



AUTODESK Platform Services

APS Online Training – Hub Browser

小笠原 龍司

Developer Advocacy & Support、Autodesk Developer Network

アジェンダ – ご紹介する内容

- はじめに – Hub Browser
- デベロッパキーの取得
- ACC カスタム統合の設定
- セットアップ
- 認証と認可
- データのブラウジング
- Viewer と UI



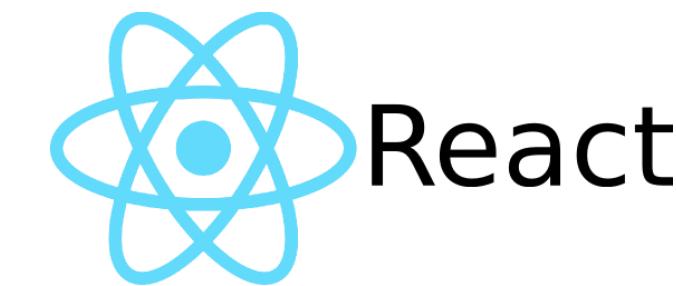
はじめに

インターネットで各社提供 API への接続が可能な時代

- 業界標準&オープンソース API テクノロジ



RESTful API
GET PUT POST DELETE



Autodesk Platform Services – 主要 API

Core APIs



Model
Derivative



Design
Automation



Viewer



Authentication



Webhooks

Data APIs



AEC
Data Model



Data
Management



Manufacturing
Data Model



Data
Exchange

Product APIs



Forma



Fusion



Reality
Capture



ACC



BIM 360



Tandem

RESTful API
GET PUT POST DELETE

React

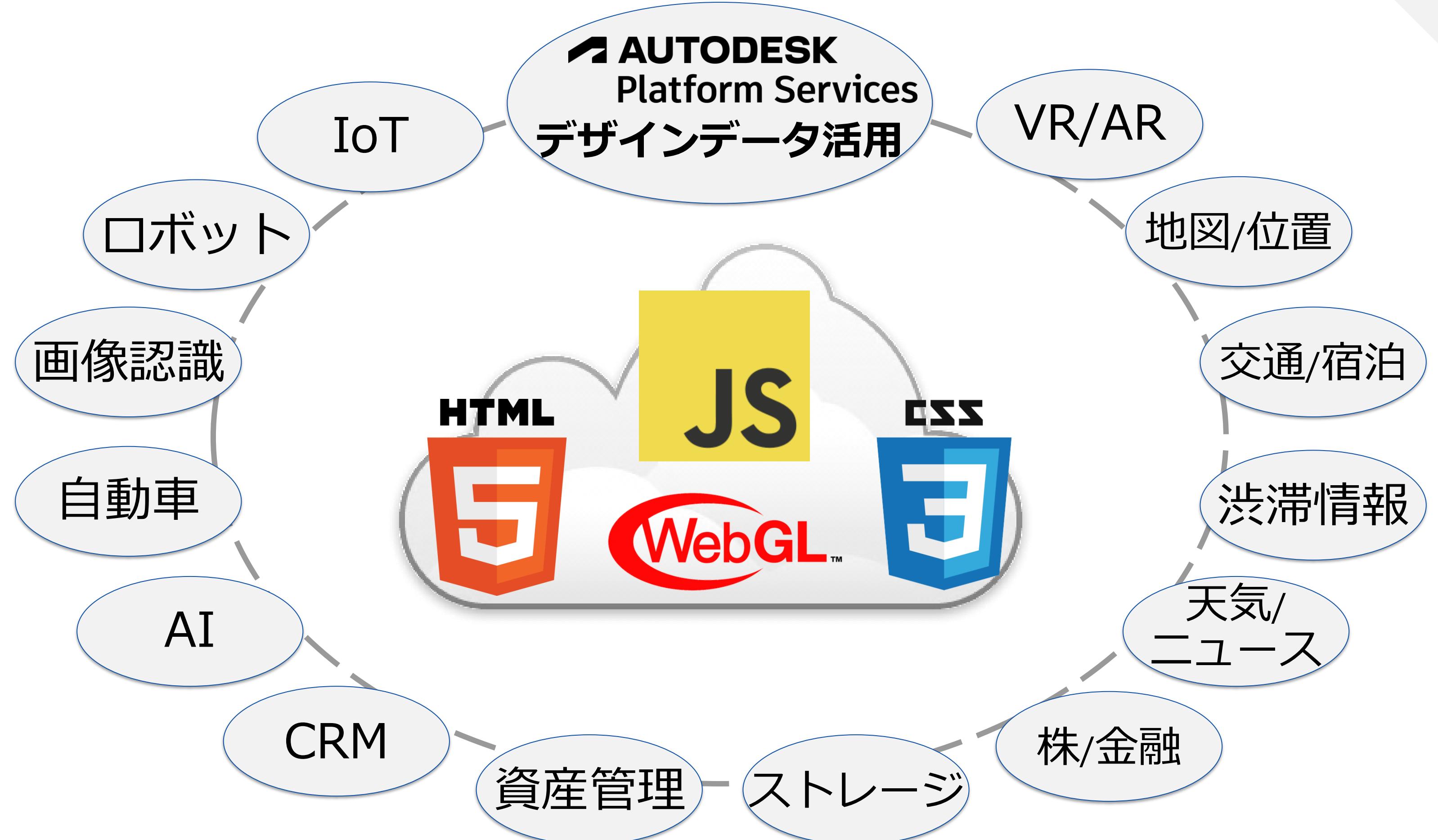
ANGULARJS

Microsoft
.NET

GraphQL

HTML
CSS
JavaScript

node.js™



デスクトップ製品のアドイン開発者の方へ

- デスクトップ開発と Web 開発の違いをご理解ください
 - Web アプリの特性 – セキュリティ視点
 - 通信経路
 - 呼び出し数制限
 - 非同期処理
 - 仮想環境
 - オンプレミスとの違い



関連ブログ記事



公式ドキュメント（英語のみ）

<https://aps.autodesk.com/developer/documentation>

- APS ポータル (<https://aps.autodesk.com/>) に記載

The screenshot shows the Autodesk Platform Services website. At the top, there's a navigation bar with links for Solutions, Getting Started, Documentation (which is currently selected and has a dropdown menu), Success Stories, Community, Support, Pricing, and App Store. Below the navigation bar, the main content area features a large image of a woman holding a transparent 3D-printed model. Overlaid on the image are several colored lines (blue, purple, red, yellow) forming a network or design pattern. To the left of the image, the text "APIs to design anything" is displayed in large, bold, black font. Below this, a smaller text block reads: "Get the APIs and services you need to build better, custom business solutions. Autodesk Platform Services unlocks your Design and Make data to power new ways of working." A blue arrow points to the "APIs & SDKs" link in the Documentation dropdown menu.

APS サンプル – GitHub リポジトリ

<https://aps.autodesk.com/code-samples>

The screenshot shows the Autodesk Platform Services website. At the top, there is a navigation bar with links: Solutions, Getting Started, Documentation (which is currently selected), Success Stories, Community, Support, Pricing, and App Store. A search bar is located above the navigation bar. On the right side of the header is a 'Sign in' button. Below the header, there is a large banner featuring a woman looking at a screen. The banner has the text 'APIs to design' and 'connected'. In the center of the banner, there is a callout box with the text 'APIs & SDKs' and 'Code Samples'. A blue arrow points from the text 'Code Samples' to the word 'Samples' in the callout box. At the bottom of the page, there is a footer with the Autodesk logo. The footer is divided into five sections: Solutions, Documentation, Resources, and About. The 'Code Samples' link is located in the 'Resources' section. A blue arrow points from the text 'Code Samples' in the 'Resources' section to the word 'Samples' in the footer. The footer also includes links for Company overview, Careers, Investor relations, Newsroom, AEC Data Model, Autodesk Construction Cloud, Autodesk Fusion, BIM 360, BuildingConnected API, AEC Data Model, Authentication, Autodesk Construction Cloud, BIM 360, Data Exchange, Get Help, API Status, Blog, FAQ, Code Samples, and ADN Member Sign-in.

Solutions	Documentation	Resources	About
AEC Data Model	AEC Data Model	Get Help	About APS
Autodesk Construction Cloud	Authentication	API Status	Pricing
Autodesk Fusion	Autodesk Construction Cloud	Blog	Success Stories
BIM 360	BIM 360	FAQ	Certified Partners
BuildingConnected API	Data Exchange	Code Samples	Partnerships

学習リソース : Learn APS Tutorial (英語のみ)

<https://get-started.aps.autodesk.com/>

Get the tools and services you need to build better, custom business solutions. Autodesk Platform Services unlocks your Design and Make data to power new ways of working.

[Sign up to try Autodesk Platform Services](#)

**APS Tuesdays**

Join a free weekly webinar to get to know APS

**Learn APS Tutorial**

Step by step guide on GitHub

**Developer Newsletter**

Stay up to date with the developer community

学習リソース : Learn APS Tutorial (英語のみ)

<https://get-started.aps.autodesk.com/>

AUTODESK Platform Services

Solutions ▾ Getting Started Documentation ▾ Success Stories Community ▾ Support ▾ Pricing App Store ▾

Getting Started

Environment Setup

Tutorials

- Simple Viewer
- Hubs Browser**
- Application Setup
- Authentication
- Data Browsing
- Viewer & UI
- Dashboard
- Design Automation
- ACC Administrator
- ACC Issues (beta)
- Learn More

Hubs Browser

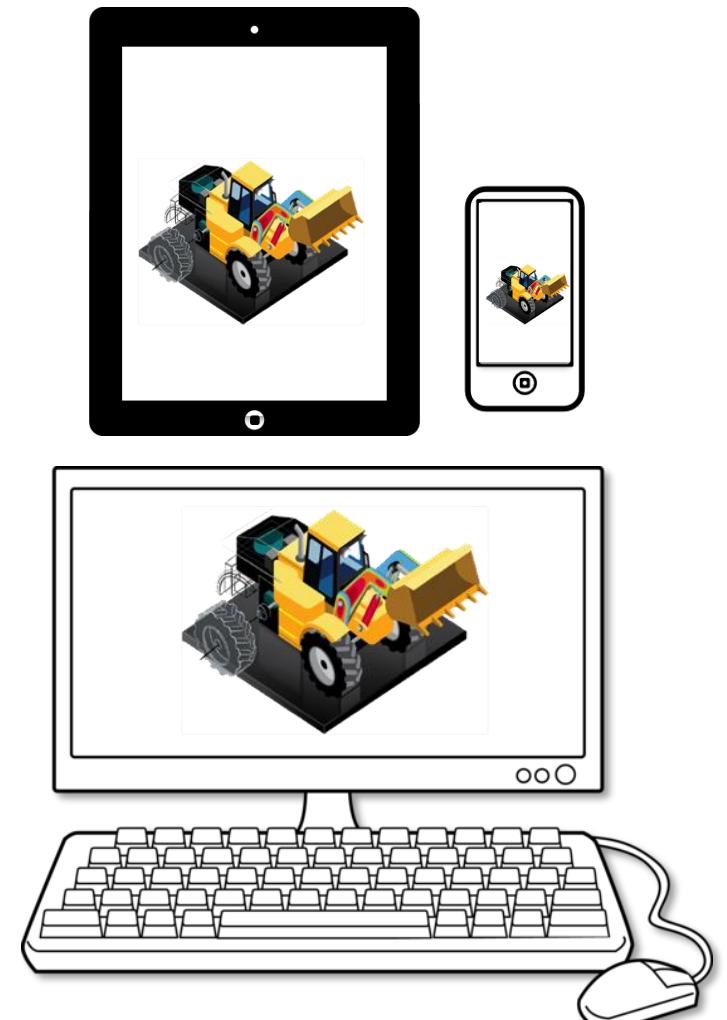
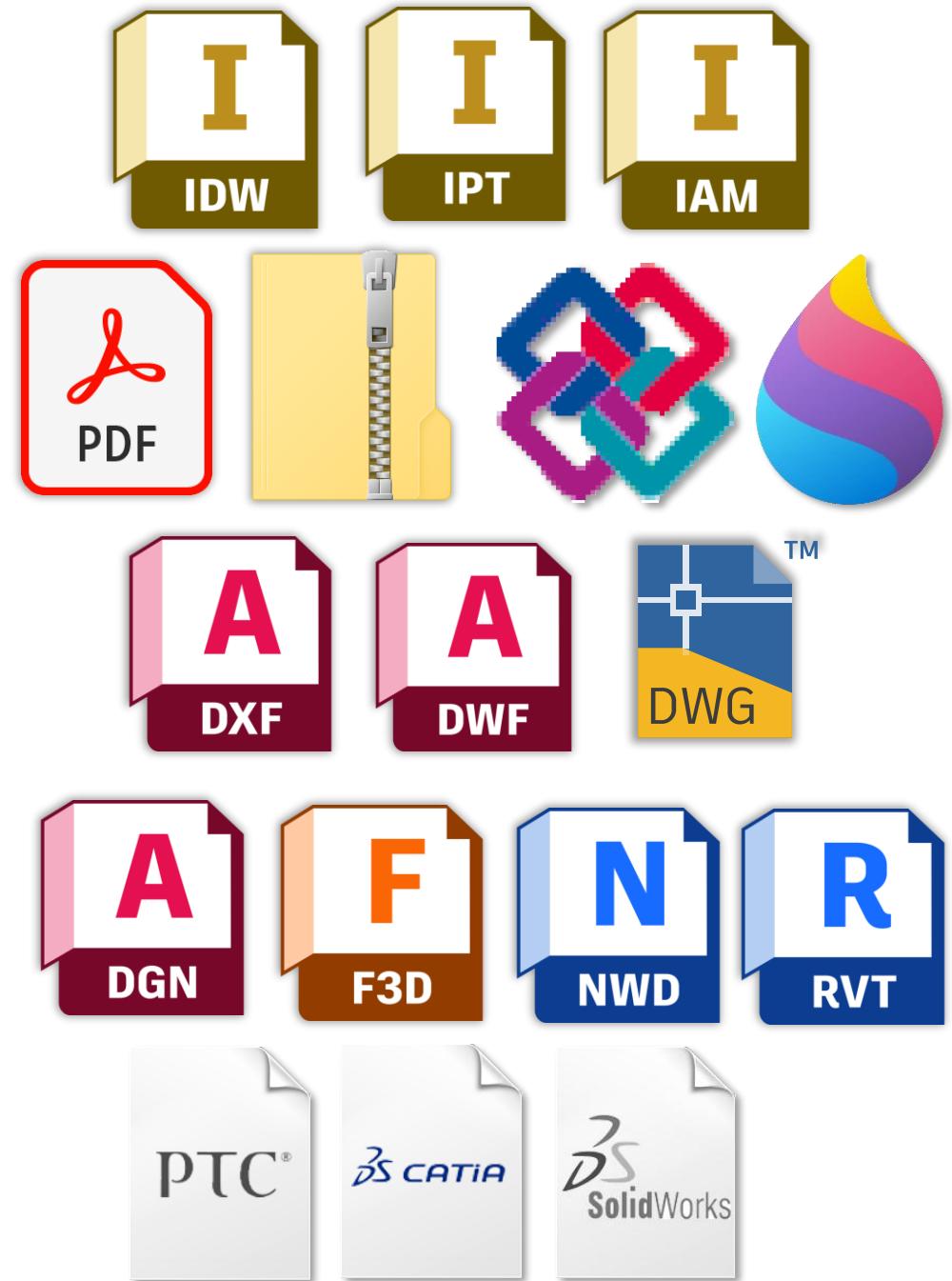
Introduction

In this tutorial we're going to build a APS application that will let us browse through the content of our authenticated users in other APS-based applications such as [BIM 360 Docs](#), [Autodesk Docs](#), or [Fusion Team](#).

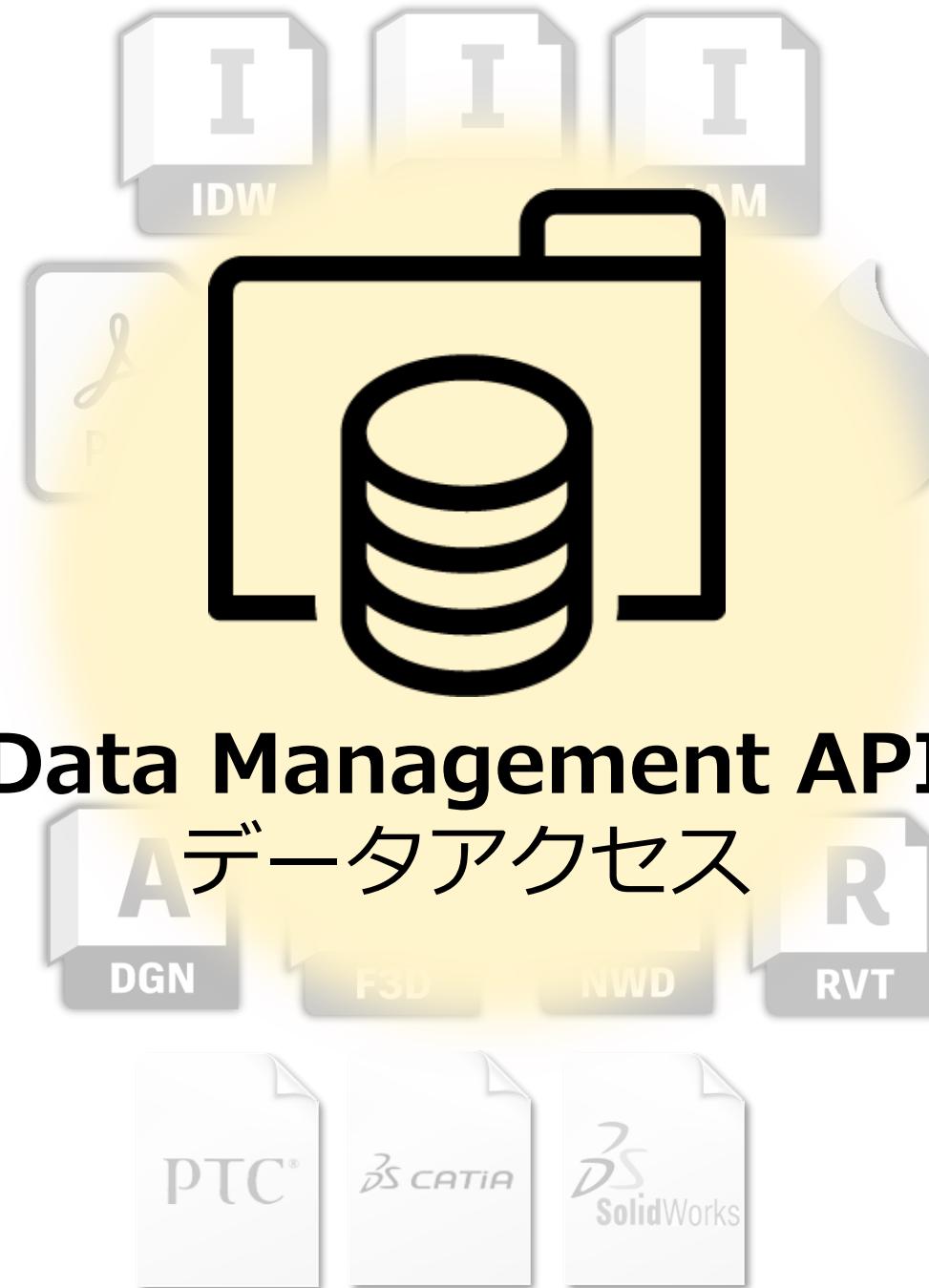
NOTE

Make sure to [provision access](#) to one of your ACC or BIM360 accounts so that the application can see your projects, folders, and documents.

Viewer ソリューション

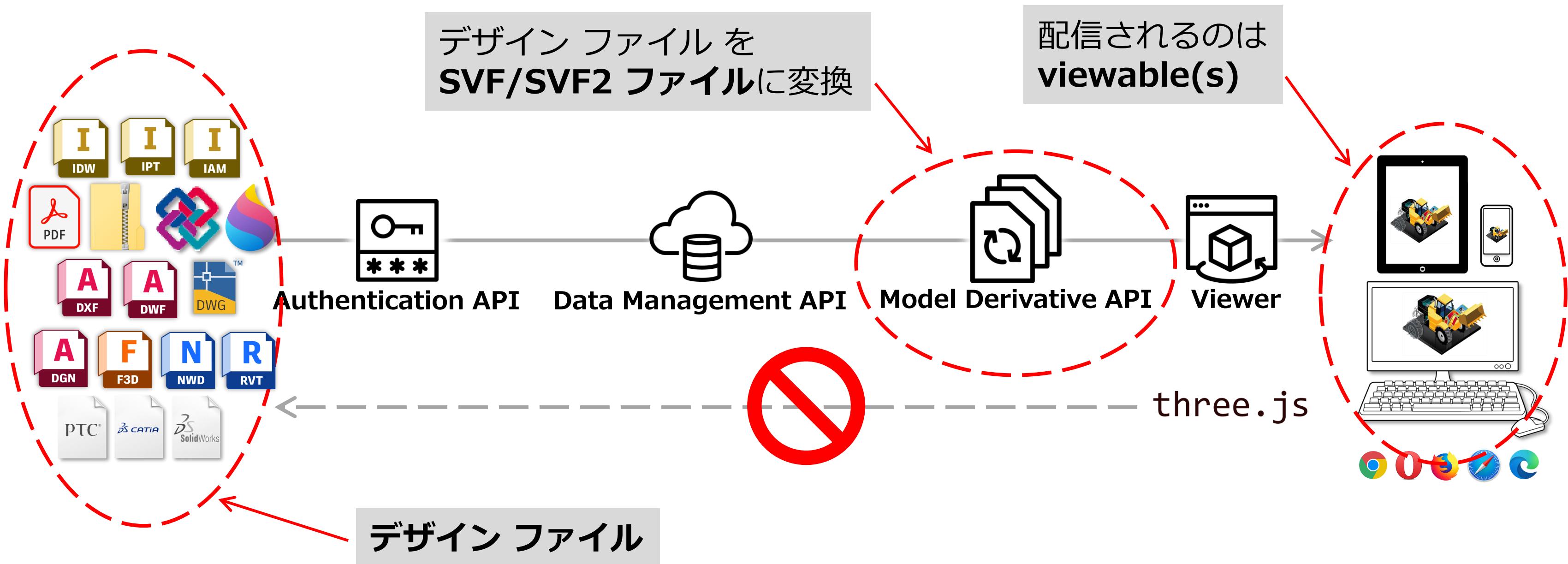


Viewer ソリューション (OSS)

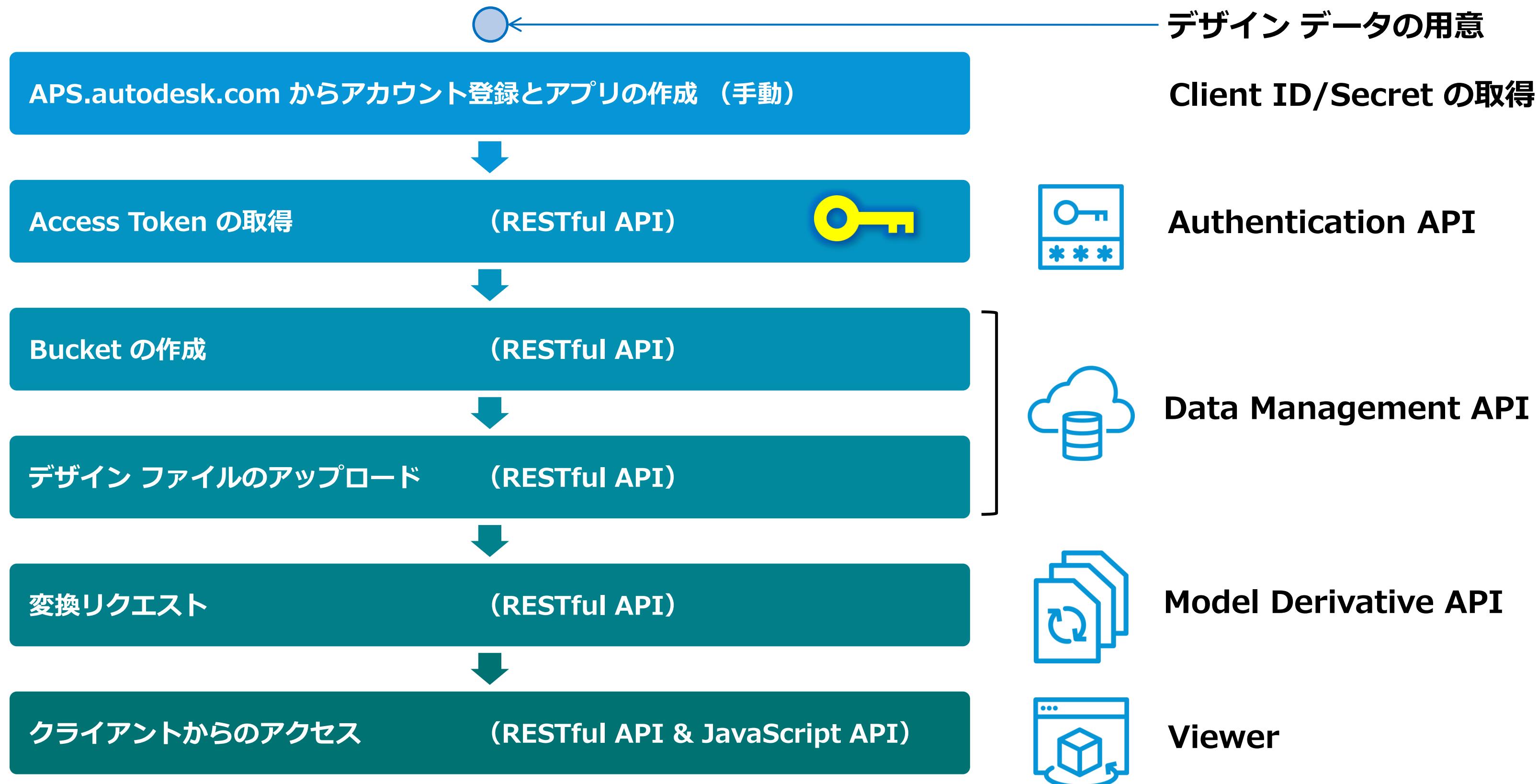


APS Viewer ソリューションのながれ (OSS)

- APS Viewer からのオリジナルデータへの反映は不可

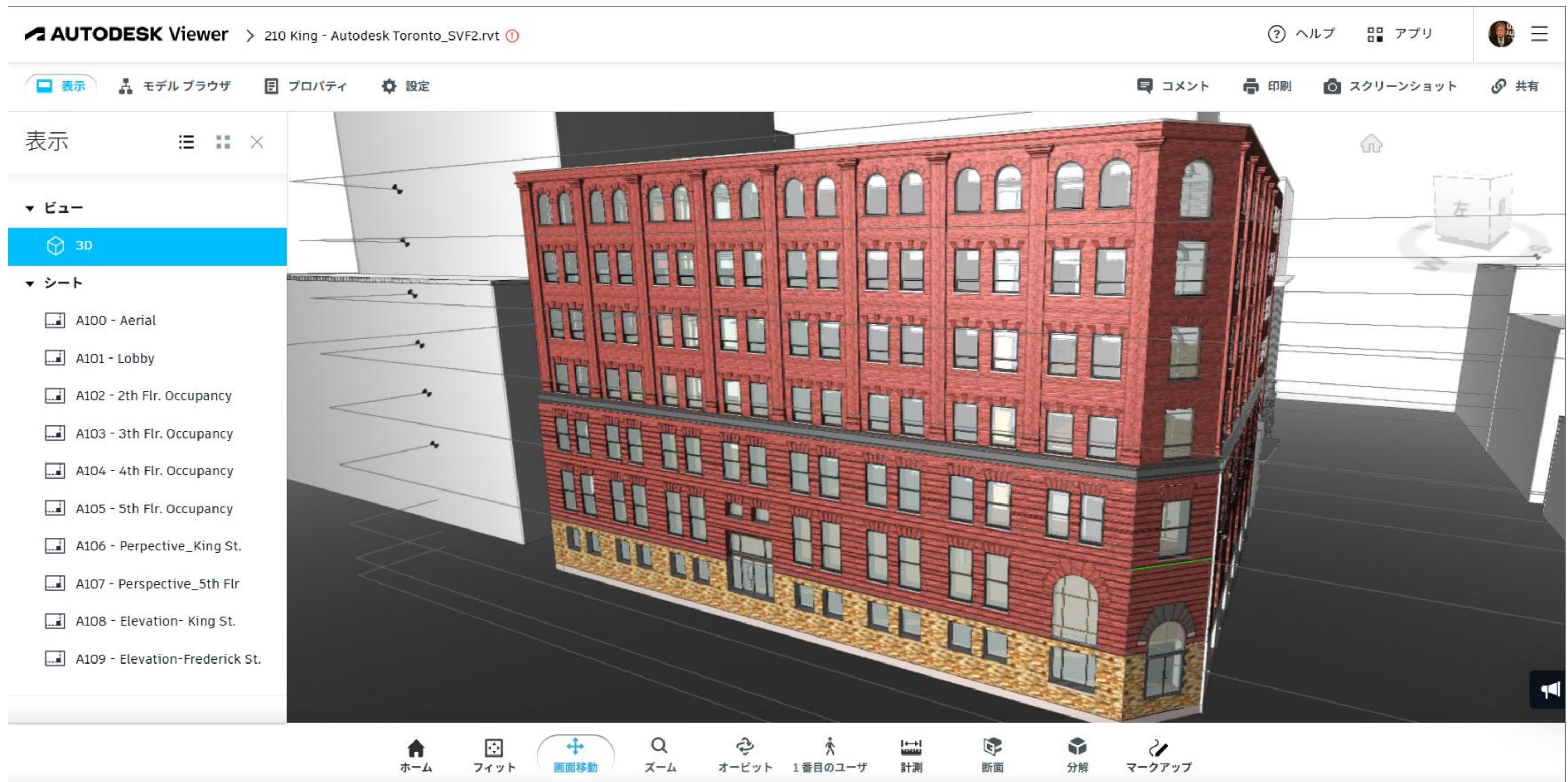


2-legged OauthでのViewer利用手順の理解



Autodesk Viewer

<https://viewer.autodesk.com/>



Autodesk クラウドストレージ サービスの変遷

建設業に特化

建設業に特化

2018年4月

BIM 360

docs.b360.autodesk.com

プロジェクトベース管理

建設業に特化

B BIM 360 TEAM

a360.autodesk.com

プロジェクトベース管理

2016年8月

個人用に特化



AUTODESK® A360

a360.autodesk.com

プロジェクトベース管理

製造業に特化



AUTODESK® A360 DRIVE

360.autodesk.com

フォルダベースの管理

2020年5月



AUTODESK® A360 TEAM

サブスクリプションに特化

AUTODESK Drive

drive.autodesk.com

フォルダベースの管理

F FUSION TEAM

a360.autodesk.com

プロジェクトベース管理

B AUTODESK®
BIM 360® COORDINATE

B AUTODESK®
BIM 360® DESIGN

B AUTODESK®
BIM 360® BUILD

B AUTODESK®
BIM 360™ DOCS



C AUTODESK
BIM Collaborate

T AUTODESK
Takeoff

B AUTODESK
Build

D AUTODESK®
DOC

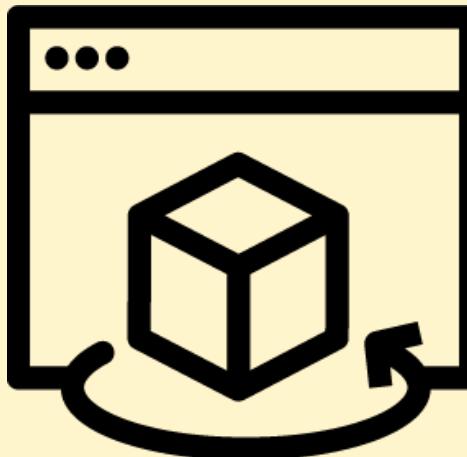


AUTODESK Construction Cloud

ストレージ統合ソリューション (ACC)



ACC API
ACC 固有機能



Viewer
ストリーミング表示



Data Management API
データアクセス

ACC モジュール（機能）別 API

Account Admin

Automate project setup, managing member and partner directories, and syncing data with external systems.

[API reference](#)

Assets

Create, manage, and search assets, improving their organization and workflow efficiency within Autodesk Construction Cloud Assets.

[API reference](#)

AutoSpecs

Access draft submittal logs, extracted from construction project specification documents.

[API reference](#)

Cost Management

Manage budget and cost changes in Autodesk Construction Cloud by accessing, exporting, and importing data to and from external systems.

[API reference](#)

Data Connector

Retrieve data from several Autodesk Construction Cloud services for local analysis, reporting, and business intelligence.

[API reference](#)

Files

Access, upload, and share 2D plans, 3D intelligent models, and any other project documents to maximize collaboration.

[API reference](#)

Forms

Access data stored in the Forms module, which enables your team to securely fill out, review, and manage project forms.

[API reference](#)

Issues

Create and update project-associated issues in Autodesk Construction Cloud for tracking and management of tasks and problems.

[API reference](#)

Locations

Configure a hierarchy of building areas in projects, to identify the location of Assets, Issues, Photos, Forms, and more.

[API reference](#)

Model Coordination

Manage issues when combining 3D models from different design disciplines into unified project coordination spaces.

[API reference](#)

Model Properties

Build and query indexes from properties in intelligent models uploaded to Autodesk and BIM 360 Docs, and calculate changes (diffs) between consecutive model versions.

[API reference](#)

Photos

Access data stored in the Photos module, a single unified place to view and manage photos and videos in Autodesk Construction Cloud.

[API reference](#)

Relationships

Easily create, retrieve, and delete links between entities across all domains in Autodesk Construction Cloud.

[API reference](#)

RFIs

Create, track, and update RFIs, assigning members, transitioning states, and adding comments.

[API reference](#)

Takeoff

Retrieve and update takeoff project components like settings, packages, classifications, and content views.

[API reference](#)

Sheets

Manage and distribute sheets for field use. You can upload, publish, and organize sheet versions and sets.

[API reference](#)

Submittals

Access the submittal items a user has permission to view in a project.

[API reference](#)

ACC : Hub(Account), Project

Hub/Account

Projects

The screenshot shows the Autodesk Construction Cloud interface. On the left, there's a sidebar with various navigation items: Account Admin, Developer Advocacy Support (highlighted with a yellow oval), プロジェクト (Project), メンバー (Members), 会社 (Company), 役割 (Role), プロジェクト テンプレ... (Project Template), ライブラリ (Library), アプリ (App), 製品とツール (Products & Tools), カスタム統合 (Custom Integration), 設定 (Settings), and BIM 360 管理者 (BIM 360 Manager). The main area is titled 'プロジェクト' (Project) and shows a list of projects. The 'Hub/Account' section is labeled above the project list, and the 'Projects' section is labeled below it. A search bar at the top right contains the text 'Ryuji'. The project list includes the following entries:

名前	番号	プラットフォーム	メンバー	会社	ステータス	使用するテン
Ryuji ACC 2022 Test Project			1	1	アクティブ	
Ryuji ACC 2023 Admin API Test Project	HP-0002		0	1	アクティブ	
Ryuji ACC 2023 Test Project			2	1	アクティブ	
Ryuji ACC 2024-2 Admin API Test with T...	HP-0002		0	0	アクティブ	Ryuji ACC 20
Ryuji ACC 2024 Admin API Test Project	HP-0002		1	1	アクティブ	
Ryuji ACC 2024 Admin API Test with Te...	HP-0002		0	0	アクティブ	Ryuji ACC 20
Ryuji ACC 2024 Test Project			1	1	アクティブ	
Ryuji ACC 2025 Test Project			2	1	アクティブ	
Ryuji ACC Design Collaboration Devit 2			1	1	アクティブ	

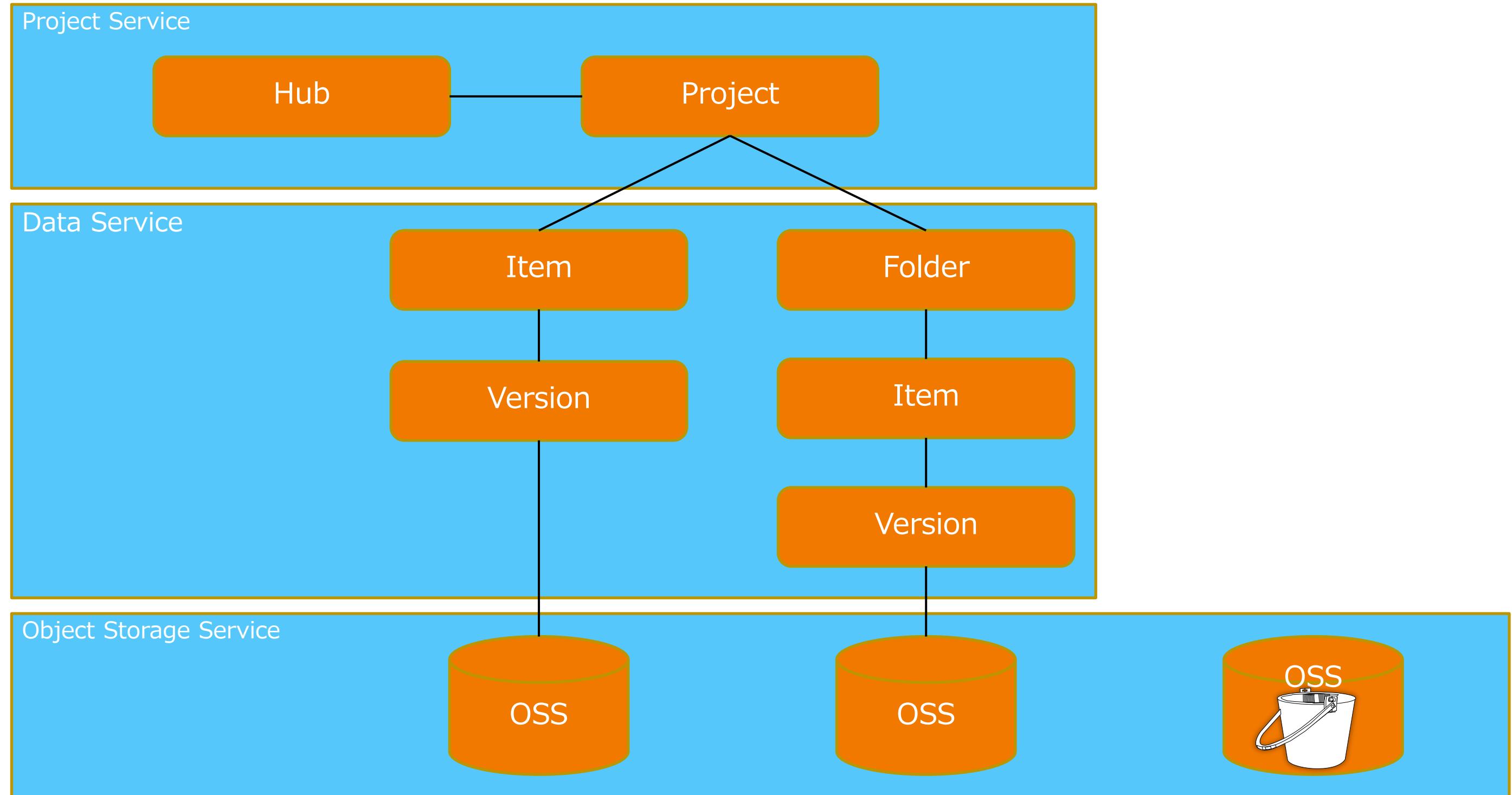
At the bottom, a footer bar shows '1 ~ 19/19 の表示中' (Showing 1 ~ 19/19) and navigation icons.

Autodesk Docs : Folder, Item

The screenshot illustrates the Autodesk Docs interface for managing project files. Key components shown include:

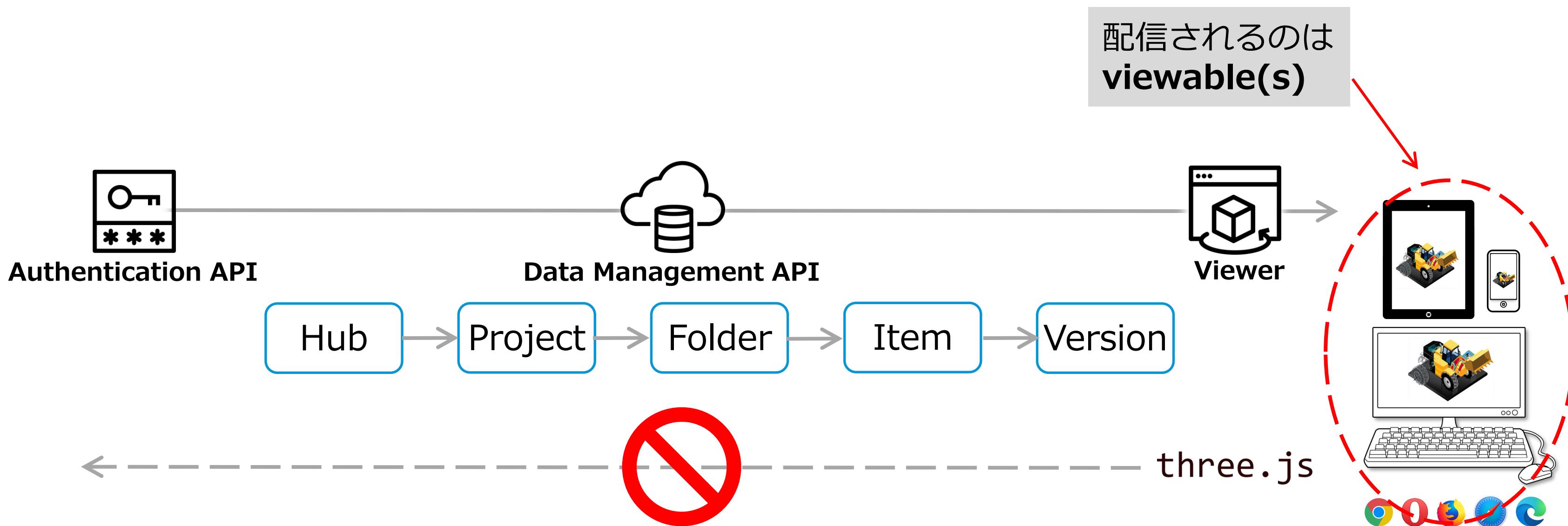
- Top Navigation:** Shows the "AUTODESK Construction Cloud" logo, the "Docs" menu, and the current "Ryuji ACC 2025 Test Project".
- Left Sidebar:** Includes links for "ファイル" (File), "仕様" (Specification), "レビュー" (Review), "ファイル転送" (File Transfer), and "指摘事項" (Matters).
- Central Area:**
 - Project File Folder:** A folder named "プロジェクトファイル" (Project File) containing "サンプル意匠.rvt".
 - User-Created Folder:** A folder named "意匠" (Design) located under the "プロジェクトファイル" folder.
 - Upload Button:** A blue button labeled "ファイルをアップロード" (Upload file).
 - Search and Filter:** Buttons for "書き出し" (Export), "検索とフィルタ" (Search and Filter), and "設定" (Settings).
- Table View:** A table listing items with columns for Name, Notes, Version, Indicator, Markers, Size, and Last Updated. An item named "サンプル意匠.rvt" is listed with Version V1.
- Annotations:**
 - A red oval highlights the "Ryuji ACC 2025 Test Project" in the top navigation bar.
 - Arrows point from the Japanese labels to their corresponding elements in the interface: "プロジェクトファイル" points to the "プロジェクトファイル" folder, "意匠" points to the "意匠" folder, "アイテム/ファイル" (Item) points to the "サンプル意匠.rvt" file, and "アイテムバージョン" (Version) points to the "V1" version indicator.
 - A message at the bottom left says "1 個の項目を表示中" (1 item displayed).

Autodesk クラウドストレージサービスの構造

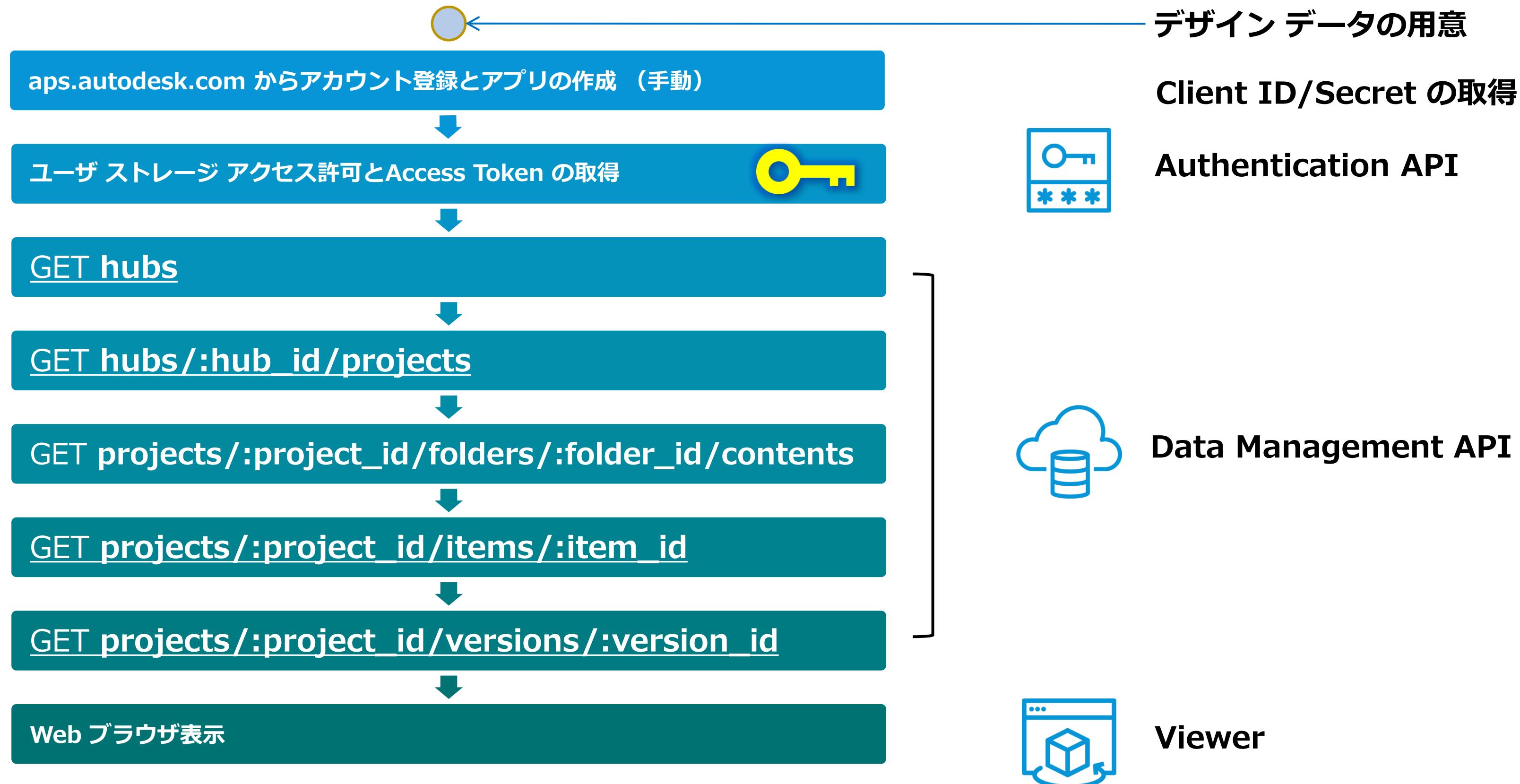


APS Viewer ソリューションの流れ (Hub)

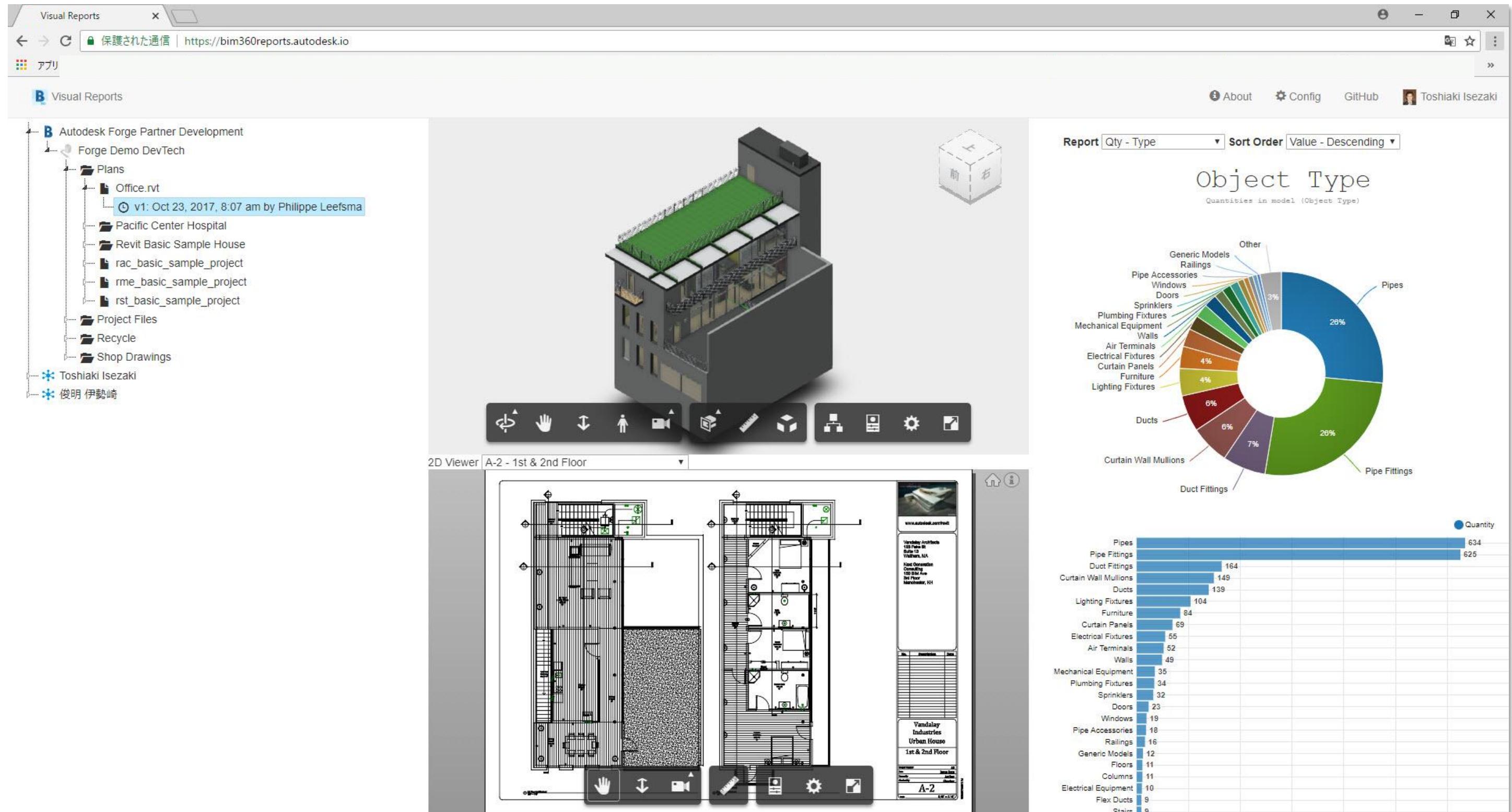
- ACC などクラウドサービスは自動的に変換を実行→変換結果を利用可
 - Model Derivative API で変換する必要はありません。



3-legged OAuthでのViewer利用手順の理解



何が出来るか? – ACC/BIM 360 統合 ダッシュボード



APS SDK : Hub Browser でも利用

- エンドポイントをラップするサーバー用途パッケージ
 - エンドポイントをラップ
 - SDKManager と API 毎に分割したパッケージで提供
 - .NET と Node.js (TypeScript) を用意
- **.NET SDK**
 - NuGet パッケージとして提供
 - <https://www.nuget.org/profiles/AutodeskPlatformServices.SDK>
- **Node.js SDK (TypeScript SDK)**
 - Node.js パッケージとして提供
 - <https://www.npmjs.com/~aps.sdk>



SDK ドキュメント

The screenshot shows the Autodesk Platform Services Developer's Guide for the Data Management API. The left sidebar has sections for Overview, API Basics, Field Guide, Pagination, Filtering, Retention Policies, Commands, Rate Limits, How-to Guide, and Reference Guide. The main content area is titled "Data" and lists several reference links:

- > REST API Reference
- > .NET SDK Reference (OSS)
- > .NET SDK Reference (Data Management)
- > TypeScript SDK Reference (OSS)
- > TypeScript SDK Reference (Data Management)

Below these links, there are sections for Schema Service and Object Storage Service (OSS). A red dashed box highlights the "Reference Guide" section in the sidebar and the "Data" section in the main content area.

エンドポイント リファレンス

.NET SDK リファレンス

Node.js リファレンス

製品機能が全てカバーされているわけではありません

- ACC 自身を拡張するカスタマイズは出来ません
 - Web アプリに ACC/BIM 360 機能を組み込み/利用が可能
- モジュール毎にサポートされている API は異なります
- 理由：さまざま考察が必要
 - 例)
 - 公開 API 利用時のパフォーマンス低下を抑止する必要あり
 1. 機能により多くの API (通信) 呼び出しが集中する可能性
 2. ACC 自体の処理速度や応答が低下してしまう懸念
 3. 呼び出し数制限の設定を検討...
 4. API の変更、廃止、新規追加などに隨時対応が必要

例) [ACC Project Admin API: プロジェクト作成とユーザー管理](#)





デベロッパキーの取得

APS を始めるには?

- まずは <https://aps.autodesk.com/> へ



Autodesk ID で
サインイン



APS アプリ作成



開発

- 目的はデベロッパキーの取得
 - Client ID** (別名 : Consumer Key)
 - Client Secret** (別名 : Consumer Secret)

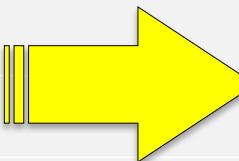
My applications からのアプリ登録

どのAPIを使うか指定

API Access

These are the APIs your app is allowed to access. It can be changed anytime.

- AEC Data Model API
- Application Management API
- Autodesk Construction Cloud API
- BIM 360 API
- BuildingConnected API
- Data Exchange API
- Data Management API
- Design Automation API
- Flow Graph Engine API
- Manufacturing Data Model API
- Model Derivative API
- Parameters API
- Premium Reporting API
- Reality Capture API
- Tandem Data API
- Token Flex Usage Data API
- Webhooks API



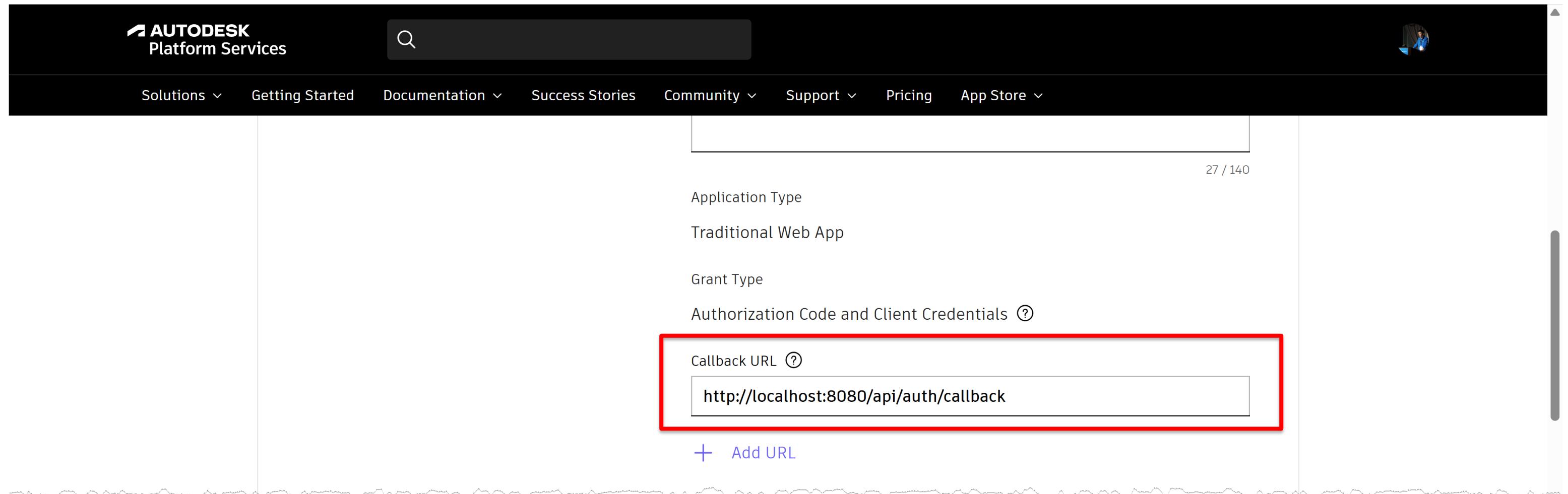
My applications からのアプリ登録



アカウントで複数のチームを編成している場合には、プライマリ管理者、または、セカンダリ管理者に適切なチームを確認してください。通常はアカウントに 1 つのチームしか関連付けられていないので、表示されるチームを割り当て (Assign) ます。

3-legged 認証利用時の必須項目

- Callback URL
 - 今回のチュートリアルでは、`http://localhost:8080/api/auth/callback`





ACC カスタム統合の設定

ACC 統合は「カスタム統合」処理が前提

- APS アプリの ACC アクセスを許可する設定



AUTODESK®
DOCS

- Account Admin アカウントによる事前設定が必須
- BIM 360 時代から存在



AUTODESK Construction Cloud

Account Admin AEC DM Developer Advocacy Support

プロジェクト メンバー 会社 役割 プロジェクトテンプレ... ライブラリ アプリ 力スタム統合 設定 BIM 360 管理者

プロジェクト

アクティブ アーカイブ済み

+ プロジェクトを作成

名前	番号	タイプ	メンバー
AEC DM Bootcamp Project	012024	●	11
AEC DM DAS Test Project	-	●	9
Construction : Sample Project - Seaport Civic Ce...	-	●	2
DAS Japan Project	007	●	2
DAS Project	-	●	5

AEC DM Developer Advocacy Support ▾

?

力スタム統合 ⓘ

+ カスタム統合機能を追加

名前	電子メール	ステータス	追加日	⚙️
AEC Data Model JP test - Toshiaki	[REDACTED]	アクティブ	2023/11/21	⋮
AEC Data Model Sample Zhong ZkKbTg...	[REDACTED]	アクティブ	2023/11/28	⋮
AEC Data Model Test -Carol	[REDACTED]	アクティブ	2023/11/23	⋮
AEC Data Model Test Ryuji	[REDACTED]	アクティブ	2023/12/05	⋮
AEC DM Explorer APP	[REDACTED]	アクティブ	2023/11/10	⋮

カスタム統合機能を追加

×

アカウント ID



統合から API にアクセスする場合は、アカウント ID が必要です。続行する前にアカウント ID を確認してください。後でアカウント ID を参照することもできます [アカウント設定](#)。

Autodesk Platform Services クライアント ID * ⓘ

Autodesk Platform Services クライアント ID

<<Client ID>>

カスタム統合名 *

カスタム統合名を入力

説明

カスタム統合機能の説明を入力

0/255

キャンセル

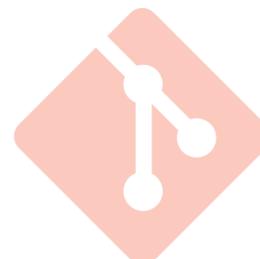
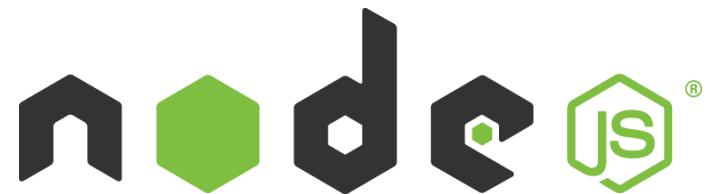
追加



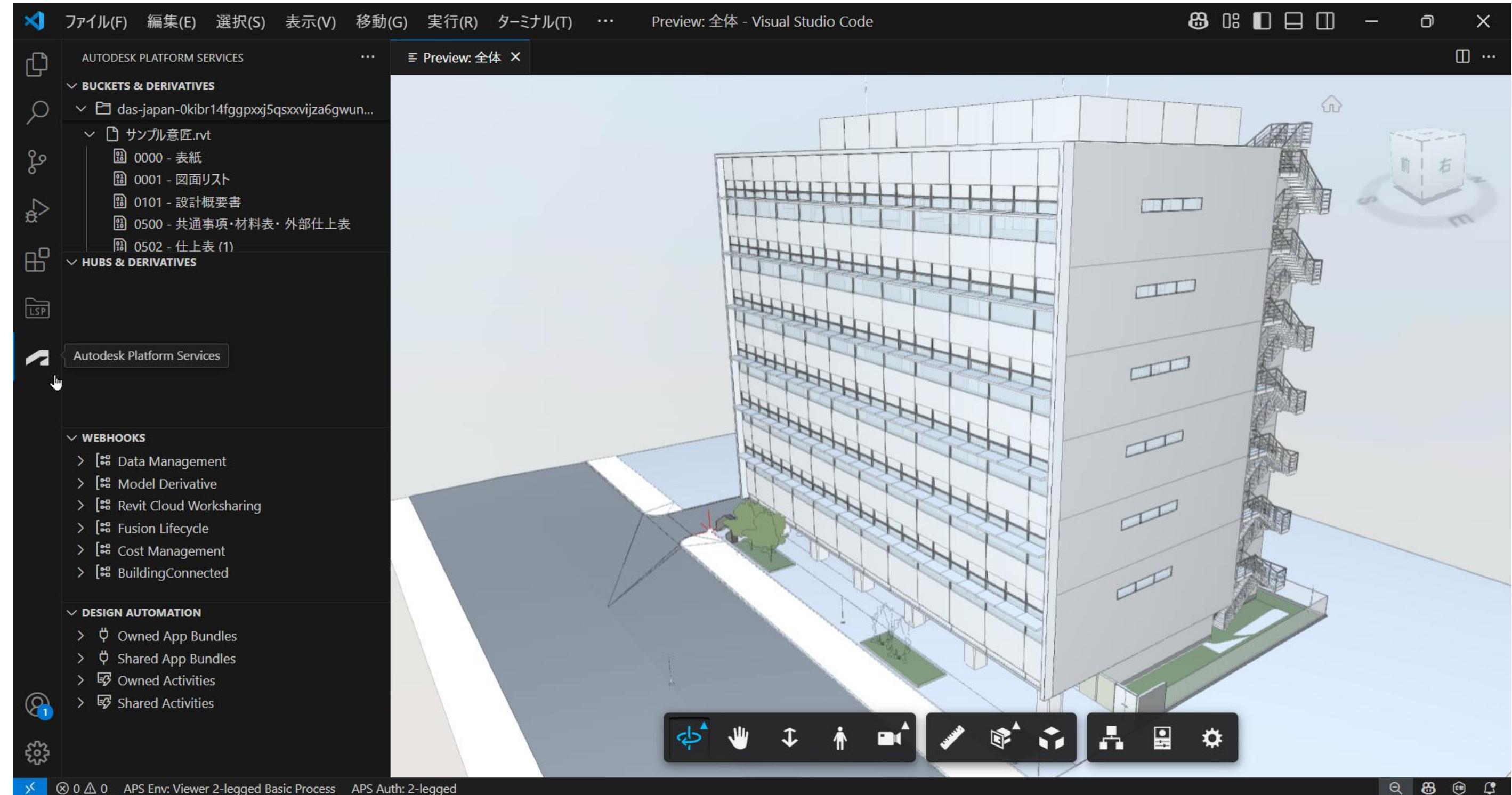
セットアップ

今回使用する開発環境 これに限定するものではありません

- Web ブラウザ : **Google Chrome**
- HTML/JavaScript エディタ : **VS Code**
- Web サーバー実装 : **Node.js**
- テストツール : **Postman**
- リポジトリユーティリティ : **git for Windows**



VS Code - APS エクステンション

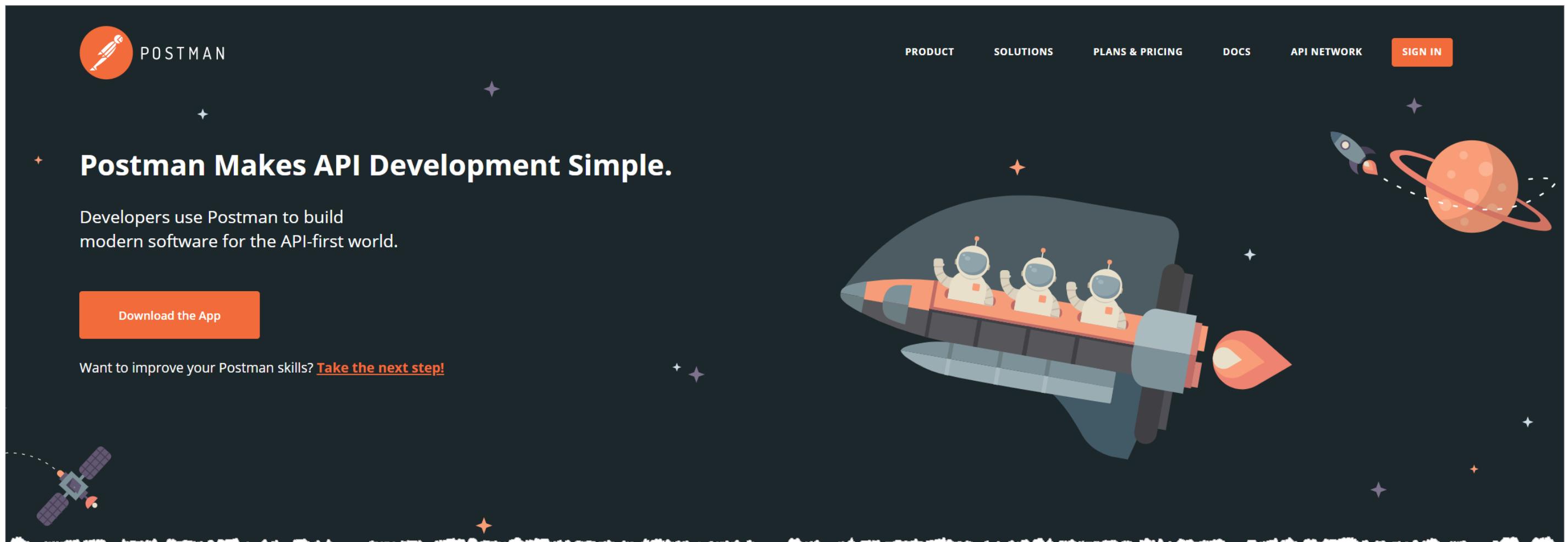


参考ブログ記事：[APS VS Code Extension](#)

RESTful API のテストツール



- Postman
 - <https://www.getpostman.com/>
 - コードを書かずに RESTful API のテストが可能



The screenshot shows the official Postman website homepage. The header features the Postman logo (an orange circle with a white pen icon), navigation links for PRODUCT, SOLUTIONS, PLANS & PRICING, DOCS, API NETWORK, and a red SIGN IN button. The main headline reads "Postman Makes API Development Simple." Below it, a sub-headline states "Developers use Postman to build modern software for the API-first world." A prominent orange "Download the App" button is visible. To the right, there's a cartoon illustration of three astronauts in a rocket ship launching into space, with a planet and a satellite in the background. At the bottom left, there's a link to "Take the next step!"

セットアップ

The screenshot shows a web browser displaying the Autodesk Platform Services website. The header includes the Autodesk logo and navigation links for Solutions, Getting Started, Documentation, Success Stories, Community, Support, Pricing, and App Store. The main content area is titled "Application Setup". On the left, there's a sidebar with links for Getting Started, Environment Setup, Tutorials (Simple Viewer, Hubs Browser, Application Setup), Authentication, Data Browsing, Viewer & UI, Dashboard, Design Automation, ACC Administrator, and Learn More. The "Application Setup" link in the Tutorials section is highlighted in blue. The main content area has a breadcrumb trail: Home > Tutorials > Hubs Browser > Application Setup. The main title is "Application Setup". Below it, a sub-section title is "Create project". There are three options: "Node.js & VSCode" (highlighted with a red dashed box), ".NET & VSCode", and ".NET & VS2022". A call-to-action button says "npm init -y". A small icon in the top right corner indicates the page can be pinned.

Let's start by creating a new project, installing required dependencies, and setting up a basic server application.

Create project

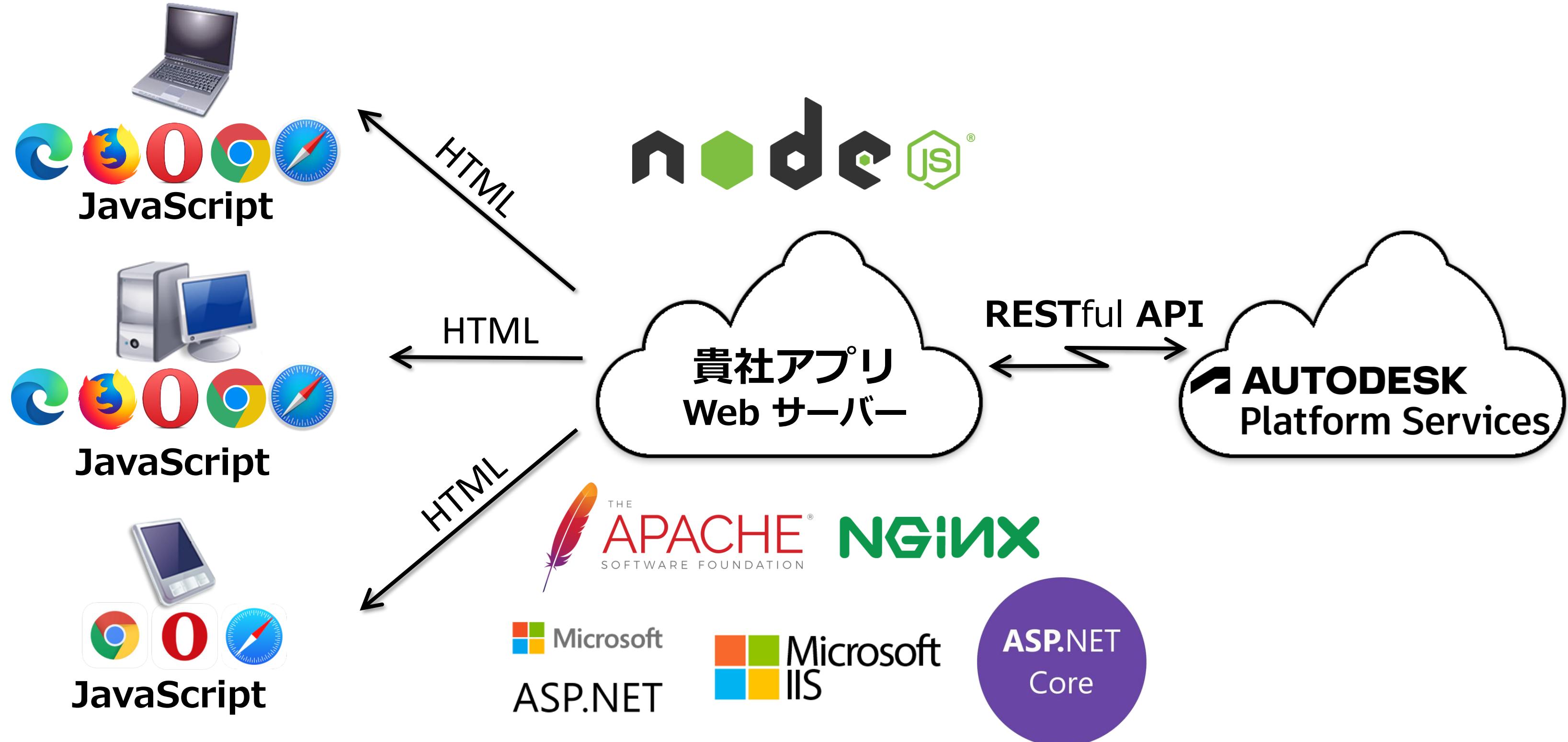
Node.js & VSCode .NET & VSCode .NET & VS2022

Create a new folder for your project, navigate to it in the command line, and initialize a new Node.js project:

```
npm init -y
```

Next install all the Node.js dependencies we're going to use. In this case it will be [datenup](#) (utility for

一般的な Web アプリの構成



Node.js とは



- オープンソース
- JavaScript をサーバー上で実行するための環境
- Google V8 JavaScript Engine が使用
- Node Package Manager で拡張可能
- 今回使用しているパッケージ(ミドルウェア):
 - `@aps_sdk/...`
 - `dotenv`
 - `express`
 - `cookie-session`

チュートリアル: VS Code 上の操作手順

1. プロジェクト フォルダの作成、VS Code でフォルダを開く
2. VS Code のターミナルで Node.js プロジェクトを初期化
3. ターミナルでAPS Node.js SDK パッケージのインストール
4. **package.json** ファイル内 “**dependencies**” の記述追記
5. **wwwroot**、**routes**、**services** サブフォルダの作成
6. **config.js** ファイルの作成、実装
7. **.env** ファイルの作成と Client ID/Secret の指定
8. **.vscode** フォルダ下に **launch.json** ファイルを作成、実装
9. **server.js** ファイルの作成とサーバー実装記述の追加
10. **package.json** ファイル内の “**scripts**” の記述変更
11. デバッグ実行 >> **localhost:8080**

AUTODESK
Platform Services

Solutions ▾ Getting Started Documentation ▾ Success Stories Community ▾ Support ▾ Pricing App Store ▾

Getting Started
Environment Setup
Tutorials
Simple Viewer
Hubs Browser
Application Setup
Authentication
Data Browsing
Viewer & UI
Dashboard
Design Automation
ACC Administrator
ACC Issues (beta)
Learn More

Home > Tutorials > Hubs Browser > Application Setup

Application Setup

Let's start by creating a new project, installing required dependencies, and setting up a basic server application.

Create project

[Node.js & VSCode](#) [.NET & VSCode](#) [.NET & VS2022](#)

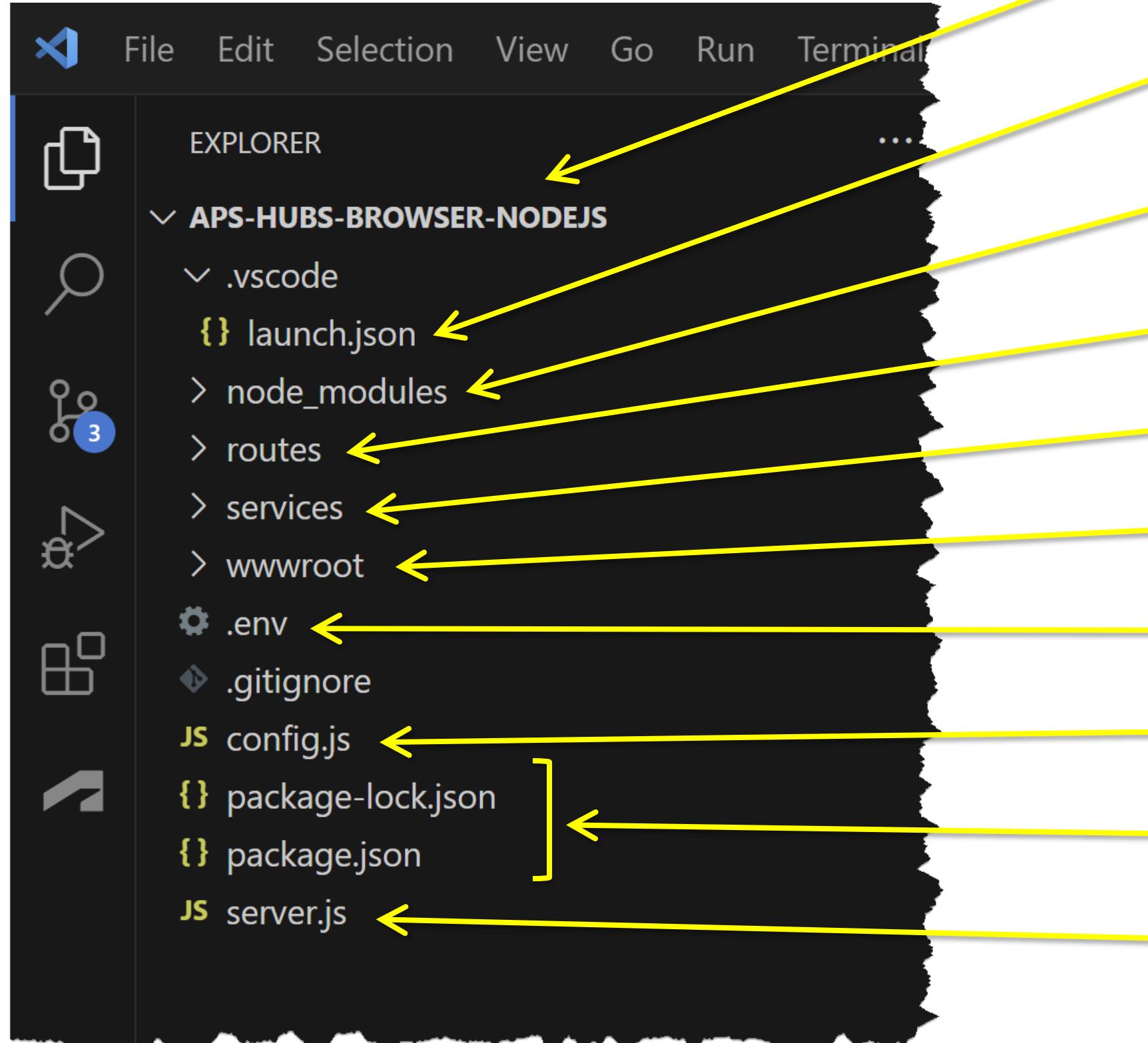
Create a new folder for your project, navigate to it in the command line, and initialize a new Node.js project:

```
npm init -y
```

Next, install all the Node.js dependencies we're going to use. In this case it will be `dotenv` (utility for initializing environment variables from a file), `Express.js` (web framework), and `cookie-session` for handling cookie-based sessions, and finally the required APS SDK modules:

```
npm install --save dotenv express cookie-session
npm install --save @aps_sdk/authentication @aps_sdk/data-management
```

ここまでプロジェクト構成



プロジェクト フォルダを
Hub Browserとした場合

VS Code デバッグで使用

依存関係からインストール
された Node.js パッケージ

ルーティング実装一式

サーバー実装一式

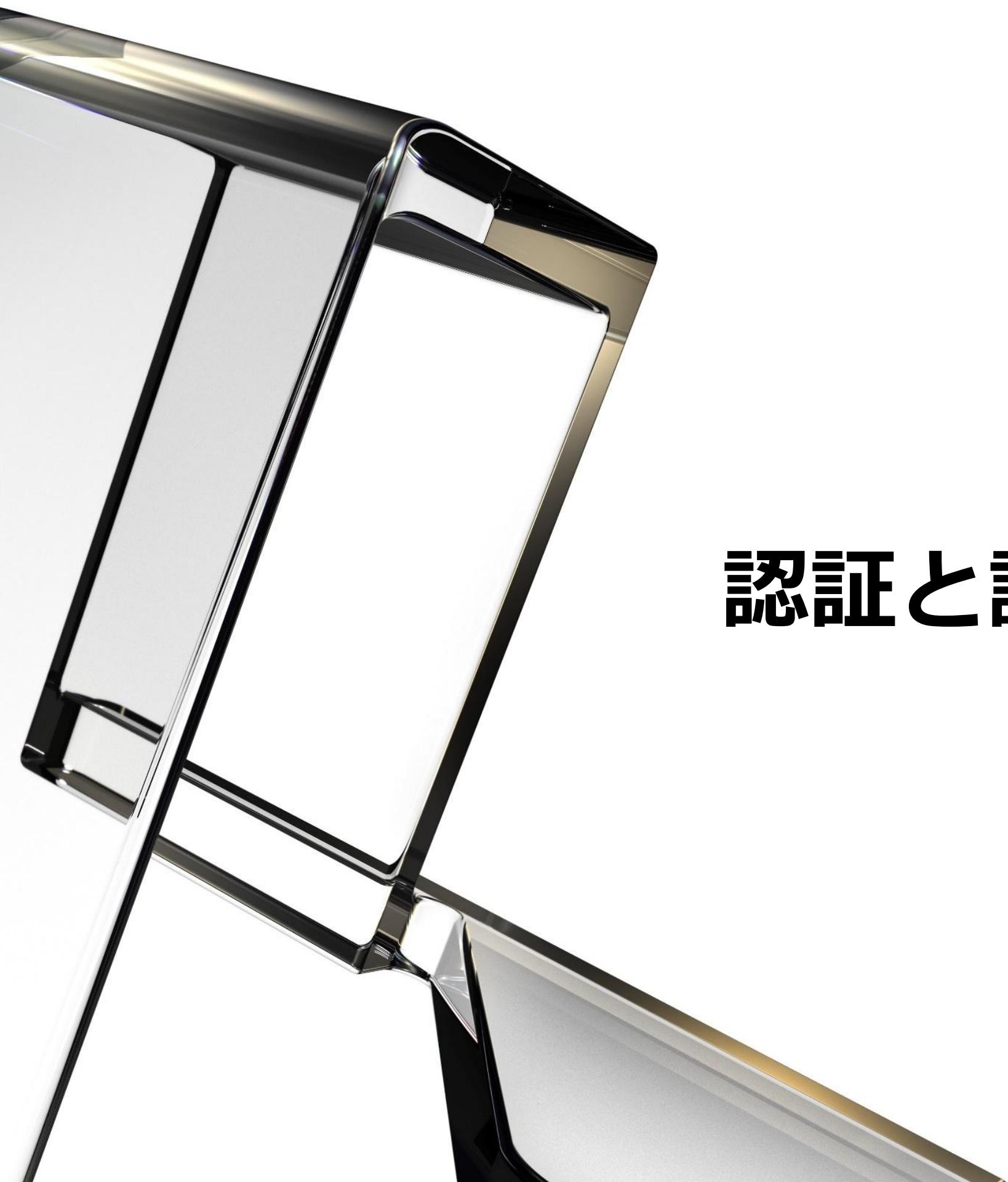
クライアント構成一式

Client ID、Secret を記述

Node.js の起動時情報指定

Node.js パッケージ依存関係

Node.js サーバーの起動構成

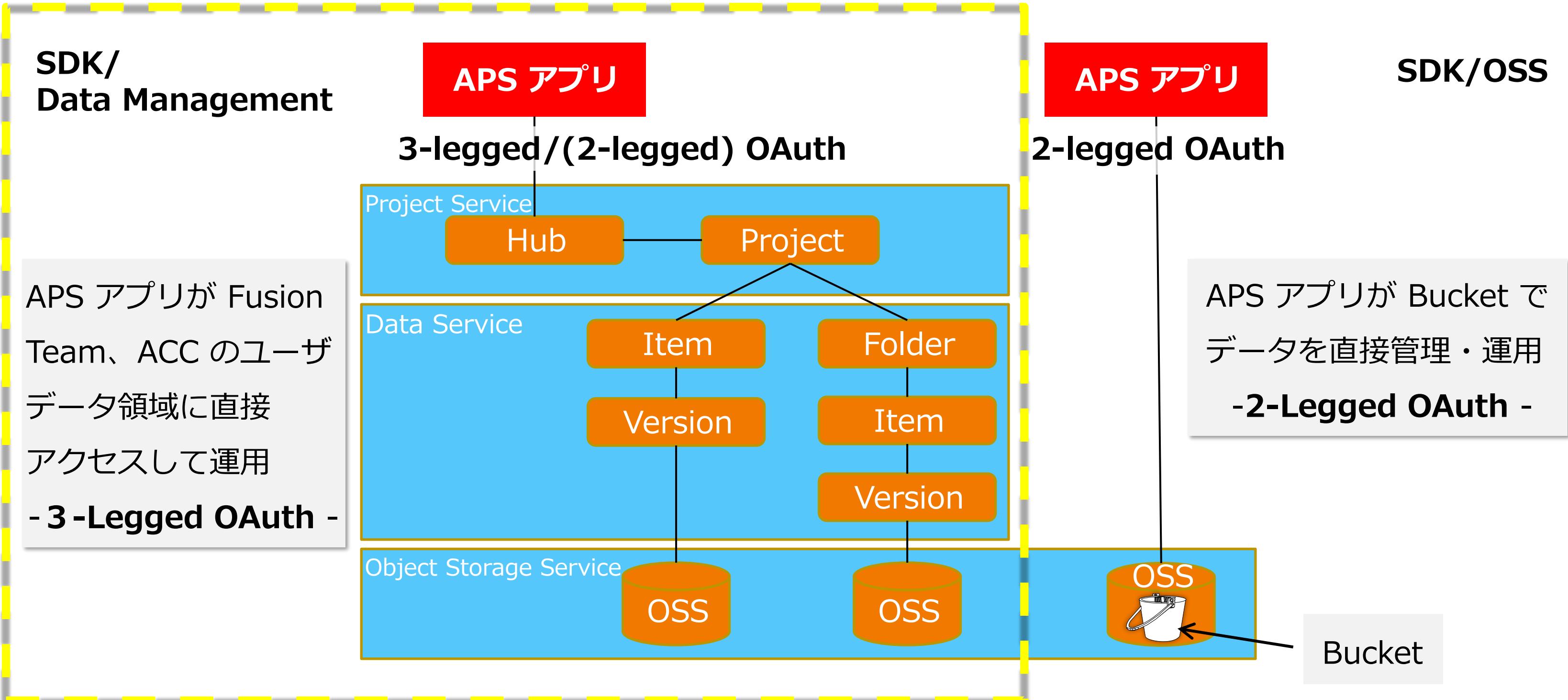


認証と認可

デベロッパキーとアクセストークン



シナリオ/目的に応じた認証フロー



アクセストークン（ Access Token ）とは

- クラウド リソースへのアクセス権限をチェックする仕組み
 - 有効期限が設定される
- 生成に必要なもの
 - デベロッパキー
 - Client ID と Client Secret のペア
 - APS ポータルでアプリ（ App ）作成時に取得可能
 - スコープ（Scope）
 - リソースへのアクセス権限を指定
 - 使用するリソースによって適切に使い分けが必要
 - 複数の Scope 文字列を半角スペースで結合して指定

クラウドサービスに応じた使い分け

- Fusion Team
 - エンドユーザーのデータにアクセスするには、APS アプリに 3-legged 認証フローで得た アクセス トークンをサーバーに提供する必要があります。
- Autodesk Docs (ACC)
 - 事前に Account Admin が Autodesk Construction Cloud (ACC) で「カスタム 統合」を使って APS アプリを登録する必要があります。
 - APS アプリは、3-legged 認証フローまたは 2-legged 認証フローのいずれかで得た アクセス トークンを使用してデータにアクセスできます。

3-legged 認証フロー：エンドユーザによる認可

サインイン

電子メール
name@example.com

次へ

初めてオートデスクをご利用ですか? [アカウントを作成](#)



ようこそ

パスワード
[お忘れですか?](#)

サインイン

サインインを保持



アプリケーションをオーソライズ

Test App さんは、次の情報に対する権限をリクエストしています:

データ
データを表示
データを管理
データを書き込む

許可

オーソライズは不要ですか? [キャンセル](#)



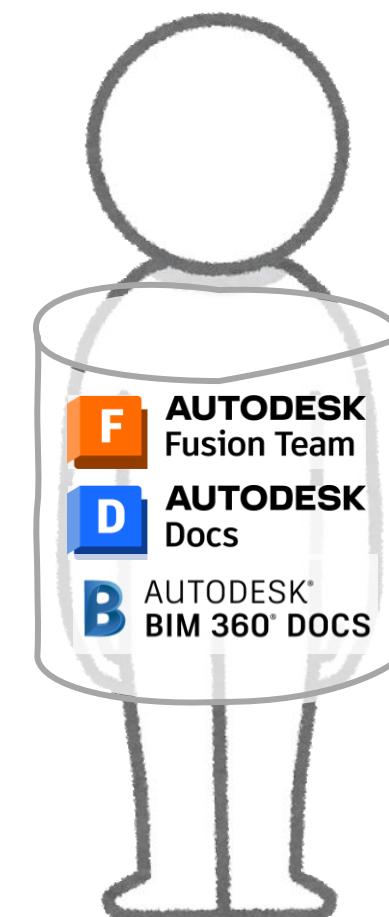
① APS アプリにアクセスすると
デザインデータの所有者である
エンドユーザにサインインを促す
ユーザ名(Autodesk ID)画面に遷移

② 同じくパスワード入力画面に遷移

③ サインインが完了した時点で
APS アプリが指定した Scope に
沿って APS アプリからのアクセス
を認可するか確認する画面に遷移

3-legged 認証フローのイメージ

データリソースのオーナー
(SaaS アプリのユーザー)



APS を利用する Web アプリ



APS サーバー (リソース)

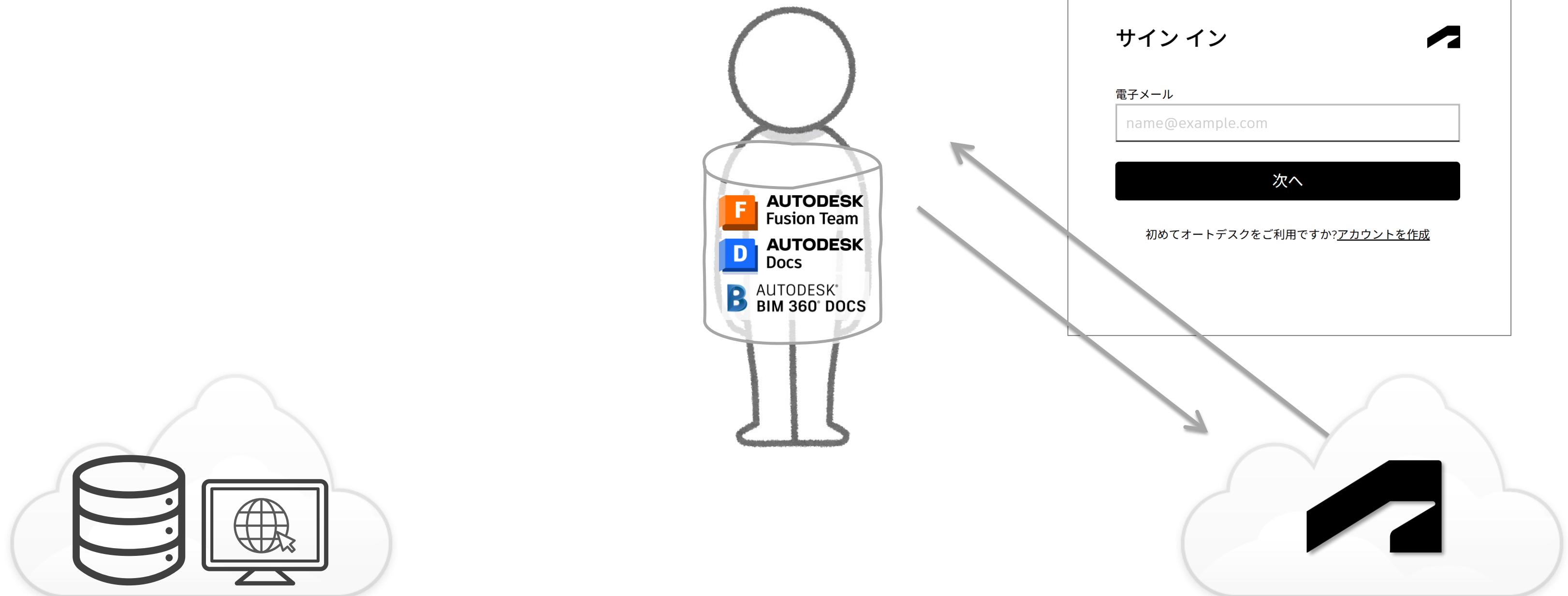
3-legged 認証フローのイメージ

- APS とユーザ データ リソースアクセスをリクエスト



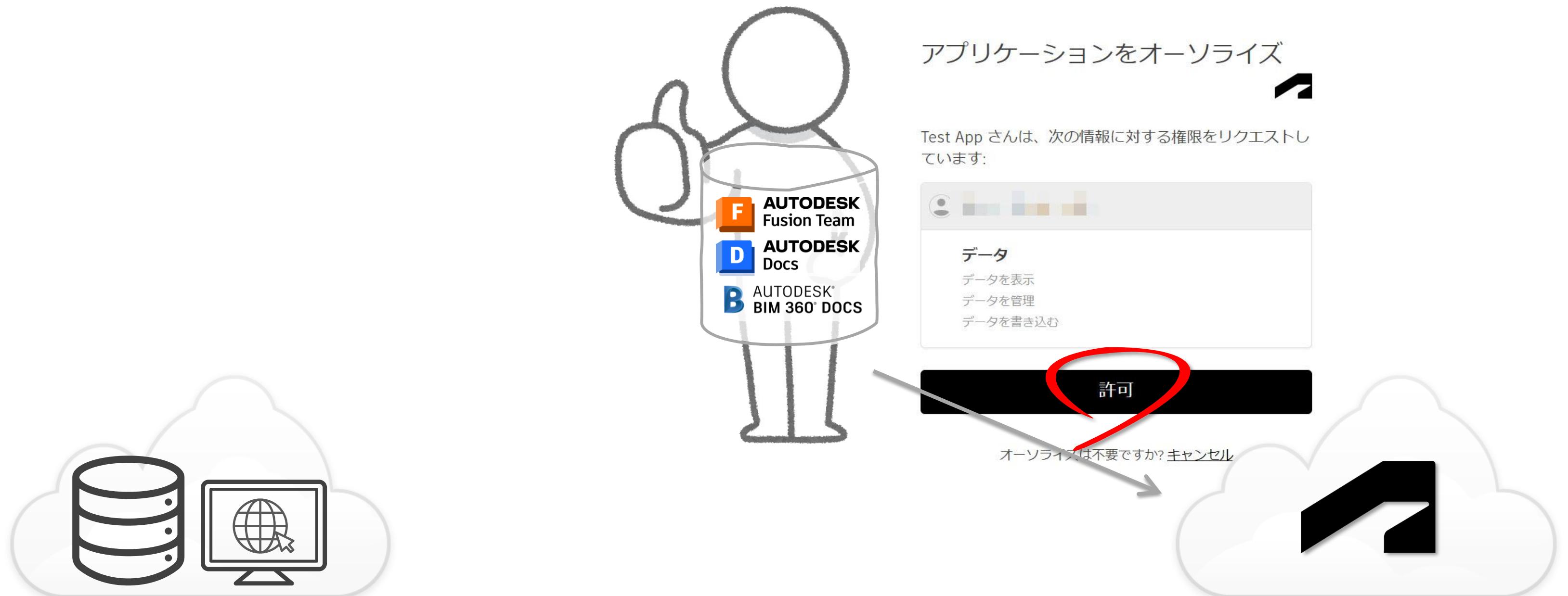
3-legged 認証フローのイメージ

- データリソースオーナーへのログインを求める



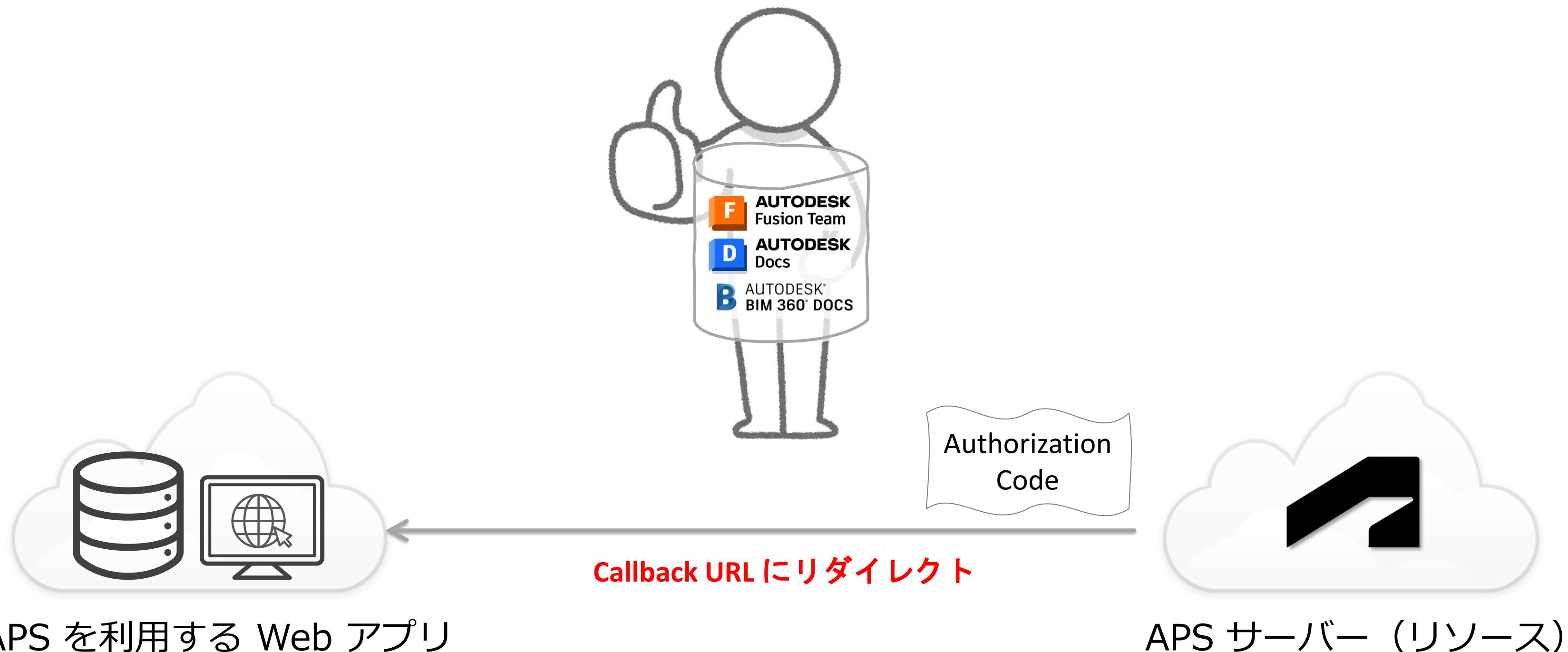
3-legged 認証フローのイメージ

- データオーナーがスコープで指定された範囲でのデータアクセスを認可



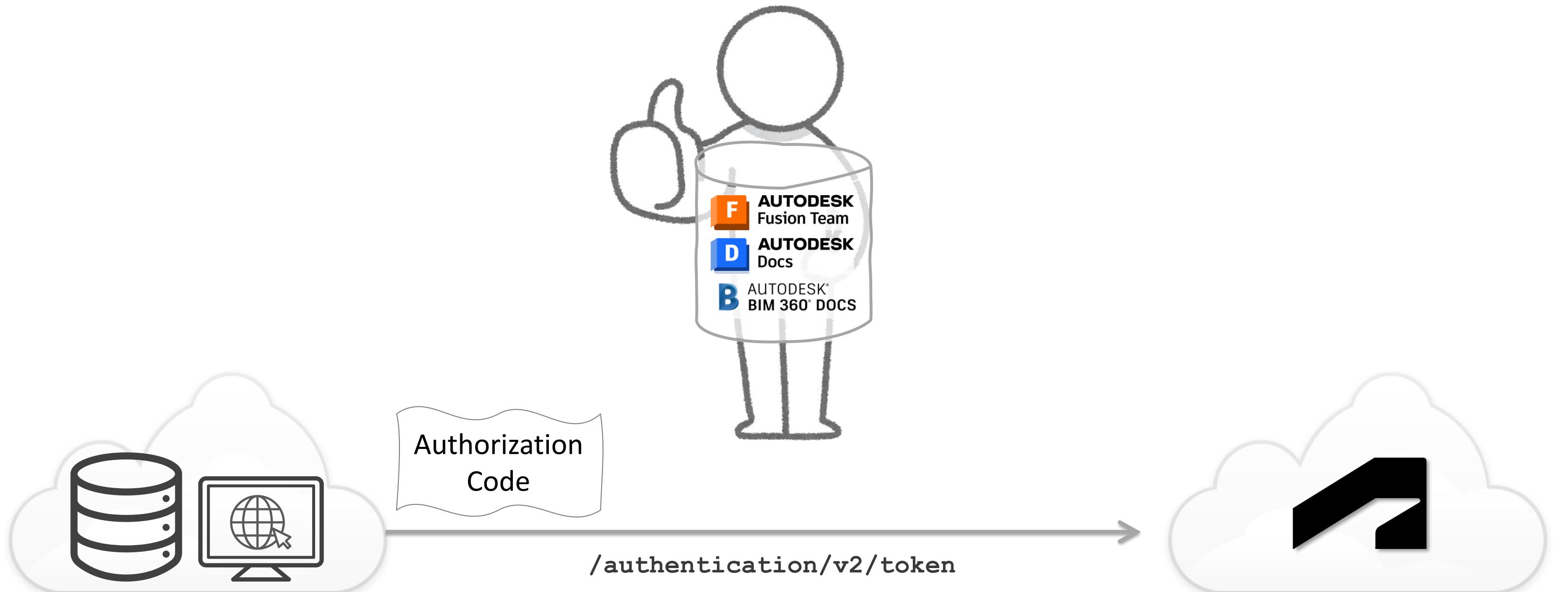
3-legged 認証フローのイメージ

- データオーナーのデータアクセス許可を明示



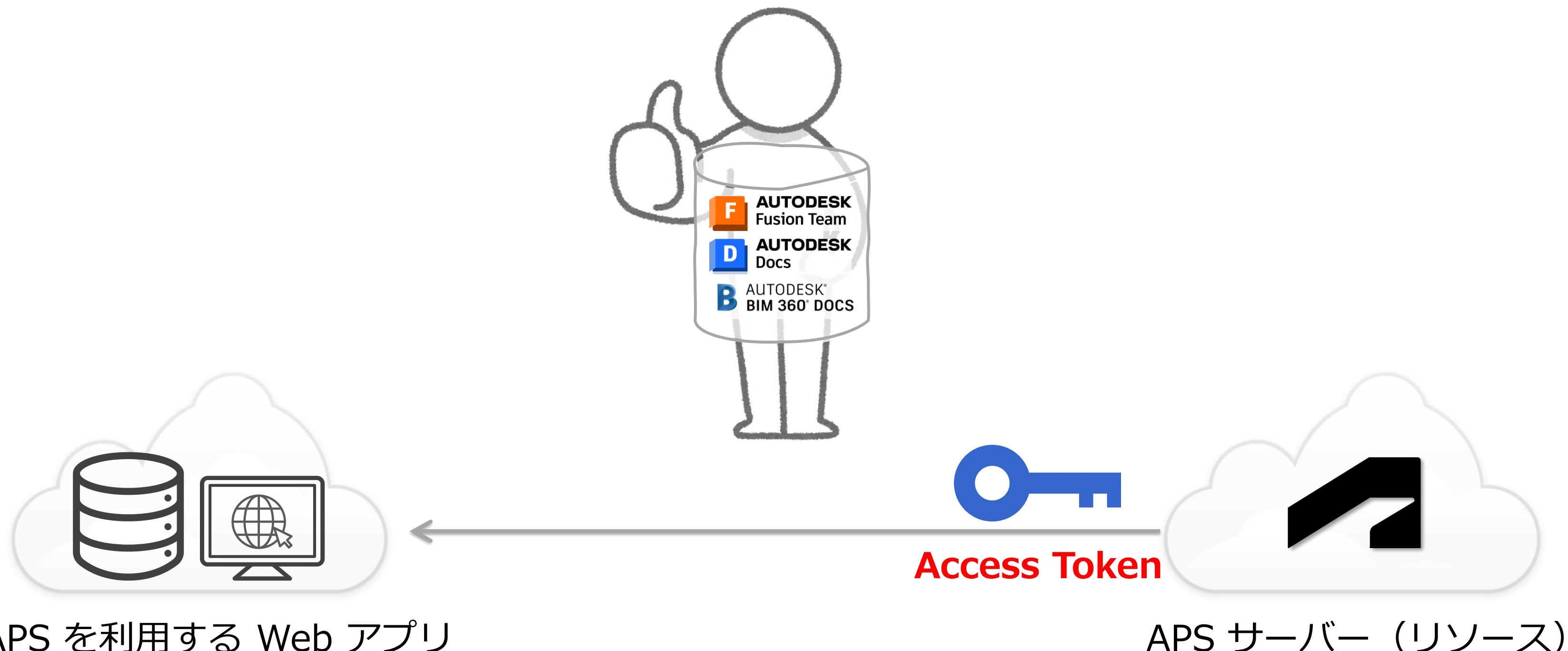
3-legged 認証フローのイメージ

- データオーナーのデータアクセス許可を明示



3-legged 認証フローのイメージ

- APS アプリを認証してアクセストークンを発行



Scope (スコープ) とは

Scope 文字列	意味
user-profile:read	プロファイル (Autodesk ID) の表示
user:read	プロファイル (Autodesk ID) の読み取り
user:write	プロファイル (Autodesk ID) の書き込み
viewables:read	変換後のデザインデータ (SVF) の読み取り (表示)
data:read	ストレージ データの読み取り
data:write	ストレージ データの書き込み (編集)
data:create	ストレージ データの作成
data:search	ストレージ データの検索
bucket:create	新しい Bucket の作成
bucket:read	Bucket の読み取り
bucket:update	Bucket の更新
bucket:delete	Bucket の削除
code:all	コードの生成または実行 (Design Automation API)
account:read	アプリやサービス アカウントの読み取り
account:write	アプリやサービス アカウントの書き込み

適切な OAuth スコープ[°]

- リファレンスで エンドポイント 毎に必要なスコープをチェック！

GET buckets

This endpoint will return the buckets owned by the application. This endpoint supports pagination.

Resource Information

Method and URI	GET https://developer.api.autodesk.com/oss/v2/buckets
Authentication Context	app-only
Required OAuth Scopes	bucket :read
Data Format	JSON

bucket:read

POST buckets

Creates a bucket. Buckets are arbitrary spaces that are created by applications and are used to store objects for later retrieval. A bucket is owned by the application that creates it.

Resource Information

Method and URI	POST https://developer.api.autodesk.com/oss/v2/buckets
Authentication Context	app-only
Required OAuth Scopes	bucket :create
Data Format	JSON

bucket:create

2-legged or 3-legged

- エンドポイント毎に必要とされるコンテキストは異なります。

<p>GET projects/:project_id/items/:item_id</p> <p>Retrieves metadata for a specified item. Items represent word documents.</p> <p>Notes:</p> <ul style="list-style-type: none"> The tip version for the item is included in the <code>included</code> array. To retrieve metadata for multiple items, see the ListItems command. <p>New! Autodesk Construction Cloud platform (ACC). Note that this endpoint is part of the Autodesk Construction Cloud APIs, see the Autodesk Construction Cloud documentation.</p> <p>Resource Information</p> <table border="1"> <tbody> <tr> <td>Method and URI</td> <td>GET https://developer.api.autodesk.com/oss/v2/projects/:project_id/items/:item_id</td> </tr> <tr> <td>Authentication Context</td> <td>user context optional</td> </tr> <tr> <td>Required OAuth Scopes</td> <td>data:read</td> </tr> <tr> <td colspan="2">Data Format</td> </tr> </tbody> </table>	Method and URI	GET https://developer.api.autodesk.com/oss/v2/projects/:project_id/items/:item_id	Authentication Context	user context optional	Required OAuth Scopes	data:read	Data Format		<p>GET users/me</p> <p>Retrieves the profile information about an end user, including the user's name, email address, and profile picture. This endpoint does not provide full control over the user's account, such as <code>create</code>, <code>update</code>, or <code>full control</code> permissions.</p> <p>New! Autodesk Construction Cloud platform (ACC). Note that this endpoint is part of the Autodesk Construction Cloud APIs, see the Autodesk Construction Cloud documentation.</p> <p>Resource Information</p> <table border="1"> <tbody> <tr> <td>Method and URI</td> <td>GET https://developer.api.autodesk.com/oss/v2/users/me</td> </tr> <tr> <td>Authentication Context</td> <td>user context required</td> </tr> <tr> <td>Required OAuth Scopes</td> <td>account:read</td> </tr> <tr> <td colspan="2">Data Format</td> </tr> </tbody> </table>	Method and URI	GET https://developer.api.autodesk.com/oss/v2/users/me	Authentication Context	user context required	Required OAuth Scopes	account:read	Data Format		<p>GET buckets/:bucketKey/objects</p> <p>List objects in a bucket. It is only available to the bucket creator.</p> <p>Resource Information</p> <table border="1"> <tbody> <tr> <td>Method and URI</td> <td>GET https://developer.api.autodesk.com/oss/v2/buckets/:bucketKey/objects</td> </tr> <tr> <td>Authentication Context</td> <td>app-only</td> </tr> <tr> <td>Required OAuth Scopes</td> <td>data:read</td> </tr> <tr> <td colspan="2">Data Format</td> </tr> </tbody> </table>	Method and URI	GET https://developer.api.autodesk.com/oss/v2/buckets/:bucketKey/objects	Authentication Context	app-only	Required OAuth Scopes	data:read	Data Format	
Method and URI	GET https://developer.api.autodesk.com/oss/v2/projects/:project_id/items/:item_id																									
Authentication Context	user context optional																									
Required OAuth Scopes	data:read																									
Data Format																										
Method and URI	GET https://developer.api.autodesk.com/oss/v2/users/me																									
Authentication Context	user context required																									
Required OAuth Scopes	account:read																									
Data Format																										
Method and URI	GET https://developer.api.autodesk.com/oss/v2/buckets/:bucketKey/objects																									
Authentication Context	app-only																									
Required OAuth Scopes	data:read																									
Data Format																										

2-legged 認証とユーザー情報の組み合わせ

POST projects/:project_id/items

Headers

Authorization * Must be `Bearer <token>`, where `<token>` is obtained via either a [two-legged](#) or [three-legged](#) OAuth flow.
string

Content-Type * Must be `application/vnd.api+json`
string

x-user-id In a two-legged authentication context, the app has access to all users specified by the administrator in the SaaS integrations UI. By providing this header, the API call will be limited to act on behalf of only the user specified.
string

Note that for a three-legged OAuth flow or for a two-legged OAuth flow with user impersonation (`x-user-id`), the user must have permission to upload files to the specified parent folder (`data.attributes.relationships.parent.data.id`).

例) アプリが 2-legged 認証フローでアップロード

- アップロードしたユーザーは誰?

**2-legged Token 使用時で
x-user-id リクエストヘッダー未指定時**

名前	最終更新日	更新者	バージョンの追加者
result.pdf	2024年11月18日 17:31	Aシ ACC システム	Aシ ACC システム
result2.pdf	2024年11月18日 18:00	Toshiaki Isezaki Developer Advocacy ...	Toshiaki Isezaki Developer Advocacy ...

**3-legged Token 使用時、または
2-legged Token 使用時で
x-user-id リクエストヘッダー指定時**

ACC 統合アプリの運用のまとめ

- プロジェクト データのオーナーは「ユーザー」
- 原則、3-legged 認証フローの利用を推奨

認証フロー	ACC	Fusion Team
2-legged 認証	○	×
2-legged 認証 x-user-id	○	×
3-legged 認証	○	○

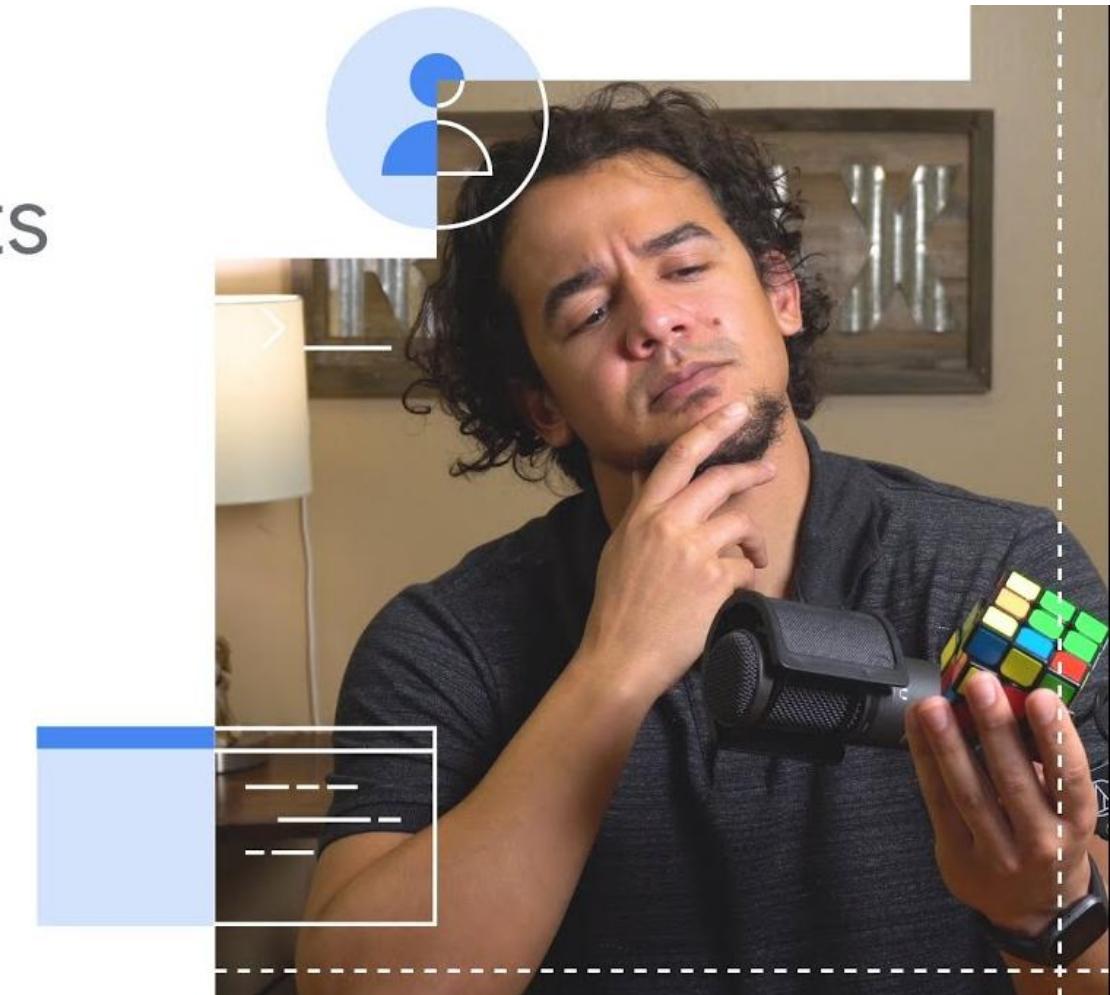
- x-user-id は [GET OIDC User Info](#) sub フィールド値
- 2-legged 認証フローはアプリケーション コンテキスト時のみ

Secure Services Accounts (セキュア サービス アカウント)

- 「スコープ付き」 サーバー間統合の重要性を理解
- 「時々強力すぎる」 2-Legged の必要性を低減/排除を目的
- 「3-legged 認証」 の必要な API でもセキュアに統合可能に

Service Accounts
Basics

https://adndevblog.typepad.com/technology_perspective/2025/04/introducing-secure-service-accounts-sa-now-in-public-beta.html



チュートリアル：認証

The screenshot shows a web browser displaying the Autodesk Platform Services Documentation website. The URL is [https://developer.autodesk.com/tutorials/hubs-browser/authentication](#). The page title is "Authentication". The main content area contains a tip about generating tokens and instructions for creating an `aps.js` file.

Sidebar (Left):

- Getting Started
- Environment Setup
- Tutorials**
 - Simple Viewer
 - Hubs Browser**
 - Application Setup
 - Authentication**
 - Data Browsing
 - Viewer & UI
 - Dashboard
 - Design Automation
 - ACC Administrator
 - ACC Issues (beta)
 - Learn More

Header:

- AUTODESK Platform Services
- Sign in

Breadcrumbs:

- Home > Tutorials > Hubs Browser > Authentication

Section Header:

Authentication

In this step we're going to extend the server implementation so that it can authenticate itself to the APS platform, guide the user through a [3-legged OAuth workflow](#), and generate access tokens for various needs.

TIP

It is a good practice to generate an "internal" token with more capabilities (for example, allowing the owner to create or delete files in the Data Management service) that will only be used by the server, and a "public" token with fewer capabilities that can be safely shared with the client-side logic.

Access tokens

Node.js & VSCode .NET & VSCode .NET & VS2022

Create an `aps.js` file under the `services` folder. This is where we will be implementing all the APS logic that will be used in different areas of our server application. Let's start by adding the following code to the file:

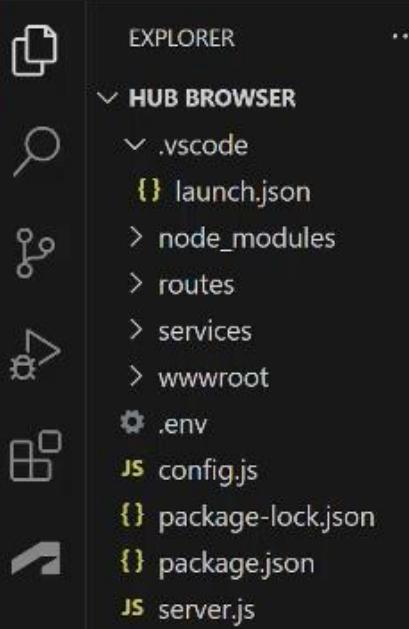
```
services/aps.js
const { AuthenticationClient, ResponseType } = require('@aps_sdk/authentication');
```

チュートリアル： VS Code 上の操作手順

1. services フォルダ下に aps.js ファイルを作成して実装
2. routes フォルダ下に auth.js ファイルを作成して実装
3. server.js ファイルに auth.js をマウント
4. デバッグ実行 >> <http://localhost:8080/api/auth/login>

The screenshot shows the Chrome DevTools Application tab interface. On the left, there's a sidebar with sections for Manifest, Service workers, and Storage (Local storage, Session storage, Extension storage, IndexedDB). Below that is a expanded section for Cookies under the http://localhost:8080 domain. The main area displays a table of cookies:

Name	Value	Domain	Path	Expires	Size
session	eyJwdWJsaWNfdG9rZW4iOiJleUpoYkdjaU9...	localhost	/	2025-...	230K
session.sig	fxRj50Gbql3UQVaDV88qcjfXvU	localhost	/	2025-...	36



Show All Commands Ctrl + Shift + P

Go to File Ctrl + P

Open Chat Ctrl + Alt + I

Find in Files Ctrl + Shift + F

Toggle Terminal Ctrl + @

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

powershell + ⌂ ⌂ ⌂ ⌂ ⌂

15 packages are looking for funding
run `npm fund` for details

found 0 vulnerabilities

● PS C:\Users\ogasawr\Hub Browser> npm install --save @aps_sdk/authentication @aps_sdk/data-management

added 44 packages, and audited 122 packages in 10s

18 packages are looking for funding
run `npm fund` for details

found 0 vulnerabilities

○ PS C:\Users\ogasawr\Hub Browser>



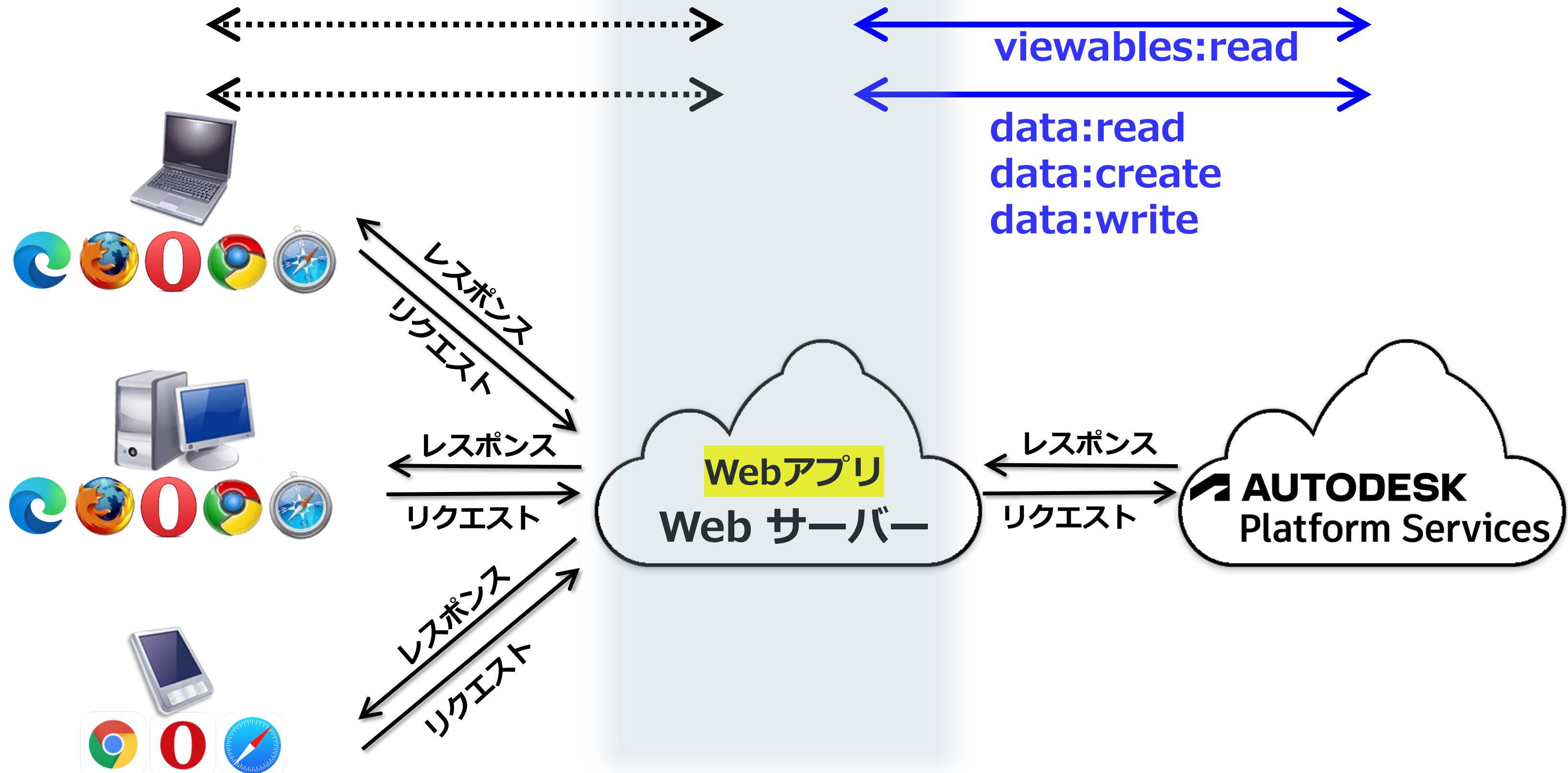
> OUTLINE

> TIMELINE

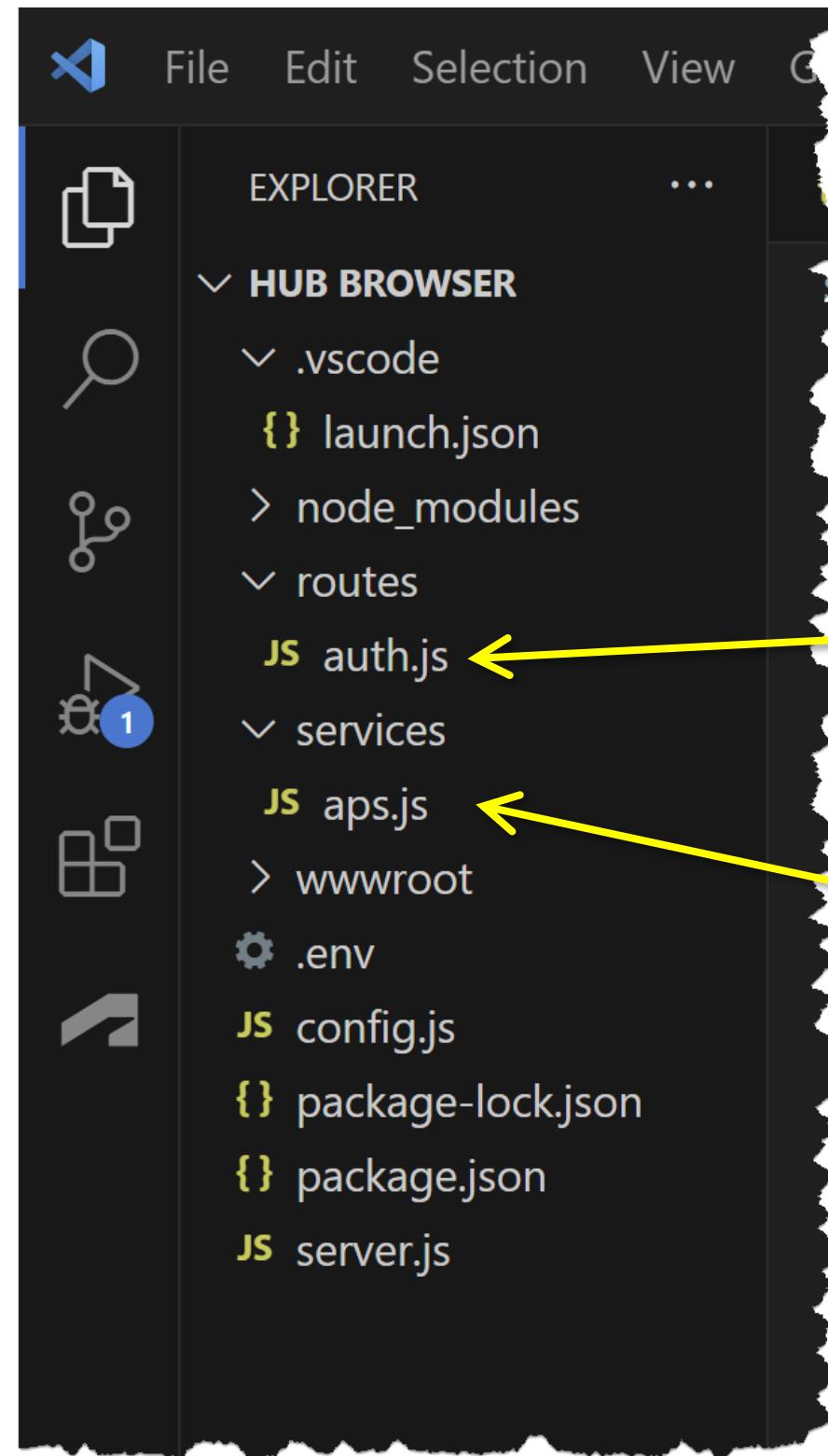
× 0 △ 0 ↗ Launch Server (Hub Browser)



スコープによるトークンを使い分け



ここまでプロジェクト構成

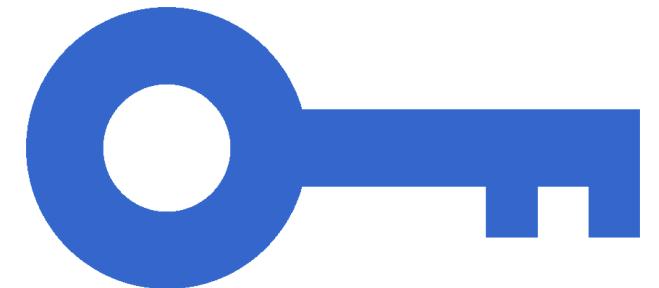


クライアントが呼ぶ出す
エンドポイントを実装
(Client ID/Secret を隠蔽し
てアクセストークンを取得)

サーバー実装からアクセス
トークンを取得する関数定義

その他、3-legged 認証単体の評価

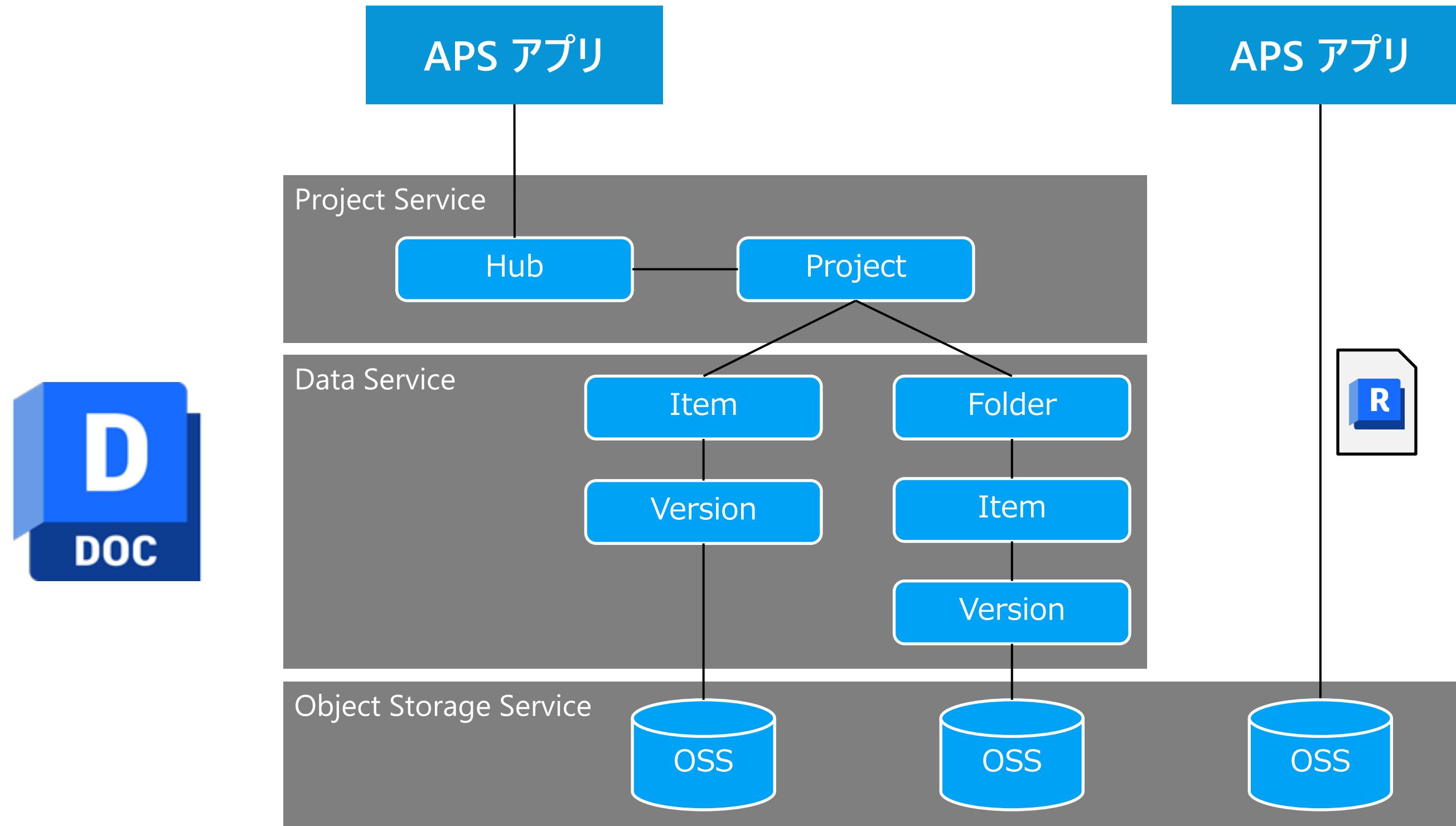
- アクセストークンの取得
- Postman
 - Postman を使った 3-legged 認証・認可(OAuth v2)
- VS Code
 - 新 Node.js SDK と VS Code で 3-legged アクセストークンを取得



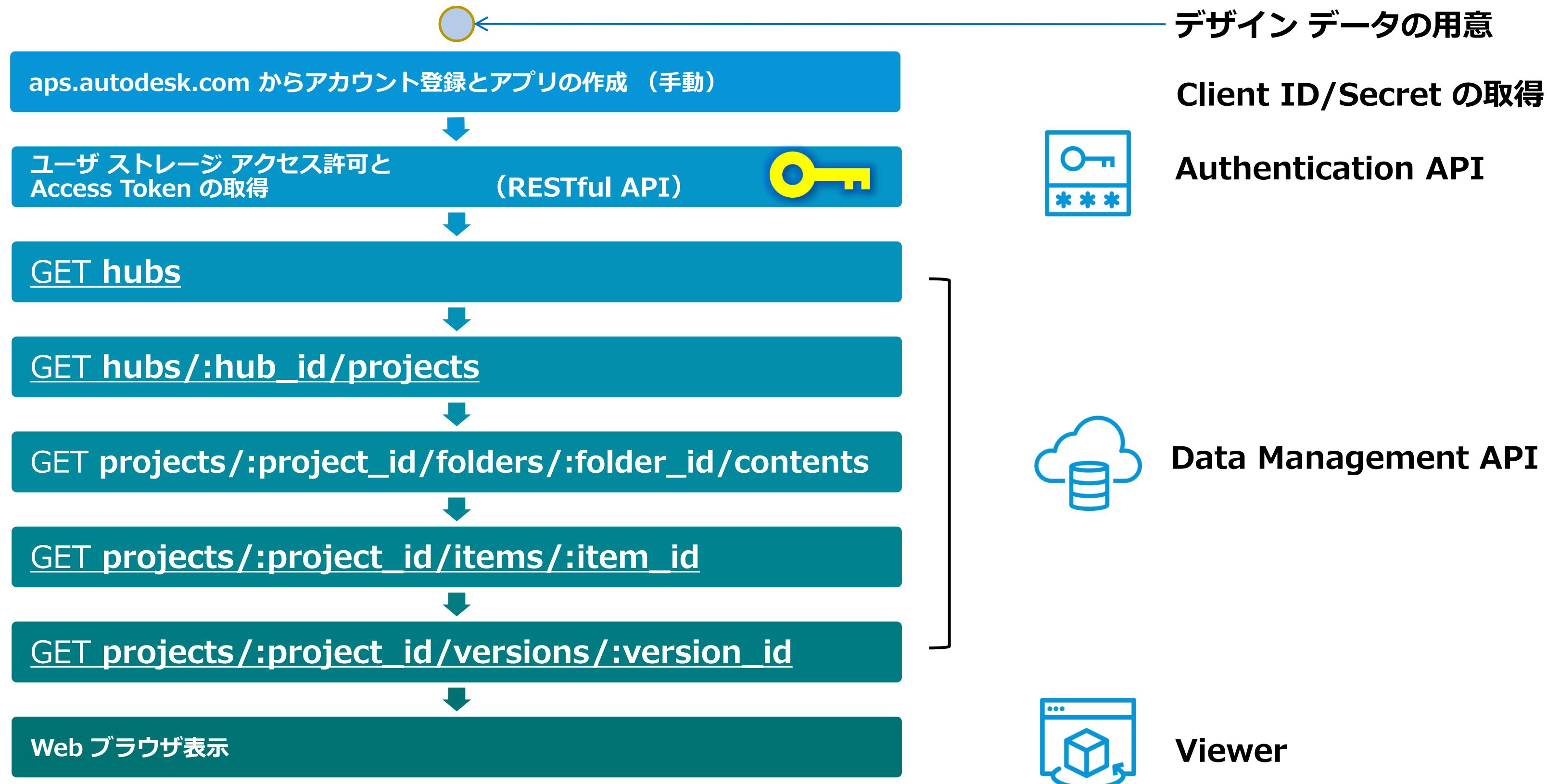


データのブラウジング

シナリオ/目的に応じた使い分けが必要



3-legged OAuthでのViewer利用手順の理解



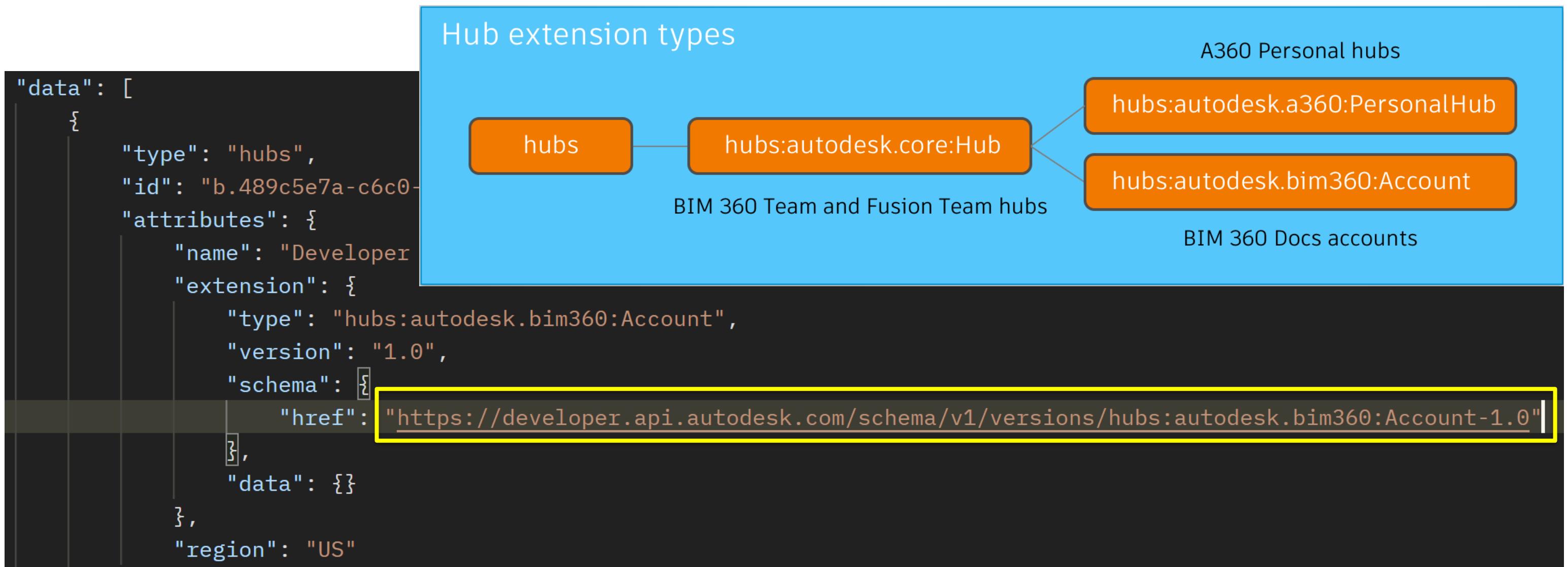
Hubs – ACC/BIM 360 とそれ以外の Hub の判別

- “Hubs” は、Autodesk Docs/BIM 360 Docs の “Account” に相当。
- Autodesk Docs/BIM 360 Docs の Hub と Project の Id は、b.xxx で始まり、それ以外のプロダクトは a.xxx で始まります。

```
"data": [
  {
    "type": "hubs",
    "id": "b.489c5e7a-", ",",
    "attributes": {
      "name": "Developer Advocacy Support",
      "extension": {
        "type": "hubs:autodesk.bim360:Account",
        "version": "1.0",
        "schema": {
          "href": "https://developer.api.autodesk.com/schema/v1/versions/hubs:autodesk.bim360:Account-1.0"
        },
        "data": {}
      },
      "region": "US"
    }
  }
]
```

Hubs – タイプの違い

- *:autodesk.core:*ではなく、*:autodesk.bim360:* の拡張タイプを使用
詳細: https://aps.autodesk.com/en/docs/data/v2/developers_guide/basics/



Projects – ACC/BIM 360 の判別

- プロジェクトが BIM 360 か ACC か判別するには、“projectType” 属性を参照します。

GET

hubs/:hub_id/projects

```
"type": "projects",
"id": "b.4fae[REDACTED]",
"attributes": {
    "name": "Ryuji ACC Test Project",
    "scopes": [
        "b360project.4fae[REDACTED]",
        "02tenant.[REDACTED]"
    ],
    "extension": {
        "type": "projects:autodesk.bim360:Project",
        "version": "1.0",
        "schema": {
            "href": "https://developer.api.autodesk.com/schema/v1/versions/projects:autodesk.bim360:Project-1.0"
        },
        "data": {
            "projectType": "ACC"
        }
    }
},
```

data.attributes.extension.data.projectType

ACC | BIM360

Folders

- [プロジェクトファイル] フォルダが起点

The screenshot shows the Autodesk Construction Cloud interface for the 'Ryuji ACC 2023 Test Project'. The left sidebar includes 'Docs' (selected), 'Reviews', 'File Transfer', 'Matters', 'Reports', 'Members', 'Bridges', and 'Settings'. The main area is titled 'Files' and shows a tree view of folders: 'Project File' (highlighted with a yellow box), 'Shared', 'Data Exchange', 'In Progress', 'Conflict Check Test...', 'Architect', 'Structure', and 'Equipment'. A blue button 'Upload File' is visible. To the right is a table listing files: 'Consumed' (version --, size --, updated 2023/10/15 16:39 by Ryuji Ogasawara) and 'サンプル意匠.rvt' (version V8, size 117.4 MB, updated 2023/10/18 23:21 by Ryuji Ogasawara). The interface includes search, filter, and sort functions.

名前	バージョン	インジケータ	サイズ	最終更新日	更新者	マークア
Consumed	--	--	--	2023年10月15日 16:39	Ryuji Ogasawara	⋮
サンプル意匠.rvt	V8	...	117.4 MB	2023年10月18日 23:21	Ryuji Ogasawara	⋮

Folders - “Project Files” Folder Id の取得

- SDK の場合、下記のメソッドで簡単に[プロジェクト ファイル]フォルダの ID を取得できます。

```
dataManagementClient.getProjectTopFolders(hubId, projectId, {accessToken});
```

- REST API の場合、

GET hubs/:hub_id/projects/:project_id/topFolders

- レスポンスデータの中から name : “Project Files” のフォルダを探す

GET projects/:project_id/folders/:folder_id/contents

- rootFolder Id を指定して、その配下のコンテンツのデータから探す

Projects – プロジェクトのルートフォルダの取得

GET hubs/:hub_id/projects

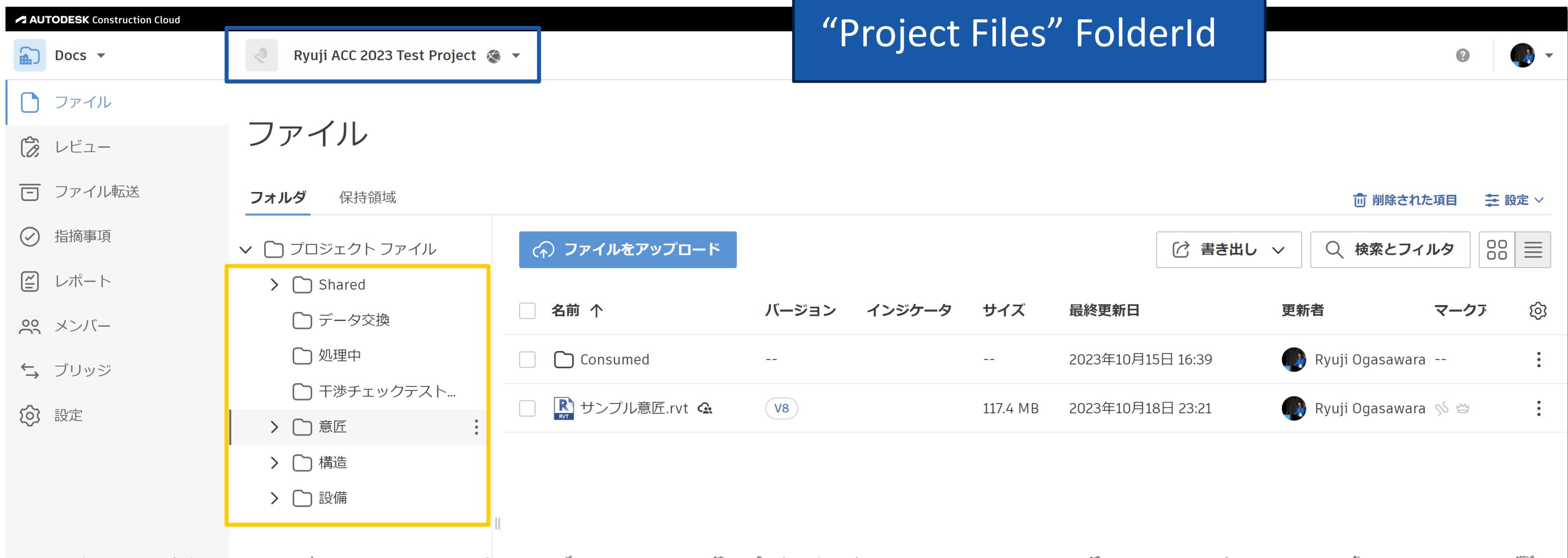
```
"relationships": {  
    "hub": {  
        "data": {  
            "type": "hubs",  
            "id": "b.489c5[REDACTED]"  
        },  
        "links": {  
            "related": {  
                "href": "https://developer.api.autodesk.com/project/v1/hubs/b.489c5[REDACTED]"  
            }  
        }  
    }  
},  
"rootFolder": {  
    "data": {  
        "type": "folders",  
        "id": "urn:adsk.wipprod:fs.folder:co.5N7Me[REDACTED]"  
    },  
    "meta": {  
        "link": {  
            "href": "https://developer.api.autodesk.com/data/v1/projects/b.4faef[REDACTED]/folders/[REDACTED]  
            urn:adsk.wipprod:fs.folder:co.5N7Me[REDACTED]"  
        }  
    }  
},  
}
```

レスポンスの relationships 配下のデータから、
プロジェクトのルートフォルダの ID を取得できます。

Folders - “Project Files” フォルダ配下のコンテンツ

GET projects/:project_id/folders/:folder_id/contents

- 直下のみ。再帰処理が必要です。



The screenshot shows the Autodesk Construction Cloud interface for the 'Ryuji ACC 2023 Test Project'. A large blue callout box highlights the 'Project Files' folder ID. The left sidebar shows navigation options like Docs, Review, File Transfer, Tasks, Reports, Members, Bridging, and Settings. The main area is titled 'ファイル' (File) and shows a list of files and folders under the 'Project Files' folder. The 'Project Files' folder itself is expanded, showing sub-folders 'Shared', 'Data Exchange', 'In Progress', and '干涉チェックテスト...', and a single item '意匠' (Design). A yellow box highlights the '意匠' folder. To the right, there's a file upload button 'ファイルをアップロード', and a table listing files with columns for Name, Version, Indicator, Size, Last Updated, and Author. Two files are listed: 'Consumed' (version --, size --, last updated 2023年10月15日 16:39 by Ryuji Ogasawara) and 'サンプル意匠.rvt' (version v8, size 117.4 MB, last updated 2023年10月18日 23:21 by Ryuji Ogasawara).

“Project Files” FolderId

名前	バージョン	インジケータ	サイズ	最終更新日	更新者	マークア
Consumed	--	--	--	2023年10月15日 16:39	Ryuji Ogasawara	
サンプル意匠.rvt	v8		117.4 MB	2023年10月18日 23:21	Ryuji Ogasawara	

Folder と Item の識別、Version Id の取得

GET projects/:project_id/folders/:folder_id/contents

type 属性で識別

- folders
- items

名前	バージョン	インジケータ	サイズ	最終更新日	更新者	マークア
Consumed	--		--	2023年10月15日 16:39	Ryuji Ogasawara	
サンプル意匠.rvt	V8				asawara	

Items の included 配下に type : versions のデータも

Items と Versions

- 詳細な情報は、Items / Versions の各エンドポイントで取得可能。

The screenshot shows the Autodesk Construction Cloud interface for the 'Ryuji ACC 2023 Test Project'. The left sidebar includes 'Docs', 'Reviews', 'File Transfer', 'Checklist', 'Reports', 'Members', 'Bridges', and 'Settings'. The main area is titled 'Files' and displays a list of files. A blue button 'Upload File' is visible. The table has columns for Name, Version, Indicator, Size, Last Updated, Updater, and Mark. Two items are listed:

名前	バージョン	インジケータ	サイズ	最終更新日	更新者	マークア
Consumed	--	--	--	2023年10月15日 16:39	Ryuji Ogasawara	...
サンプル意匠.rvt	V8	Cloud	117.4 MB	2023年10月18日 23:21	Ryuji Ogasawara	...

データのブラウジング

AUTODESK Platform Services

Solutions ▾ Getting Started Documentation ▾ Success Stories Community ▾ Support ▾ Pricing App Store ▾

Getting Started
Environment Setup
Tutorials
Simple Viewer
Hubs Browser
Application Setup
Authentication
Data Browsing
Viewer & UI
Dashboard
Design Automation
ACC Administrator
ACC Issues (beta)
Learn More

Home > Tutorials > Hubs Browser > Data Browsing

Data Browsing

In this step we'll extend our server so that we can browse the content of other APS-based applications such as BIM 360 Docs or Autodesk Docs. We will basically follow the Data Management service's hierarchy of hubs, projects, folders, items, and versions:

```

graph TD
    subgraph ProjectService [Project Service]
        Hub[Hub]
        Project[Project]
        Hub --- Project
    end
    subgraph DataService [Data Service]
        Item[Item]
        Folder[Folder]
        Version1[Version]
        Version2[Version]
        Item --- Version1
        Folder --- Version2
    end
    Project --- Item
    Project --- Folder
  
```

Browsing hubs
Server endpoints
Try it out

チュートリアル: VS Code 上の操作手順

1. **services** フォルダ下の **aps.js** ファイルに実装を追加
 - Data Management API
2. **routes** フォルダ下に **hubs.js** ファイルを追加して実装
3. **server.js** ファイルに **hubs.js** をマウント
4. デバッグ実行 >> <http://localhost:8080/api/hubs>

The screenshot shows the Visual Studio Code interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Search Bar:** Hub Browser.
- Explorer Sidebar:** Shows the project structure under "HUB BROWSER".
- Current File:** aps.js (highlighted in yellow).
- Code Content:** The code in aps.js handles authentication middleware and user profile retrieval.

```
services > aps.js > ...
26     service.authRefreshMiddleware = async (req, res, next) => {
27         ...
28         const publicCredentials = await authenticationClient.refreshToken(internalCredentials.refresh_token, APS_CLIENT_ID, {
29             clientSecret: APS_CLIENT_SECRET,
30             scopes: PUBLIC_TOKEN_SCOPES
31         });
32         req.session.public_token = publicCredentials.access_token;
33         req.session.internal_token = internalCredentials.access_token;
34         req.session.refresh_token = publicCredentials.refresh_token;
35         req.session.expires_at = Date.now() + internalCredentials.expires_in * 1000;
36     }
37     req.internalOAuthToken = {
38         access_token: req.session.internal_token,
39         expires_in: Math.round((req.session.expires_at - Date.now()) / 1000),
40     };
41     req.publicOAuthToken = {
42         access_token: req.session.public_token,
43         expires_in: Math.round((req.session.expires_at - Date.now()) / 1000),
44     };
45     next();
46 }
47
48 service.getUserProfile = async (accessToken) => {
49     const resp = await authenticationClient.getUserInfo(accessToken);
50     return resp;
51 }
52
53
54
55
56
57
58
59
60
61 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
15 packages are looking for funding
  run `npm fund` for details

  found 0 vulnerabilities
● PS C:\Users\ogasawr\Hub Browser> npm install --save @aps_sdk/authentication @aps_sdk/data-management

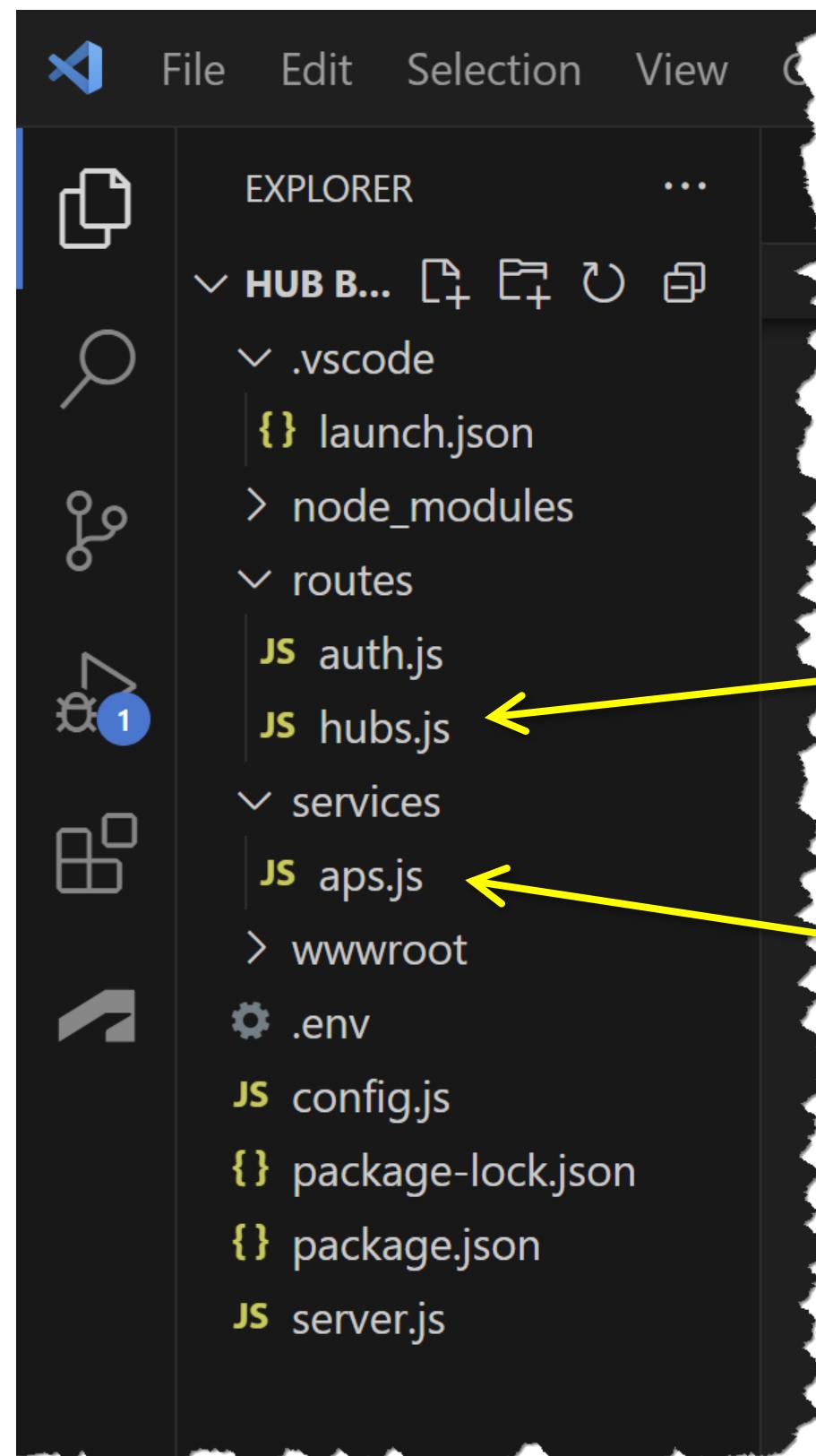
  added 44 packages, and audited 122 packages in 10s
```

```
18 packages are looking for funding
  run `npm fund` for details
```

```
found 0 vulnerabilities
○ PS C:\Users\ogasawr\Hub Browser>
```



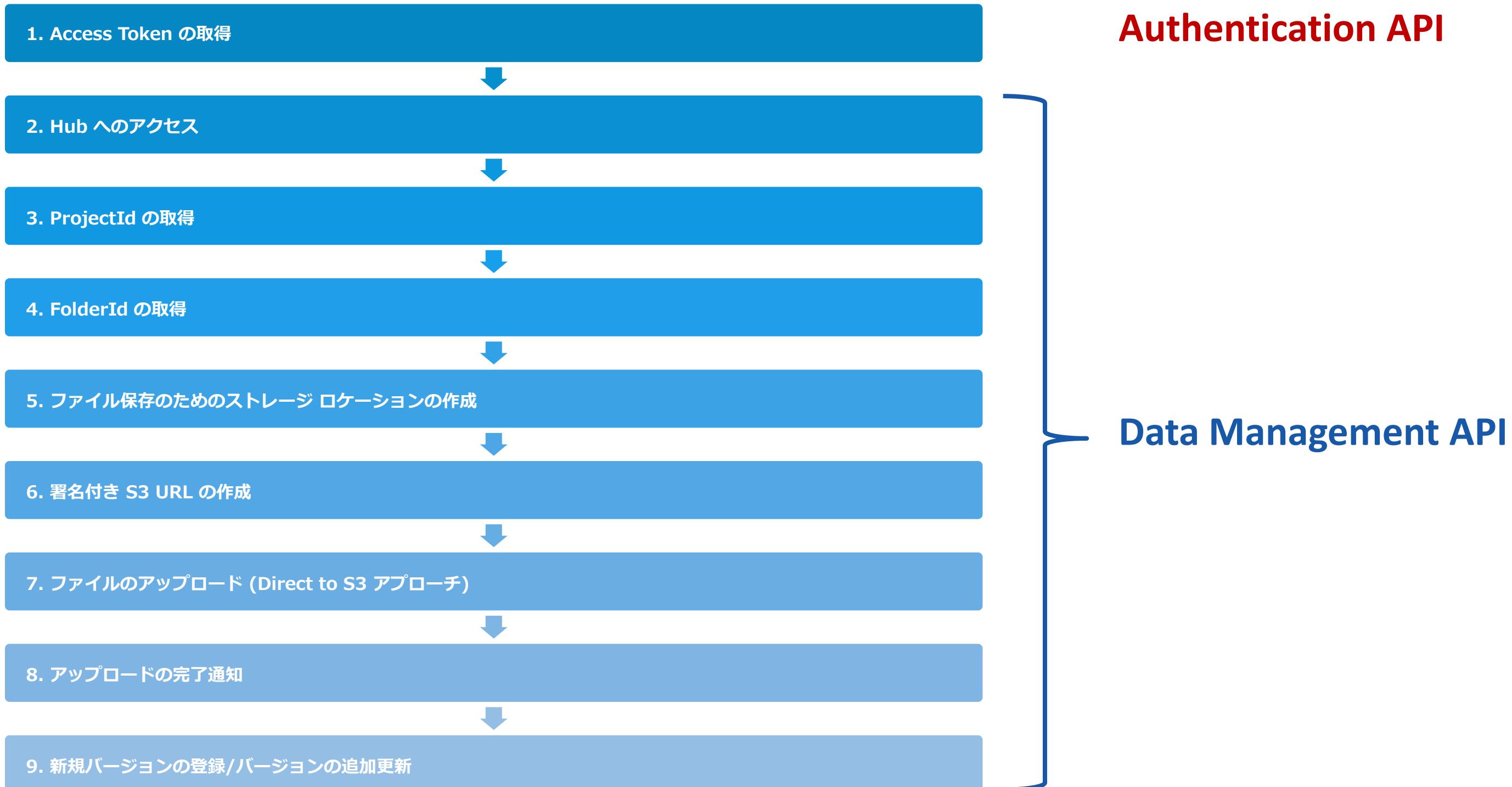
ここまでプロジェクト構成



クライアントが呼ぶ出す
エンドポイントを実装

Autodesk Docs の情報を取
得するロジックを定義

ファイルアップロードのフロー



チュートリアル: <https://aps.autodesk.com/en/docs/data/v2/tutorials/upload-file/>

SDK 利用方法: https://adndevblog.typepad.com/technology_perspective/2024/10/uploading-a-file-to-acc-using-the-net-sdk.html

ストレージ ロケーションの作成

- プロジェクトにアップロードするためのストレージロケーションを作成。

- REST API

POST projects/:project_id/storage

- SDK

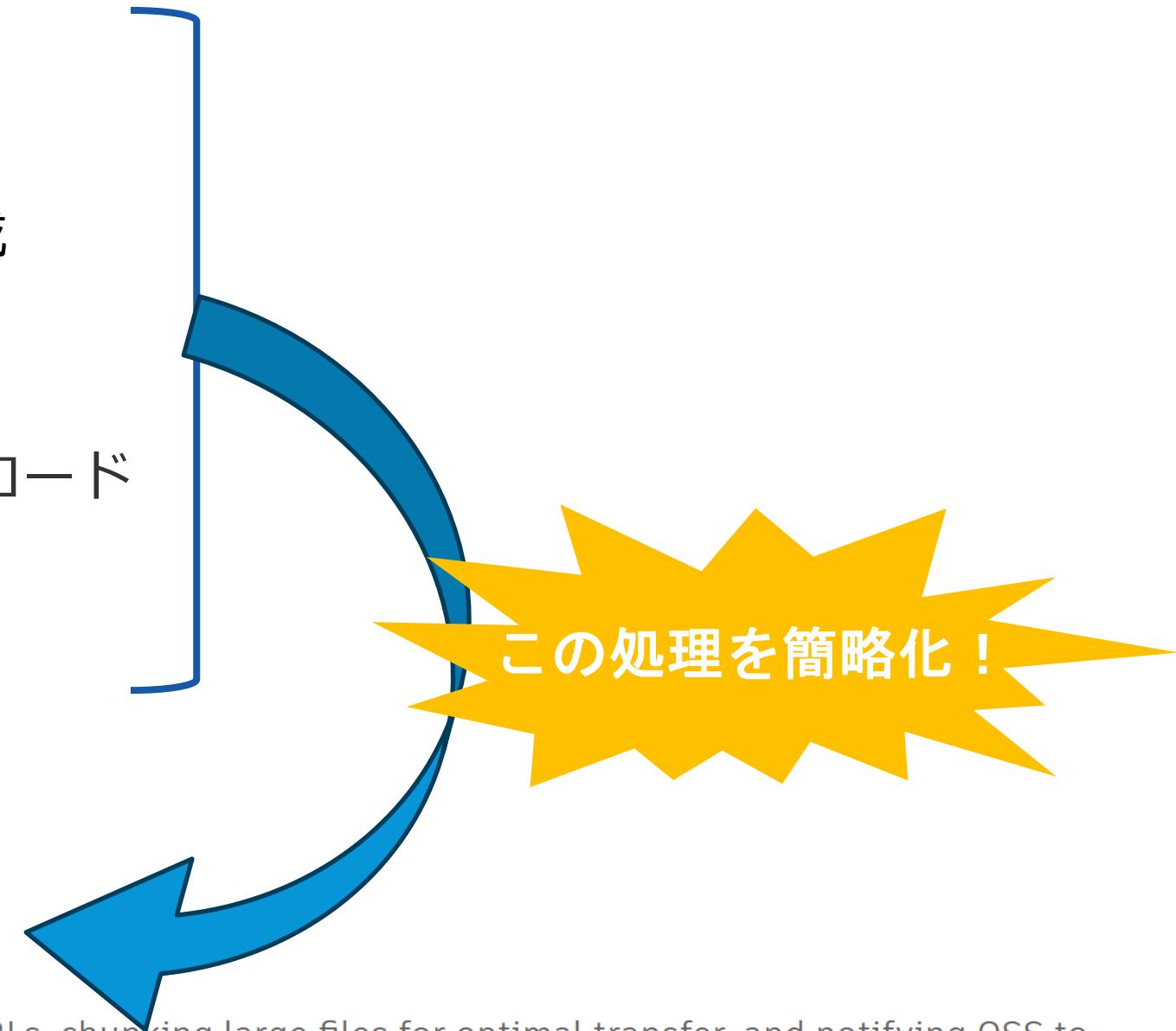
createStorage()

Operation: Create a Storage Location in OSS

```
curl -v 'https://developer.api.autodesk.com/data/v1/projects/:project_id/storage' \
-X 'POST' \
-H 'Authorization: Bearer AuIPTf4KYLTYGVnOHQ0cuolwCW2a' \
-H 'Content-Type: application/vnd.api+json' \
-d '{
    "jsonapi": {
        "version": "1.0"
    },
    "data": {
        "type": "objects",
        "attributes": {
            "name": "drawing.dwg"
        },
        "relationships": {
            "target": {
                "data": {
                    "type": "folders",
                    "id": "urn:adsk.wipprod:fs.folder:co.mgS-lb-BThaTdHnhIN_mbA"
                }
            }
        }
}
```

Direct-S3 アプローチによるアップロード

1. 署名付き S3 URL の作成
 - アップロードするファイルのパート数を算出
 - チャンクアップロードのための URL (最大25) を生成
 - bucketKey と objectKey、uploadKey
2. ファイルのアップロード
 - パートファイルを、対応する署名付き URL にアップロード
3. アップロードの完了
 - uploadKey を使用してアップロードを確定させる



SDK

`uploadObject()`

Operation: Uploads a file by transparently handling operations like obtaining signed upload URLs, chunking large files for optimal transfer, and notifying OSS to assemble the uploaded parts.

```
uploadObject( bucketKey , objectKey , sourceToUpload , optionalArgs ?): Promise <ObjectDetails>
```

新規バージョンの登録/バージョンの追加更新

- アップロード完了しただけでは、Autodesk Docs 上に表示されません。
- 初回アップロード時は、Item の新規登録が必要です。
- 2回目以降は、バージョンを更新します。

`createItem()`

Operation: Create an Item

`createItem(projectId , itemPayload , optionalArgs): Promise <CreatedItem>`

`createVersion()`

Operation: Create a Version

`createVersion(projectId , versionPayload , optionalArgs): Promise <CreatedVersion>`



<input type="checkbox"/> 名前	説明	バージ…	マーク…	サイズ
<input type="checkbox"/> ファミリタイプ選択用	--	--	--	--
<input type="checkbox"/> サンプル設備.rvt	v2	👑	82.7 MB	
<input type="checkbox"/> サンプル意匠.rvt	v1	👑	62.3 MB	
<input type="checkbox"/> サンプル構造.rvt	v1	👑	48 MB	

アップロード後の SVF/SVF2 変換と完了通知

	WebHooks API	Model Derivative API
1. Autodesk Docs にアップロード完了	<code>dm.version.added</code> <code>dm.version.modified</code>	
2. 自動的に SVF に変換	<code>extraction.updated</code> <code>extraction.finished</code>	<p>変換の完了</p> <p>GET:Manifest Polling</p>
3. SVF 変換後、自動的に SVF 2 に変換	<code>extraction.updated</code> <code>extraction.finished</code>	<ul style="list-style-type: none"> • <code>pending</code> • <code>inprogress</code> • <code>success</code> • <code>failed</code> • <code>timeout</code>



Viewer と UI

Viewer & UI

The screenshot shows a dark-themed web page for Autodesk Platform Services. At the top, there's a navigation bar with the Autodesk logo, a search bar, and a 'Sign in' button. Below the navigation is a secondary navigation bar with links for 'Solutions', 'Getting Started', 'Documentation', 'Success Stories', 'Community', 'Support', 'Pricing', and 'App Store'. On the left side, there's a sidebar with a tree-like menu. The 'Tutorials' section is expanded, showing 'Simple Viewer', 'Hubs Browser', and 'Viewer & UI'. Under 'Viewer & UI', there are links for 'Application Setup', 'Authentication', 'Data Browsing', 'Dashboard', 'Design Automation', 'ACC Administrator', 'ACC Issues (beta)', and 'Learn More'. The main content area has a breadcrumb navigation ('Home > Tutorials > Hubs Browser > Viewer & UI') and a title 'Viewer & UI'. A text block says, 'Finally, we're ready to build the client-side piece of our application.' Below it is a 'TIP' section with the text 'If you're developing with Node.js, you can use TypeScript definitions for the Viewer. Run' followed by a code block: 'npm install --save-dev @types/forge-viewer'. Another text block below says 'in your terminal to add the TypeScript definition file to your project.' At the bottom, there's a section titled 'Viewer logic' with the text 'Let's start by implementing the Viewer functionality for our application. Create a `viewer.js` file under the `wwwroot` subfolder with the following code:'.

A vertical sidebar on the right lists five categories: 'Viewer logic', 'Sidebar logic', 'Application logic', 'User interface', and 'Try it out'.

- Viewer logic
- Sidebar logic
- Application logic
- User interface
- Try it out

Viewer: 初期化

```
export function initViewer(container) {
  return new Promise(function (resolve, reject) {
    Autodesk.Viewing.Initializer({ env: 'AutodeskProduction', getAccessToken }, function () {
      const config = {
        extensions: ['Autodesk.DocumentBrowser']
      };
      const viewer = new Autodesk.Viewing.GuiViewer3D(container, config);
      viewer.start();
      viewer.setTheme('light-theme');
      resolve(viewer);
    });
  });
}
```

```
async function getAccessToken(callback) {
  try {
    const resp = await fetch('/api/auth/token');
    if (!resp.ok)
      throw new Error(await resp.text());
    const { access_token, expires_in } = await resp.json();
    callback(access_token, expires_in);
  } catch (err) {
    alert('Could not obtain access token. See the console for more details.');
    console.error(err);
  }
}
```

<div id="preview"></div>

document.getElementById('preview')

Scope
viewables:read

フォーマット: SVF vs SVF2



https://adndevblog.typepad.com/technology_perspective/2020/11/svf-and-svf2.html

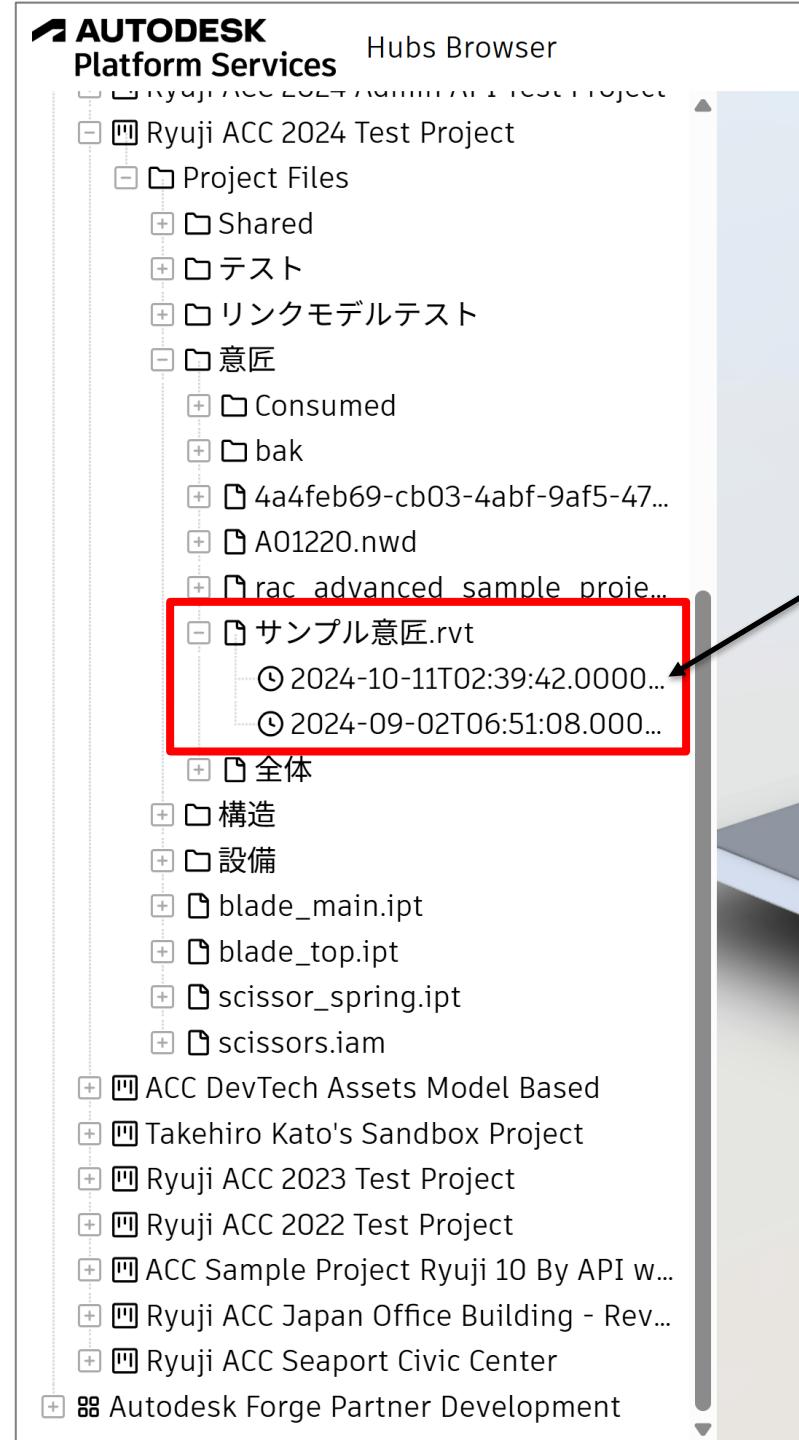
https://adndevblog.typepad.com/technology_perspective/2021/12/svf2-and-metadata-api-endpoints.html

Viewer: 初期化時の設定

```
var options = {
  env: 'AutodeskProduction2',
  api: 'streamingV2', // for models uploaded to EMEA change this option to 'streamingV2_EU'
  getAccessToken: function(onTokenReady) {
    var token = 'YOUR_ACCESS_TOKEN';
    var timeInSeconds = 3600; // Use value provided by APS Authentication (OAuth) API
    onTokenReady(token, timeInSeconds);
  }
};
```

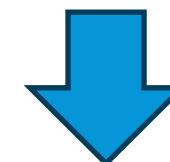
Parameter	SVF	SVF2
env	AutodeskProduction	AutodeskProduction2
api	derivativeV2 (for US)	streamingV2 (for US)
	derivativeV2_EU (for EU)	streamingV2_EU (for EMEA)
	derivativeV2_AUS (for AUS)	streamingV2_AUS (for AUS)

Viewer: URN の取得と Base64 エンコード



- チュートリアルでは、ツリーコンポーネントにて `versionId` が保持されています。
- `main.js` 内で `window.btoa()` メソッドを使用。

```
<li class="collapsed selectable draggable drop-target leaf" data-uid="version|urn:adsk.wipprod:fs.file:vf.省略?version=2">
  <div class="title-wrap">
    <a class="title icon icon-version" tabindex="1" unselectable="on">
      2024-10-11T02:39:42.0000000Z
    </a>
  </div>
<div class="wholerow"></div></li>
```



dXJuOmFkc2sud2lwcHJvZDpmcy5maWxlOnZmLIZHZUpyN
HFwUWJTekZPX0dRLWh1M0E/dmVyc2lvbj0y

Viewer: モデルのロードと AECModelData

```

export function loadModel(viewer, urn) {
  function onDocumentLoadSuccess(doc) {
    doc.downloadAecModelData( () => viewer.loadDocumentNode(doc, doc.getRoot().getDefaultGeometry()));
  }
  function onDocumentLoadFailure(code, message) {
    alert('Could not load model. See console for more details.');
    console.error(message);
  }
  Autodesk.Viewing.Document.load('urn:' + urn, onDocumentLoadSuccess, onDocumentLoadFailure);
}

```

- Doc.downloadAecModelData()
 - IFC ファイルの場合は、ifcLevelsEnabled: true を設定すると、レベルの情報が抽出されます。
 - AECModelData は、レベル、通芯、フェーズ、ビューポートなどの情報を保持しています。



Viewer: AECModelData の取得方法

```
// From document  
await doc.downloadAecModelData()  
  
// From bubble  
await Autodesk.Viewing.Document.getAecModelData(viewer.model.getDocumentNode())
```

```
NOP_VIEWER.model.getDocumentNode().getAecModelData()  
▼ {version: "1.0.0", documentId: "e5c3c912-80ce-4e47-8a5f-ec197cf063b0",  
copeBoxes: Array(0), ...} ⓘ  
  documentId: "e5c3c912-80ce-4e47-8a5f-ec197cf063b0"  
► grids: (27) [...], [...], [...], [...], [...], [...], [...], [...], [...], [...], [...],  
► level0CcluderIds: (5) [2432, 2462, 2538, 3091, 3330]  
► levels: (5) [...], [...], [...], [...], [...]  
► linkedDocuments: []  
► locationParameters: {placeName: "Boston, MA"}  
► phases: (2) [...], [...]  
► refPointTransformation: (12) [1, 0, 0, 0, 1, 0, 0, 0, 1, 30.19028688  
► scopeBoxes: []  
  version: "1.0.0"  
► viewports: (10) [...], [...], [...], [...], [...], [...], [...], [...], [...]  
► __proto__: Object
```

チュートリアル：VS Code 上の操作手順

1. **wwwroot** フォルダ下に **viewer.js** ファイルを作成して実装
2. **wwwroot** フォルダ下に **sidebar.js** ファイルを作成して実装
3. **wwwroot** フォルダ下に **main.js** ファイルを作成して実装
4. **wwwroot** フォルダ下に **index.html** ファイルを作成して実装
5. **wwwroot** フォルダ下に **main.css** ファイルを作成して実装
6. デバッグ実行 >> **localhost:8080**

The screenshot shows a dark-themed interface of the Visual Studio Code (VS Code) code editor. At the top, the menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a set of standard window controls. The title bar displays "Hub Browser".

The left sidebar contains the Explorer view, which lists the project structure:

- EXPLORER
- ... (ellipsis)
- JS aps.js
- JS hubs.js (highlighted)
- X JS auth.js
- JS server.js
- JS config.js

The main editor area shows the content of the `hubs.js` file, which defines several router.get() callbacks for API endpoints related to hubs, projects, and contents. The code uses async/await syntax and imports from `getProjects`, `getProjectContents`, and `getItemVersions` functions.

```
routes > JS hubs.js > router.get('/api/hubs/:hub_id/projects/:project_id/contents') callback
17  router.get('/api/hubs/:hub_id/projects', async function (req, res, next) {
18    try {
19      const projects = await getProjects(req.params.hub_id, req.internalOAuthToken.access_token);
20      res.json(projects.map(project => ({ id: project.id, name: project.attributes.name })));
21    } catch (err) {
22      next(err);
23    }
24  );
25
26  router.get('/api/hubs/:hub_id/projects/:project_id/contents', async function (req, res, next) {
27    try {
28      const entries = await getProjectContents(req.params.hub_id, req.params.project_id, req.query.folder_id, req.internalOAuthToken.access_token);
29      res.json(entries.map(entry => ({ id: entry.id, name: entry.attributes.displayName, folder: entry.type === 'folders' })));
30    } catch (err) {
31      next(err);
32    }
33  );
34
35  router.get('/api/hubs/:hub_id/projects/:project_id/contents/:item_id/versions', async function (req, res, next) {
36    try {
37      const versions = await getItemVersions(req.params.project_id, req.params.item_id, req.internalOAuthToken.access_token);
38      res.json(versions.map(version => ({ id: version.id, name: version.attributes.createTime })));
39    } catch (err) {
40      next(err);
41    }
42  );
43
44  module.exports = router;
```

Below the editor, the bottom navigation bar includes links for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (underlined), and PORTS. The TERMINAL tab is active, showing the output of an npm command:

```
15 packages are looking for funding
  run `npm fund` for details

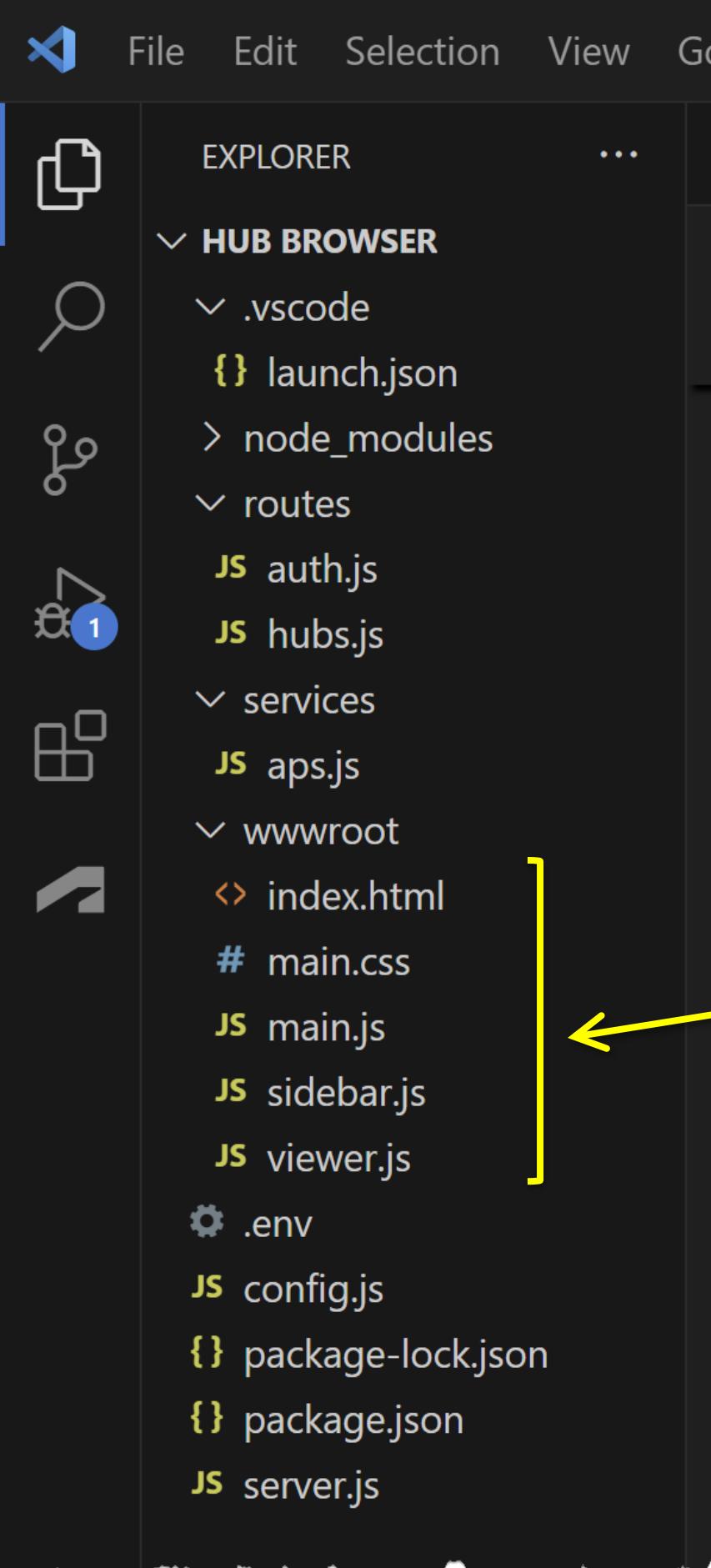
  found 0 vulnerabilities
● PS C:\Users\ogasawr\Hub Browser> npm install --save @aps_sdk/authentication @aps_sdk/data-management

  added 44 packages, and audited 122 packages in 10s

  18 packages are looking for funding
  run `npm fund` for details
```

The bottom status bar indicates the current file is "Launch Server (Hub Browser)" and provides status information: Line 27, Column 10, Spaces: 4, UTF-8, CRLF, and a JavaScript icon.

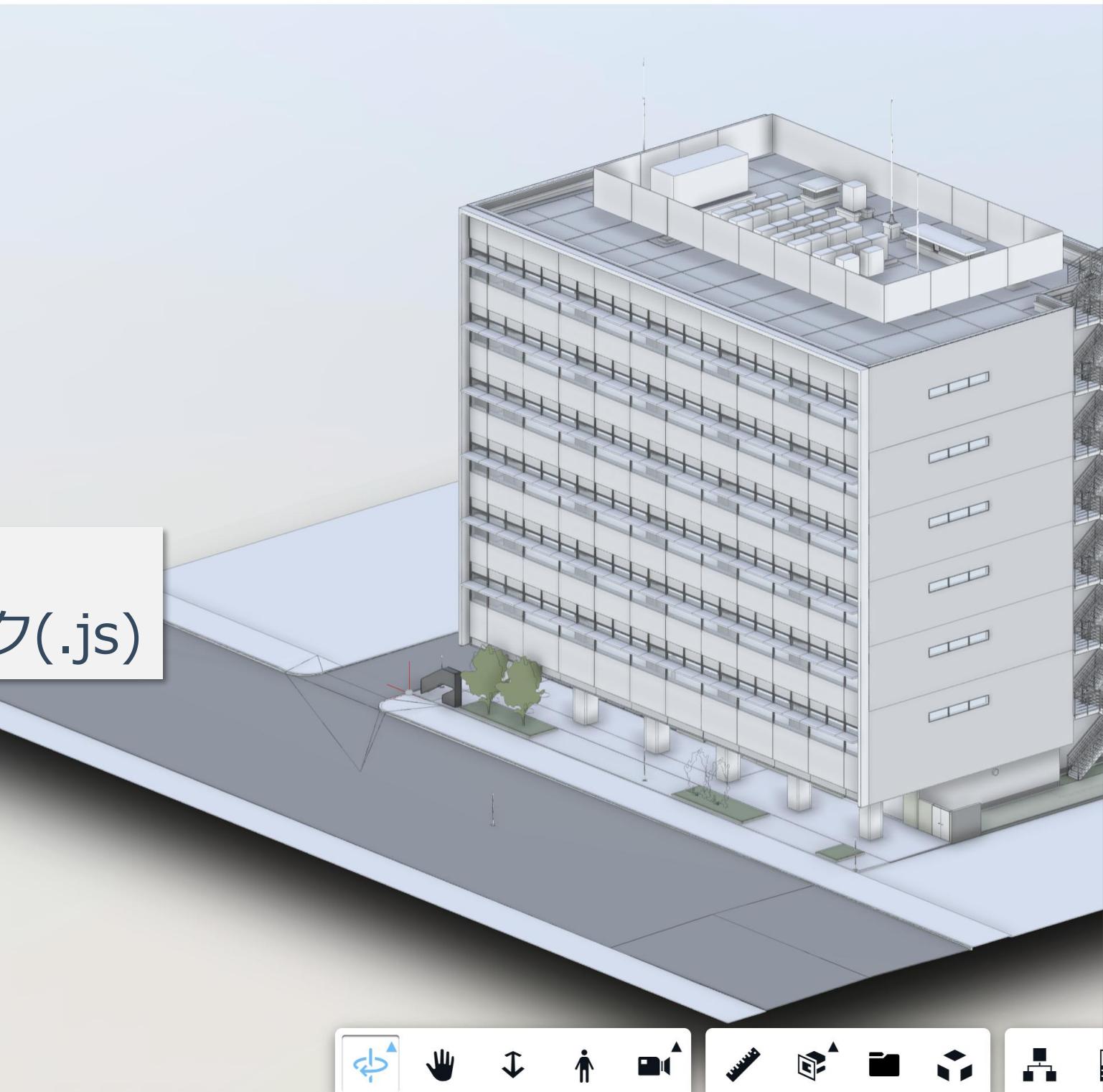
ここまでプロジェクト構成



AUTODESK Platform Services Hubs Browser

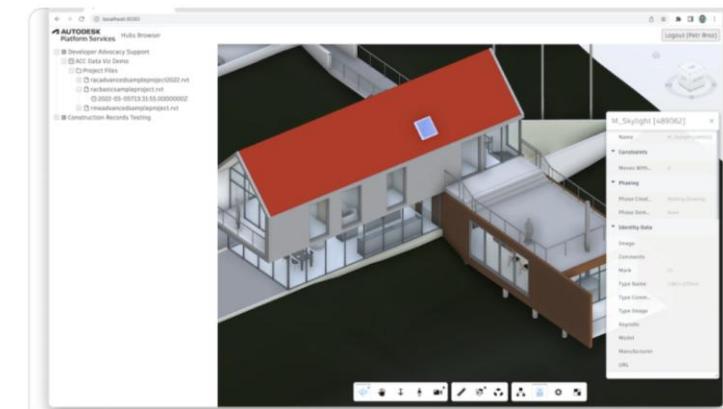
- Ryuji Ogasawara
- Developer Advocacy Support
 - Ryuji ACC Worksharing 2024
 - Ryuji ACC Japan Office Building
 - ACC Sample Project Ryuji 8 By API wi...
 - Component Library Project 024d6467...
 - Sample Project
 - ACC Sample Project Ryuji 11 By API w...
 - Ryuji ACC 2025 Test Project
 - Project Files
 - 意匠
 - サンプル意匠.rvt
 - 2025-04-16T01:30:14.0000...
 - Ryuji ACC Test Project
- ACC Sample Project Ryuji 9 By API wi...
- ACC Sample Project Ryuji 12 By Manu...
- Ryuji ACC 2024 Admin API Test Project
- Ryuji ACC 2024 Test Project
- ACC DevTech Assets Model Based
- Takehiro Kato's Sandbox Project
- Ryuji ACC 2023 Test Project
- Ryuji ACC 2022 Test Project
- ACC Sample Project Ryuji 10 By API w...
- Ryuji ACC Japan Office Building - Rev...
- Ryuji ACC Seaport Civic Center
- Autodesk Forge Partner Development

Web ページ定義一式
UI(.html、.css)とロジック(.js)



Hub Browser GitHub リポジトリ

- Code Sample : Hub Browser
 - Node.js ベース
 - APS SDK を使用
 - 3-legged 認証
 - Autodesk クラウドストレージにアクセス
 - Viewer 表示



Hubs Browser (.NET)

Autodesk Platform Services application following the Hubs Browser tutorial from https://github.com/Autodesk-Platform/hubs-browser-dotnet

Data Management, Viewer, C#

[View in GitHub](#) [Demo](#)



Hubs Browser (Node.js)

Autodesk Platform Services application built by following the Hubs Browser tutorial from https://github.com/Autodesk-Platform/hubs-browser-nodejs

Data Management, Viewer, JavaScript

[View in GitHub](#) [Demo](#)

Tips: Viewer Extension (機能拡張)

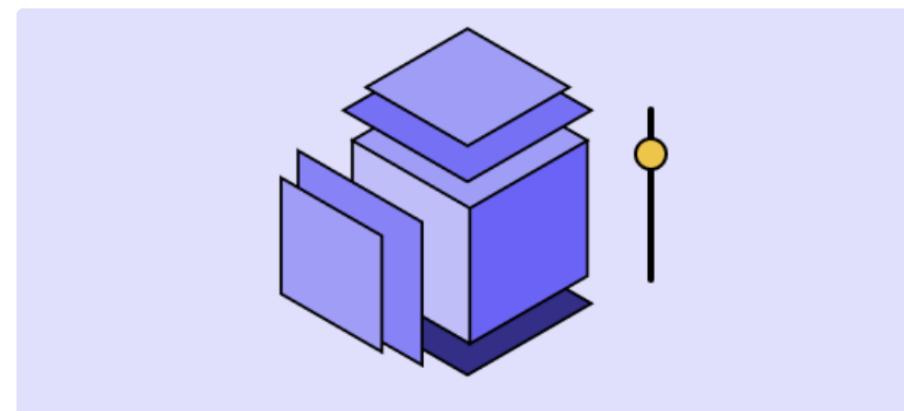
```
14  export function initViewer(container) {
15      return new Promise(function (resolve, reject) {
16          Autodesk.Viewing.Initializer({ env: 'AutodeskProduction', getAccessToken }, function () {
17              const config = {
18                  extensions: ['Autodesk.DocumentBrowser', 'Autodesk.AEC.Minimap3DExtension', 'Autodesk.AEC.LevelsExtension']
19              };
20              const viewer = new Autodesk.Viewing.GuiViewer3D(container, config);
21              viewer.start();
22              viewer.setTheme('light-theme');
23              resolve(viewer);
24          });
25      });
26  }
27 }
```

- Autodesk 公開の Extension をロード
 - <https://aps.autodesk.com/en/docs/viewer/v7/reference/Extensions/>
- 独自に Extension を作成してロード
 - <https://get-started.aps.autodesk.com/tutorials/dashboard/>
 - [Writing an Extension](#)

- + Ryuji Ogasawara
- Ryuji ACC Worksharing 2024
- + Ryuji ACC Japan Office Building
- + ACC Sample Project Ryuji 8 By API wi...
- + Component Library Project 024d6467...
- + Sample Project
- + ACC Sample Project Ryuji 11 By API w...
- Ryuji ACC 2025 Test Project
 - Project Files
 - 意匠
 - サンプル意匠.rvt
- ⌚ 2025-04-16T01:30:14.0000...
- + Ryuji ACC Test Project
- + ACC-point-clouds
- + Ryuji ACC Design Collaboration Revit ...
- + Ryuji ACC 2024 Template Project
- + Ryuji ACC 2024 Template
- + ACC Sample Project Ryuji 9 By API wi...
- + ACC Sample Project Ryuji 12 By Manu...
- + Ryuji ACC 2024 Admin API Test Project
- + Ryuji ACC 2024 Test Project
- + ACC DevTech Assets Model Based
- + Takehiro Kato's Sandbox Project
- + Ryuji ACC 2023 Test Project
- + Ryuji ACC 2022 Test Project
- + ACC Sample Project Ryuji 10 By API w...
- + Ryuji ACC Japan Office Building - Rev...
- + Ryuji ACC Seaport Civic Center
- + Autodesk Forge Partner Development

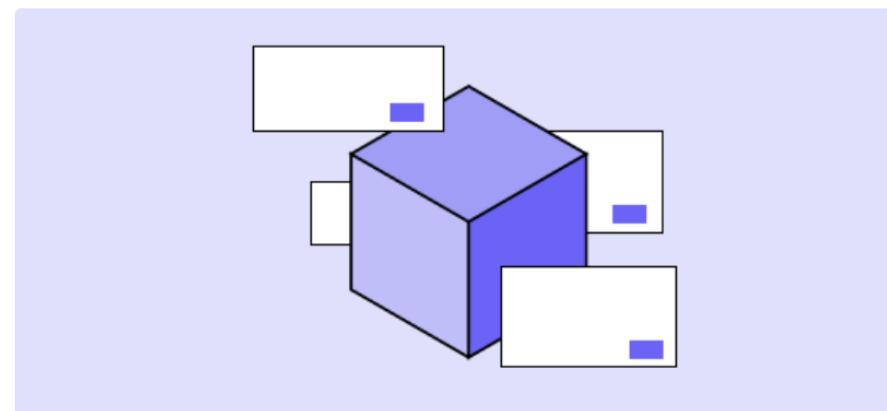


Interactive Examples



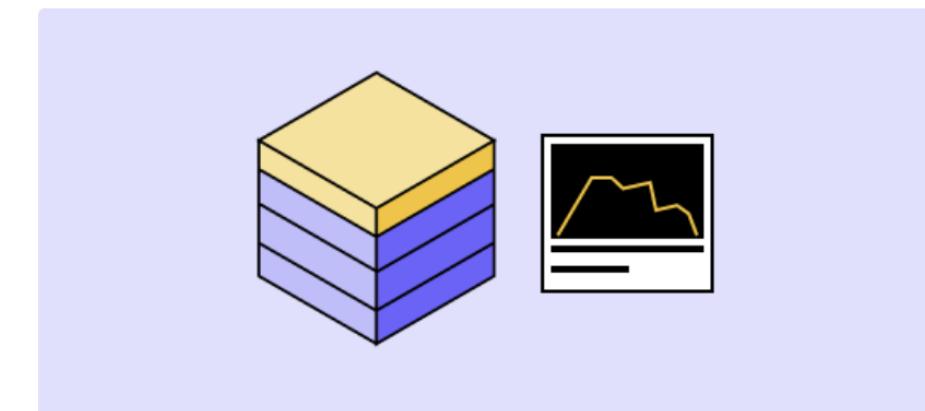
Controlling Viewer states

Learn to programmatically control Viewer states, like exploding a model.



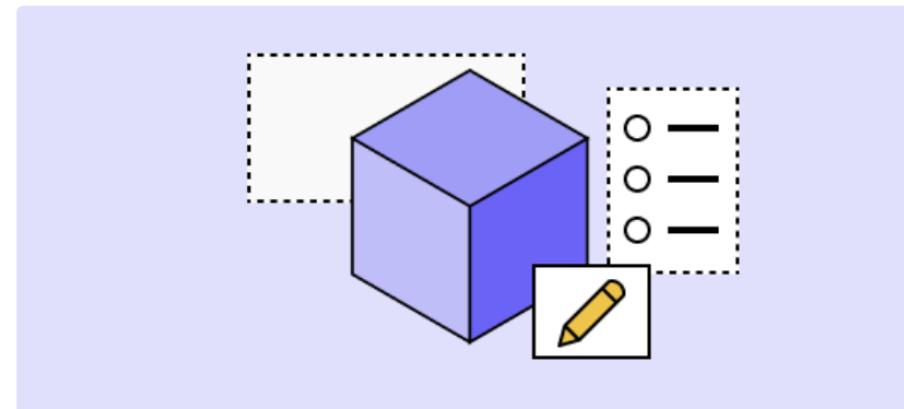
Handling Viewer events

Learn to respond to users' actions and perform tasks based on specific triggers.



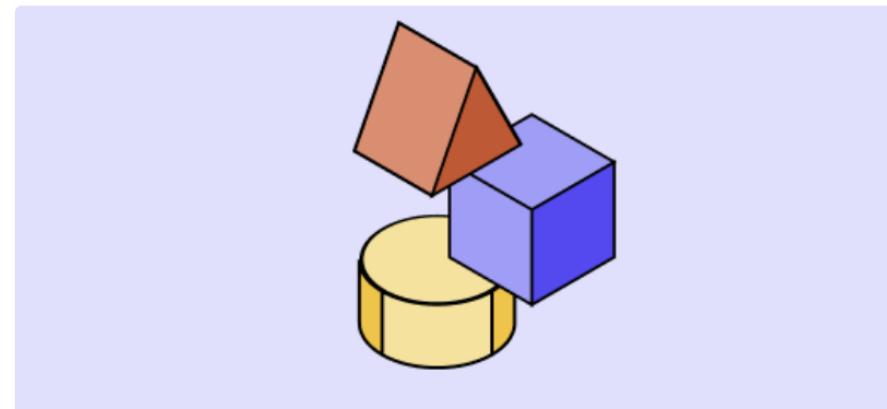
Querying model properties

Add functionality to search for specific model property values and display results in a dialog window.



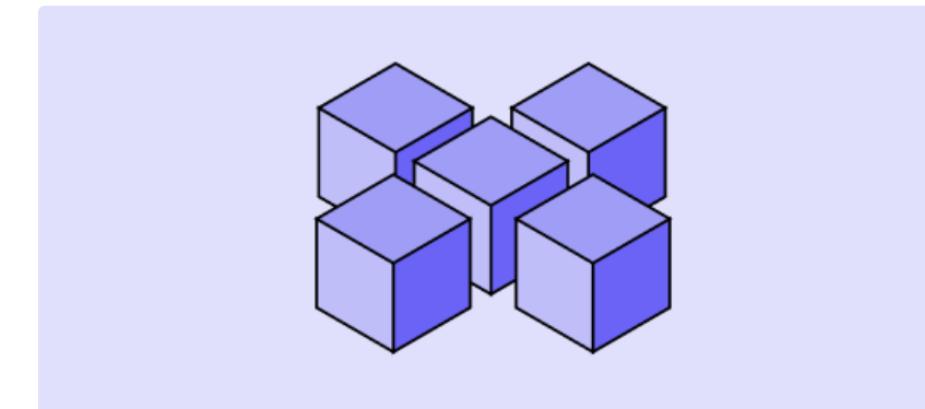
Customizing Viewer UI

Create custom UI elements for unique interactions, like a button that displays a dialog window when clicked.



Customizing UI scene

Create and interact with custom three.js geometry that integrates like standard Viewer models.

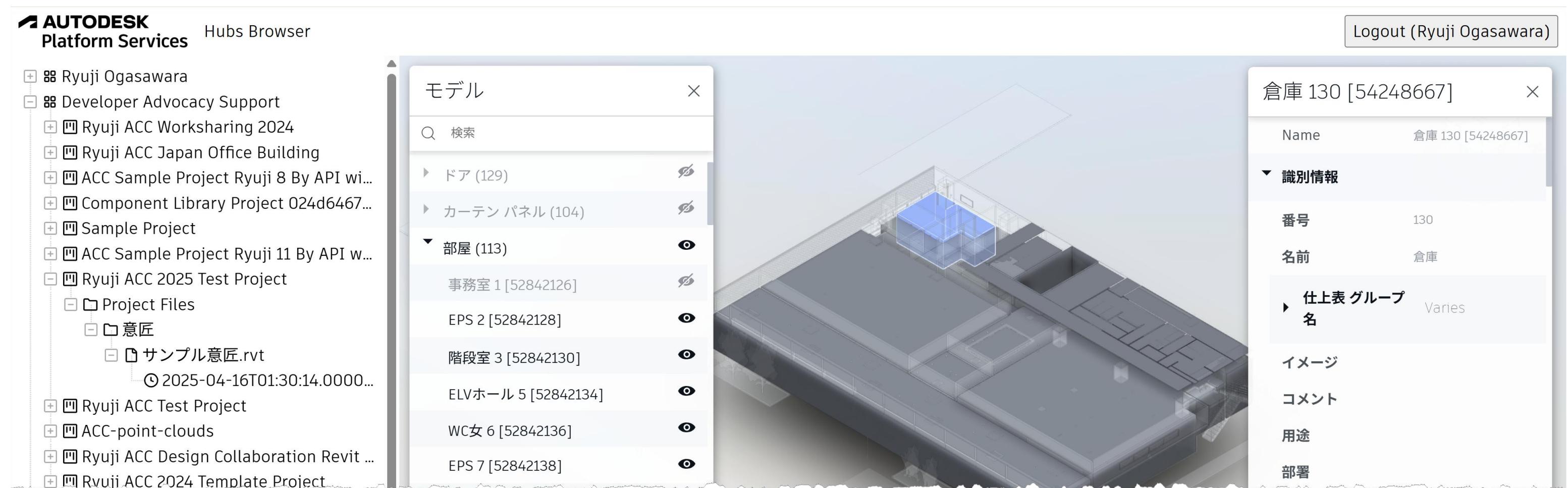


Model aggregation

Combine a Revit model and a Fusion 360 model in the same view.

Tips: Revit モデル固有の要素の表示

- 「部屋」「スペース」「ゾーン」を 3D ビュー上で表示できます。ただし、デフォルトでは非表示。
- ACC / BIM 360 では、**generateMasterViews** を true に設定して変換されます。
 - Revit モデルで作成したフェーズ毎に、「部屋」などの要素を含むマスタービュー（3D ビュー）が生成されるようになります。
- マスタービューの表示名は、既定値として Revit モデルのフェーズ名になります。



Tips: Revit モデル固有の要素の表示

- マスター ビューの検索に Special Id を使用できます。
 - '08f99ae5-b8be-4f8d-881b-128675723c10'

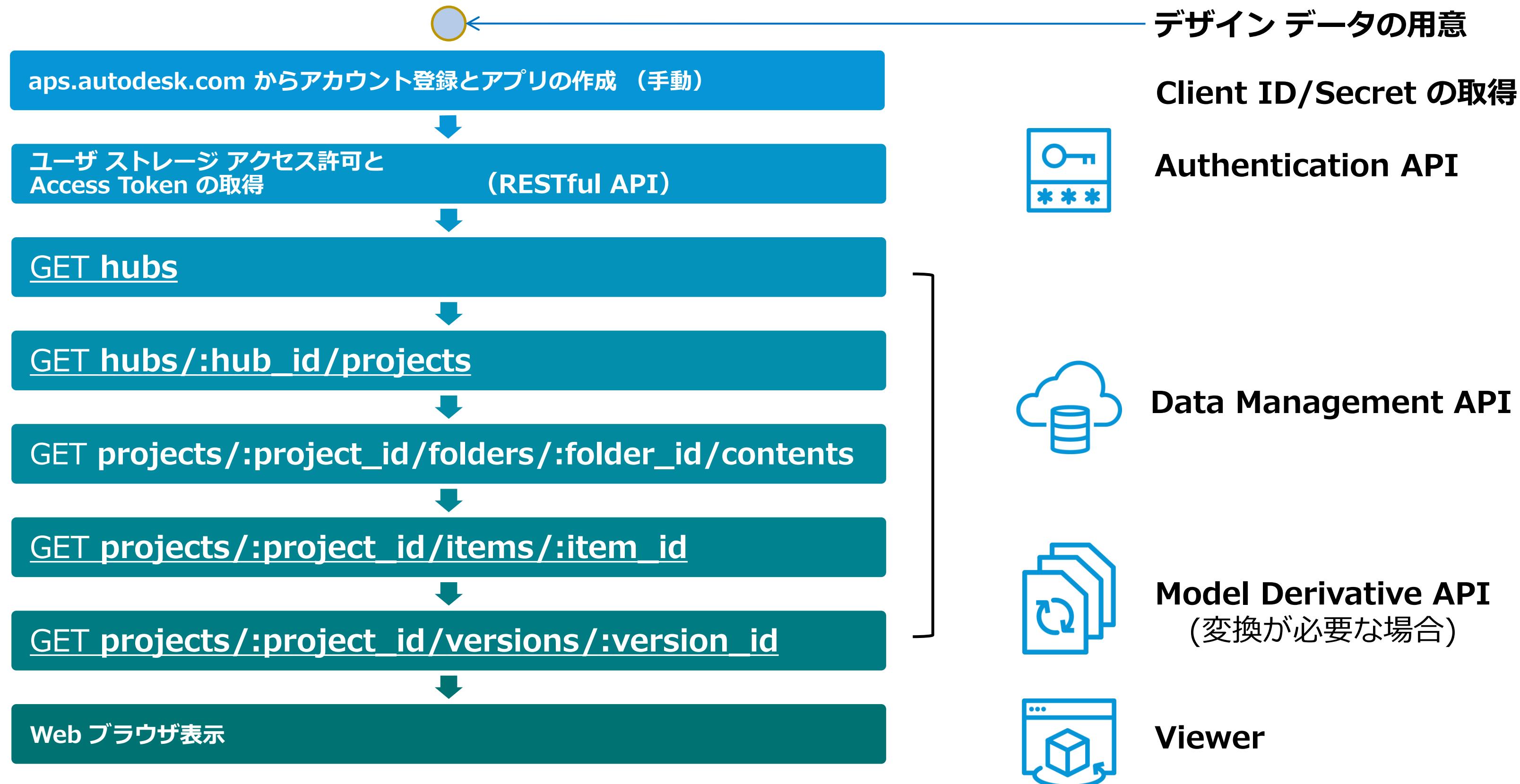
```
28  export function loadModel(viewer, urn) {
29    function onDocumentLoadSuccess(doc) {
30      const root = doc.getRoot();
31      const viewables = root.search({ 'type': 'geometry', 'role': '3d' });
32      console.log('Viewables:', viewables);
33      const phaseViews = viewables.filter(v => v.data.name === v.data.phaseNames &&
34        v.getViewableRootPath().includes('08f99ae5-b8be-4f8d-881b-128675723c10'));
35      console.log('Master Views:', phaseViews);
36      viewer.loadDocumentNode(doc, phaseViews[0], { skipHiddenFragments: false });
37    }
38    function onDocumentLoadFailure(code, message) {
39      alert('Could not load model. See console for more details.');
40      console.error(message);
41    }
42    Autodesk.Viewing.Document.load('urn:' + urn, onDocumentLoadSuccess, onDocumentLoadFailure);
43  }
```

Tips: 要素の識別

```
{
  "objectid": 1706, = dbId
  "name": "11030_FAN_消音ボックス付送風機 [17299672]", 
  "externalId": "706c1fe4-e098-4258-8c15-25674aab130c-0107f8d8", = Revit の場合は Element.UniqueId
  "properties": {
    "拘束": {
      "風量": "400.000 CMH",
      "機外静圧": "118.000 Pa",
      "基準レベル": "1F",
      "基準レベルからの高さ": "3327.500 mm",
      "ホスト": "レベル : 1F",
      "ホストからのオフセット": "3327.500 mm",
      "既定の高さ": "0.000 mm",
      "番手": "#1 1/4"
    },
    "文字": {
      "■給気／□排気": "Yes",
      "特殊仕様": ""
    }
  }
}
```

- **objectid / dbId**
APS Viewer が要素を識別するための ID。
特定の要素のプロパティにアクセスする際などにも利用。
- **externalId**
Revit の UniqueId に保持されている GUID と同じ値。

3-legged OAuthでのViewer利用手順の理解



ご注意：進化し続ける APS・開発作業に終わりはなし クラウドサービス

- APS の機能向上によって動作/仕様が変わる可能性
 - 例) Design Automation API への OAuth スコープ code:all 適用
 - APS Viewer は特定バージョン指定も可能
- クラウド開発にはメインテナンスが必須
- デスクトップ製品のアドイン開発とは異なります！
- 開発ベンダーとのサブスクリプション？契約が必要！！



付録： コストとサポート

APS 利用に対する‘課金’とは？

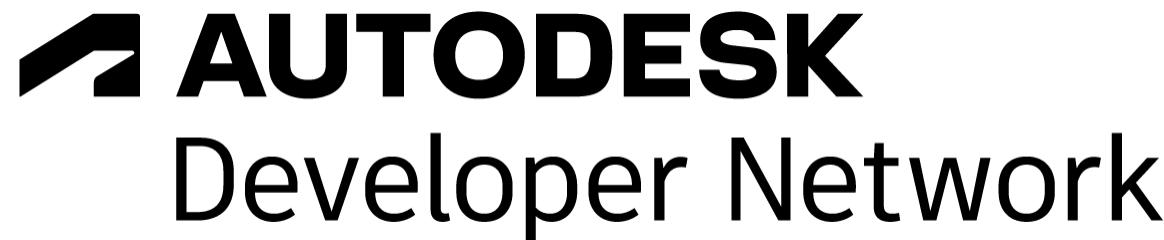
<https://aps.autodesk.com/pricing>

- 特定の APS API 利用に Autodesk Flex による従量課金

プレミアムAPIとサービス	コスト	情報
Design Automation API	2.0	Flex トークン / <u>処理時間</u>
Flow Graph Engine API	1.0	Flex トークン / <u>処理時間</u>
Model Derivative API	0.5	Flex トークン / <u>コンプレックスジョブ</u>
	0.1	Flex トークン / <u>シンプルジョブ</u>
Reality Capture API	1.0	50枚の写真毎のFlexトークン



Autodesk Developer Network (ADN) とは



- 年間契約の開発サポートプログラム（有償）です
 - 新製品/バージョン、API、テクノロジの啓蒙と流布
 - 無償ブログ公開、ウェビナー等イベント開催、その他
 - ADN 加入メンバーへの API 開発サポートの提供
 - API の問題点、不具合、要望の米国本社へのフィードバック
 - Autodesk App Store アプリの公開審査とサポート

ADN 特典

1. ADN Extranet (会員専用サイト) アクセスを提供
2. オートデスク API についてのお問合せ窓口提供
3. 開発用途の製品ライセンスを提供
4. マーケティング用ロゴの提供
5. ユーザー数に応じた Autodesk Flex の提供
6. 開発者向け有償イベントへの優待価格提供

今後のリソース

- サンプル コード : GitHub
 - <https://github.com/Developer-Autodesk>
 - <https://github.com/Autodesk-APS>
- サポート : StackOverflow
 - <https://APS.autodesk.com/en/support/get-help>
- ブログ : APS Community Blog
 - <https://APS.autodesk.com/blog>



Autodesk and the Autodesk logo are registered trademarks or trademarks of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and/or other countries. All other brand names, product names, or trademarks belong to their respective holders. Autodesk reserves the right to alter product offerings and specifications at any time without notice, and is not responsible for typographical or graphical errors that may appear in this document.

© 2025 Autodesk. All rights reserved.