に活用したい場合でも、技術的スキルがないとアクセスできないため、プロジェクトが生まれない、生まれても進行が遅れることがある。また、専門家不足が運用上のボトルネックになることも。
- **コストの増大**

  古いGISソフトウェアや特定のベンダー製品に依存していると、誰もが簡単に操作できないため、データ活用が停滞するリスクが高まる。

Eukarya

これまでのスマートシティを振り返って

「見捨てられた・見放された・見向きもされない人の存在を忘れていないか」

技術の壁による排除
GISやセンサーデータ、IoTといった技術を使いこなすのは技術者や専門家に限られ、一般市民がそれらにアクセスする手段は乏しく、多くの人が取り残されてしまっている。

参加機会の欠如
スマートシティの施策は、行政や技術企業によって決定されることが一般的です。その結果、技術的な知識がない市民が実際には参加できないという問題が生じている。

デジタルデバイドの拡大
デジタルリテラシーが低い市民は、スマートシティの恩恵を受けにくく、データや技術から疎外される状況がある。

誰でも使える技術を選定する
プログラミングスキルのない市民でも直感的に操作でき、都市データを活用したアプリケーションやサービスを開発するが可能な簡単に使える技術を選定する。

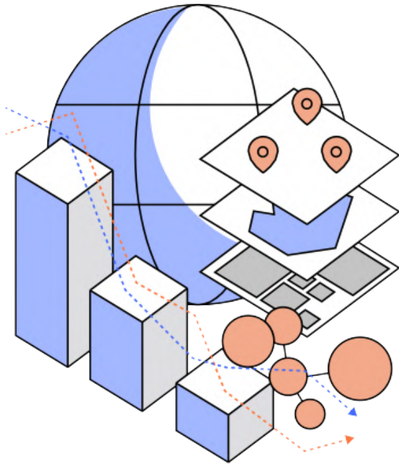
市民が仕事をつくっていける技術を選定する
ノーコードやローコードツールのように、市民が都市データを使って仕事やビジネスを創出できるような要素がある技術を選定する。

低スペックなPCやネット環境で動く技術を選定する
デジタルリテラシーが低い市民は、スマートシティの恩恵を受けにくく、データや技術から疎外される状況がある。

Eukarya

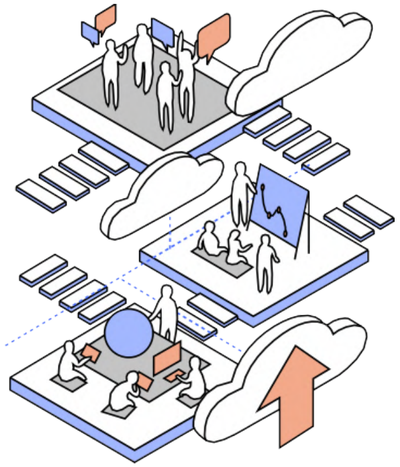
今後のシステム開発の着眼点

先端技術で誰でも都市データを高度に扱えるようにする



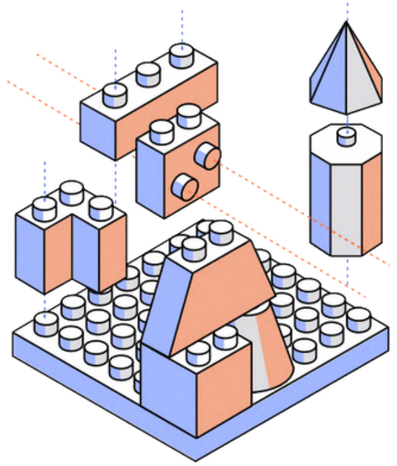
ノーコード

データサイエンティストやエンジニア等の専門家でない人でも、データ加工・管理・配信、分析・可視化、アプリケーション公開までノーコードで扱えるようにする。



Webで完結

専用の高価なソフトウェアは不要で、インターネット環境とWebブラウザさえあれば操作を可能する。リアルタイムデータやセンサーデータも扱えるように



機能の拡張性を担保

例えば、プラグイン機能によって、機能拡張性を担保する。機能追加のための技術的、時間的コストを削減し、類似機能の重複開発は不要

Eukarya

すべての市民が都市データを享受する未来

NHK　NHKについて　コロナ・感染症　ニュース

NEWS WEB　新着　天気　動画　特集　社会　気象・災害　科学・文化　政治　ビジネス　国際　スポーツ

山口 NEWS WEB　◀ 山口の深

下関 テレワーク体験プログラムに参加 成果を発表する会

02月04日　16時17分

下関市豊北町で、テレワークを体験する国のプログラムに参加した人たちが成果を発表する会が開かれました。

人口減少が進む下関市豊北町では去年１１月からテレワークによって地域の課題解決を図る実証事業として、総務省が行う体験プログラムに参加しています。

先月２７日に開かれた発表会には、プログラムに参加した自営業の人や主婦など１２組が、地図上に写真や数字などのデータを重ねて地理空間を分析するＧＩＳ＝地理情報システムを活用した取り組みなどを紹介しました。

このうち、漁師の男性はテレワークで制作した全国の灯台に関するプログラムを紹介し、地図上の灯台をクリックすると灯台の写真とともに高さや歴史、それに灯台に上れるのかどうかなども表示されると説明しました。

また、父と娘の親子で制作した「暮らし体験マップ」は所有者に了解を得たうえで、移住を考えている人向けに地元の空き家の間取りや内部の様子などを写真付きで紹介していて、両者をマッチングする仕組み作りを提案しました。

「暮らし体験マップ」を制作した５０代の自営業の男性は、「テレワークで仕事として実際に何ができるかを感じることができて良かった」と話していました。

https://www3.nhk.or.jp/lnews/yamaguchi/20240204/4060019307.html

日本で約3000万人のデジタルワーカーが不足

※ Amazon Web Services, Inc.（AWS）「APACのデジタルの可能性を拓く：変化するデジタルスキルへのニーズと政策へのアプローチ」調査レポートより

デジタル人材不足の一方で、日本の労働構造による格差

- シングルペアレント世帯の貧困（子どもの貧困等）
- 地方から都市への就労者の集中（地方の人材不足等）

DPGsと最新Web技術の組み合わせによって、
市民が街のデータを作成・管理・配信・活用する
デジタルタレントへ！

Eukarya

Web + GISのスキルには需要がある

複雑で大規模化される都市データを扱えるWebGISのニーズが増している

Civil Engineering

Web Engineering

City Planning

UI/UX Design

Disaster Prevention

Wed GIS Skill

Eukarya

NEW AGENDA

最先端GISによるジョブインクルージョン

Tech

GIS

for

Scheme

OSS

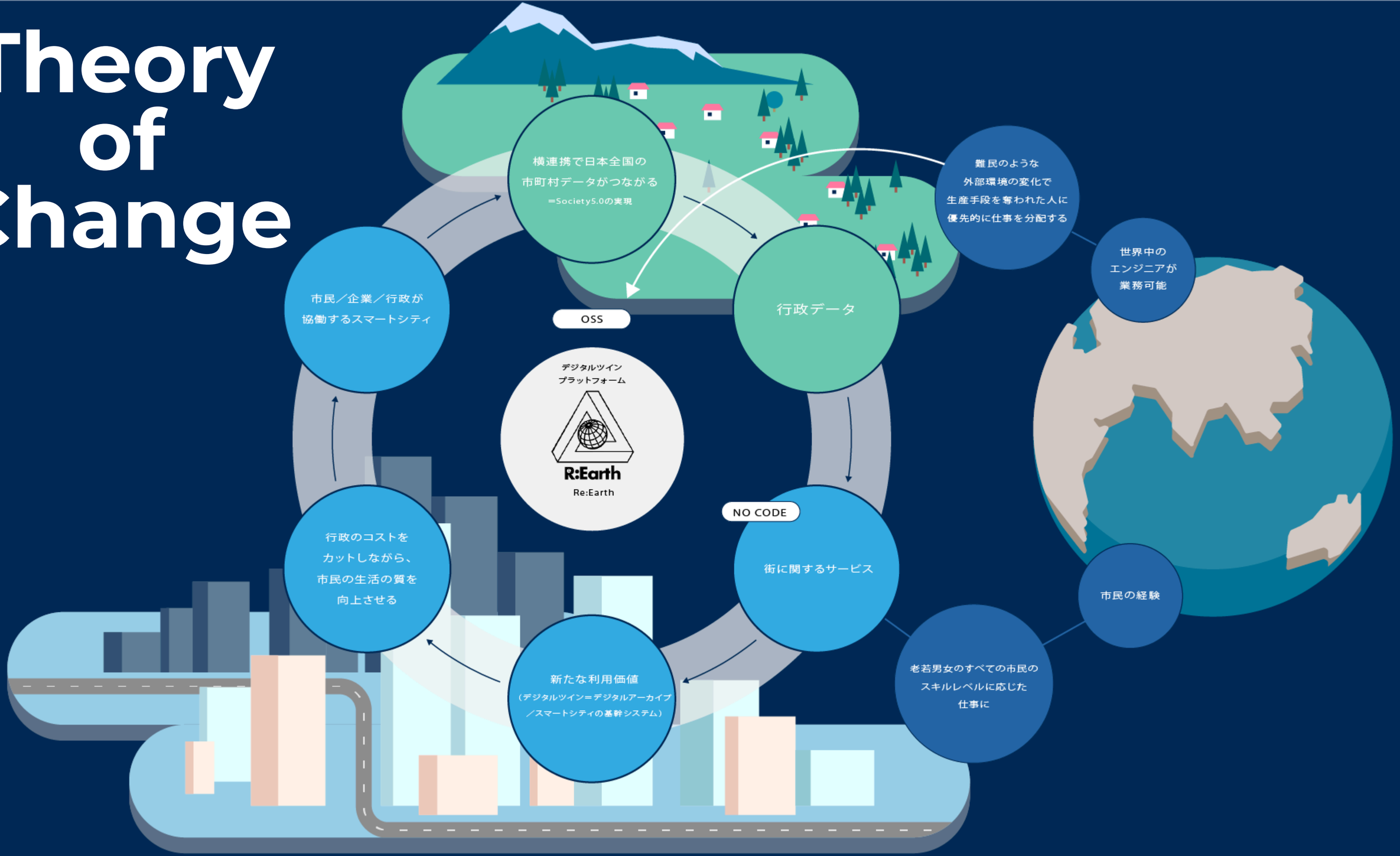
for

Purpose

JOB

Eukarya

Theory of Change



まとめ

大規模な都市データを支える技術が、Webの進化により拓けている

1. GISでは、Web化されていない領域や業務も、積極的にWeb化·クラウド化
2. Web技術やゲーム技術で生まれたアイデアをGISの世界に置き換えて、手つかずの領域を発見
3. ノーコードのアプリケーションから、ライブラリ、データ処理エンジンのミドウェアまで、あらゆるレイヤーで積極的にR&Dを

まずは、テクノロジードリブンな3Dデジタルマップの社会実装に向けて、最新技術を積極的に使っていく環境を整備

Eukarya

Re:Earth Engineering Blog

Eukaryaの最新技術をWebで発信中

https://reearth.engineering/

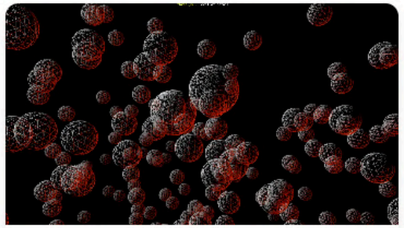
Re:Earth Engineering

Re:Earth's Engineering blog. We will provide daily information about our technology and engineering organization.

2024-09-12

LOD Algorithm in 3DCG

Level of Detail (LOD) algorithms in 3DCG are important for efficient display of complex 3D models. This post summarizes these methods and findings while developing the map engine.



2024-09-12

3DCGにおけるLODアルゴリズム

3DCGにおけるLOD（Level of Detail）アルゴリズムは、複雑な3Dモデルの表示を効率化するために重要です。地図エンジンを開発する中で、多くの手法について学んだので知見をまとめました。

2024-08-02

envsubstを使用してNginx Dockerイメージに環境変数を埋め込む

envsubstを使用してNginxのDockerイメージに動的に環境変数を埋め込むことで、より柔軟なフロントエンドのデプロイが可能になります。その方法について説明します。

2024-08-02

Embed Env Vars in Files on a Nginx Docker Image with envsubst

Learn to configure dynamic environment variables on a Nginx Docker image for flexible deployment. Ideal for high-traffic sites and microservices architectures.

2024-07-24

Take a Screenshot of Cesium Applications with GitHub's GPU Runner and Playwright

Describes how to use GitHub's GPU Runner and Playwright to take screenshots of a Cesium application and detect degregs; how to set up GPU

2024-07-24

GitHubのGPUランナーとPlaywrightでCesiumアプリケーションのスクリーンショットを撮影する

GitHubのGPUランナーとPlaywrightを使用してCesiumアプリケーションのスクリーンショットを撮影し、デグレを検出する方法について説明します。GPU

Thank You