



Revit モデルをクラウドに接続する

小笠原 龍司
オートデスク株式会社

アジェンダ

- 1 Revit モデルをクラウドに移行する
- 2 APS アプリからのアプローチ
- 3 Revit API からのアプローチ

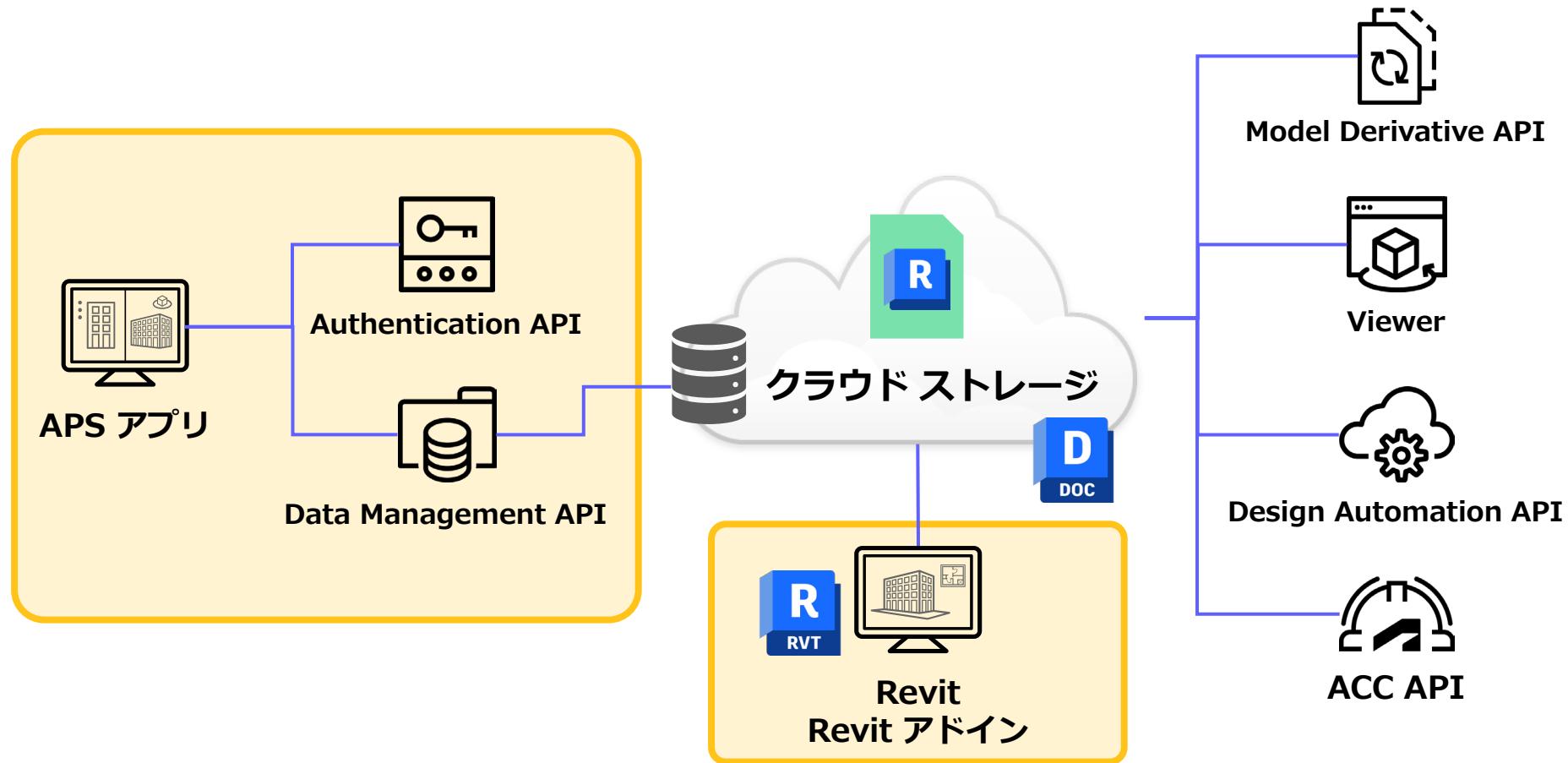




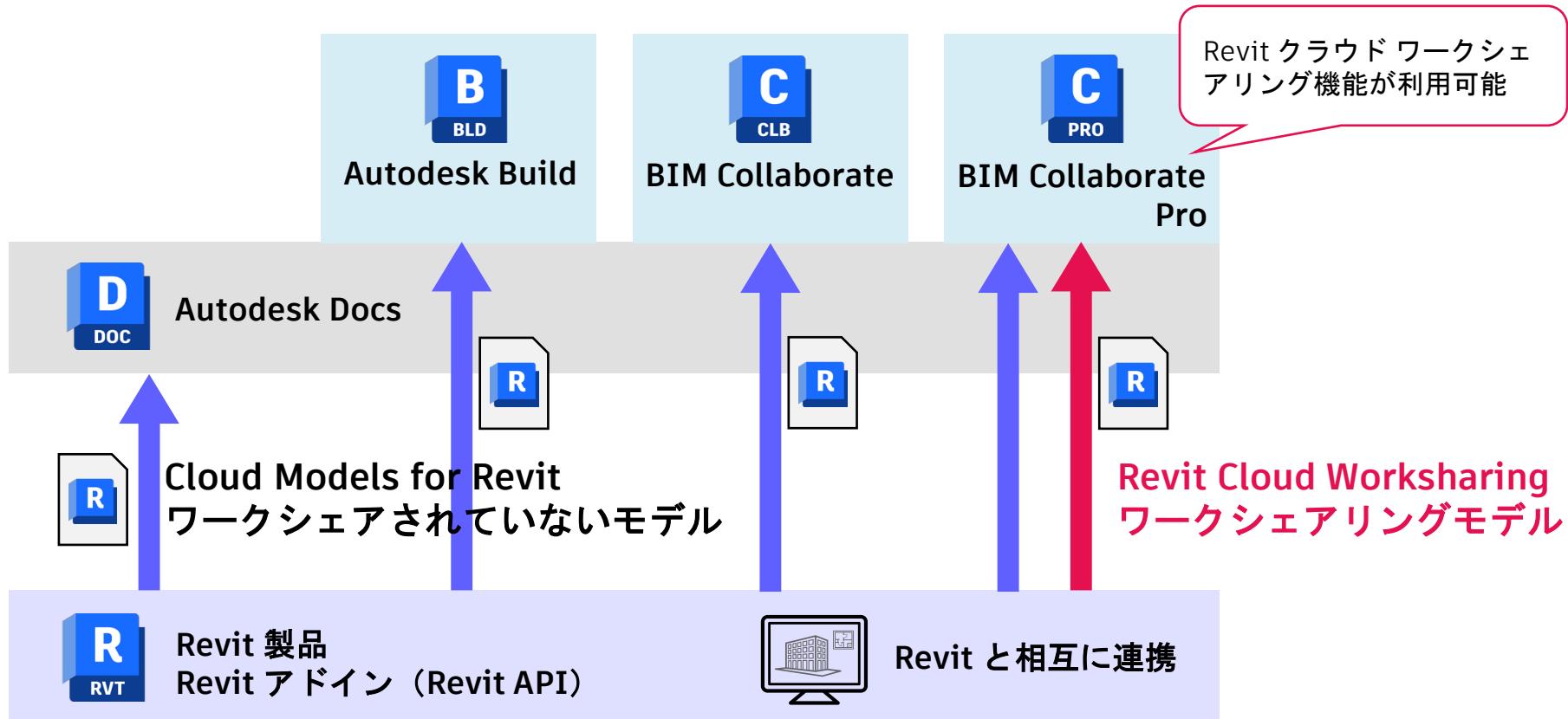
Revit モデルをクラウドに移行する



Revit モデルをクラウドに移行する方法



Revit/Revit アドインからクラウドモデルを保存



Revit/Revit アドインからクラウドモデルを保存

- ワークシェアされていない Revit モデル

- Cloud Models for Revit

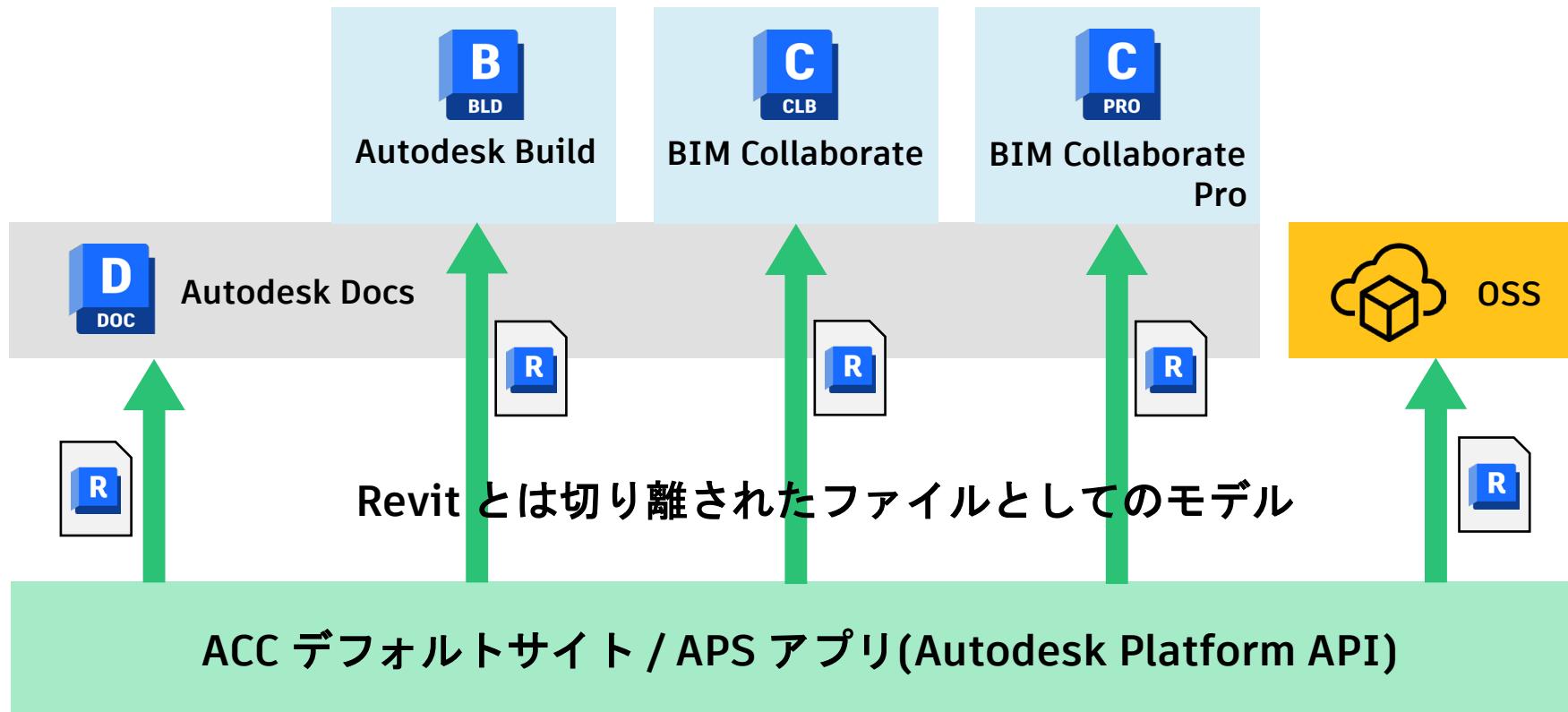
- Autodesk Docs/BIM 360 Docs アカウント、Revit のサブスクリプションユーザー
 - ローカル モデルを ACC/BIM 360 の選択したプロジェクトのフォルダに保存する。
 - Revit からモデルを開いて編集可。ただし同時に編集は不可。
 - 後から Revit でワークシェアリングを有効化することができる。

- ワークシェアされている Revit モデル

- Revit Cloud Worksharing

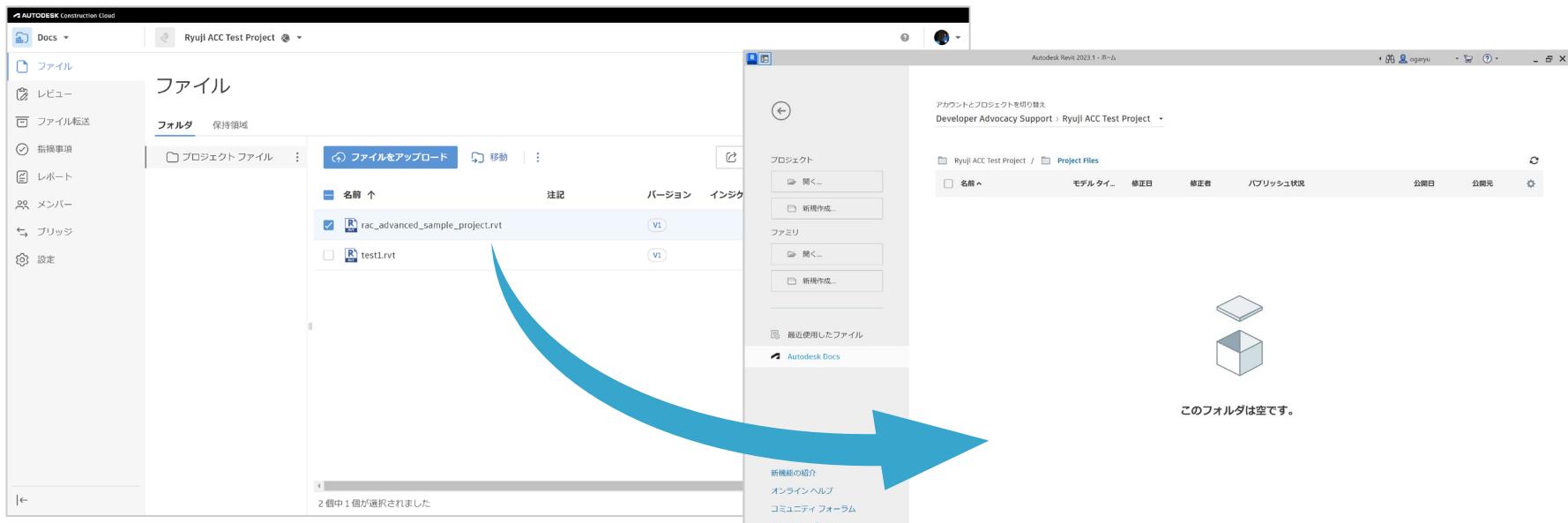
- ACC Docs/BIM 360 Docs アカウント、Revit のサブスクリプションユーザー
 - ACC (BIM Collaborate または Collaborate Pro) /BIM 360(BIM 360 Design) のサブスクリプションユーザー
 - Revit からコラボレート機能を通じて モデルをアップロードする。

APS アプリから Autodesk Docs への保存



APS アプリから Autodesk Docs への保存

- Revit を介さずに Web ブラウザ・ Data Management API でアップロード
 - Cloud Models for Revit や Revit Cloud Worksharing とは切り離された単純なファイル管理
 - Revit からクラウドモデルとしてアクセスすることはできません。 (要ダウンロード)



回避策: APS - Design Automation API

1. Design Automation API の AppBundle からクラウド上のモデルを開く
2. Revit API でワークシェアリングを有効化
3. 別名でクラウドモデルとして保存
 - 上書き保存は不可。新規作成または別名保存のみサポート。 (Revit 2022+)

```
Document doc = data.RevitDoc;
Application app = data.RevitApp;

CloudModelLocation cloudModelLocation = JsonConvert.DeserializeObject<CloudModelLocation>(File.ReadAllText("params.json"));

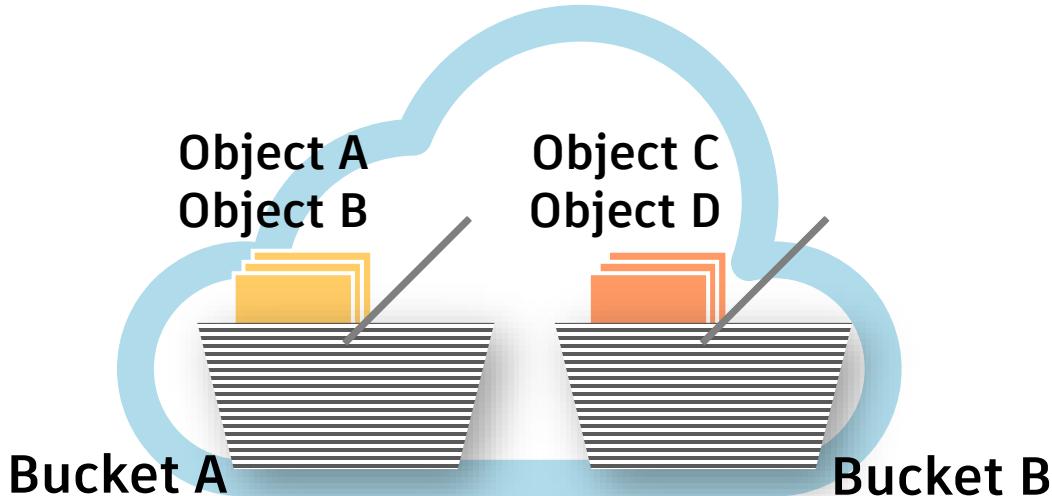
bool checkResult = doc.CanEnableWorksharing();

if (checkResult)
{
    doc.EnableWorksharing("Shared Levels and Grids", "Workset1");

    doc.SaveAsCloudModel(
        new Guid(cloudModelLocation.AccountId),
        new Guid(cloudModelLocation.ProjectId),
        cloudModelLocation.FolderId,
        "newRCWModel.rvt");
}
```

OSS のクラウドストレージに保存

- Object Storage Service (OSS)
- APS アプリ専用の共有ストレージ領域 (S3)
- Autodesk Docs に搭載のバージョン管理、フォルダ管理機能はありません。
- Bucket と Object をアプリ開発者が管理

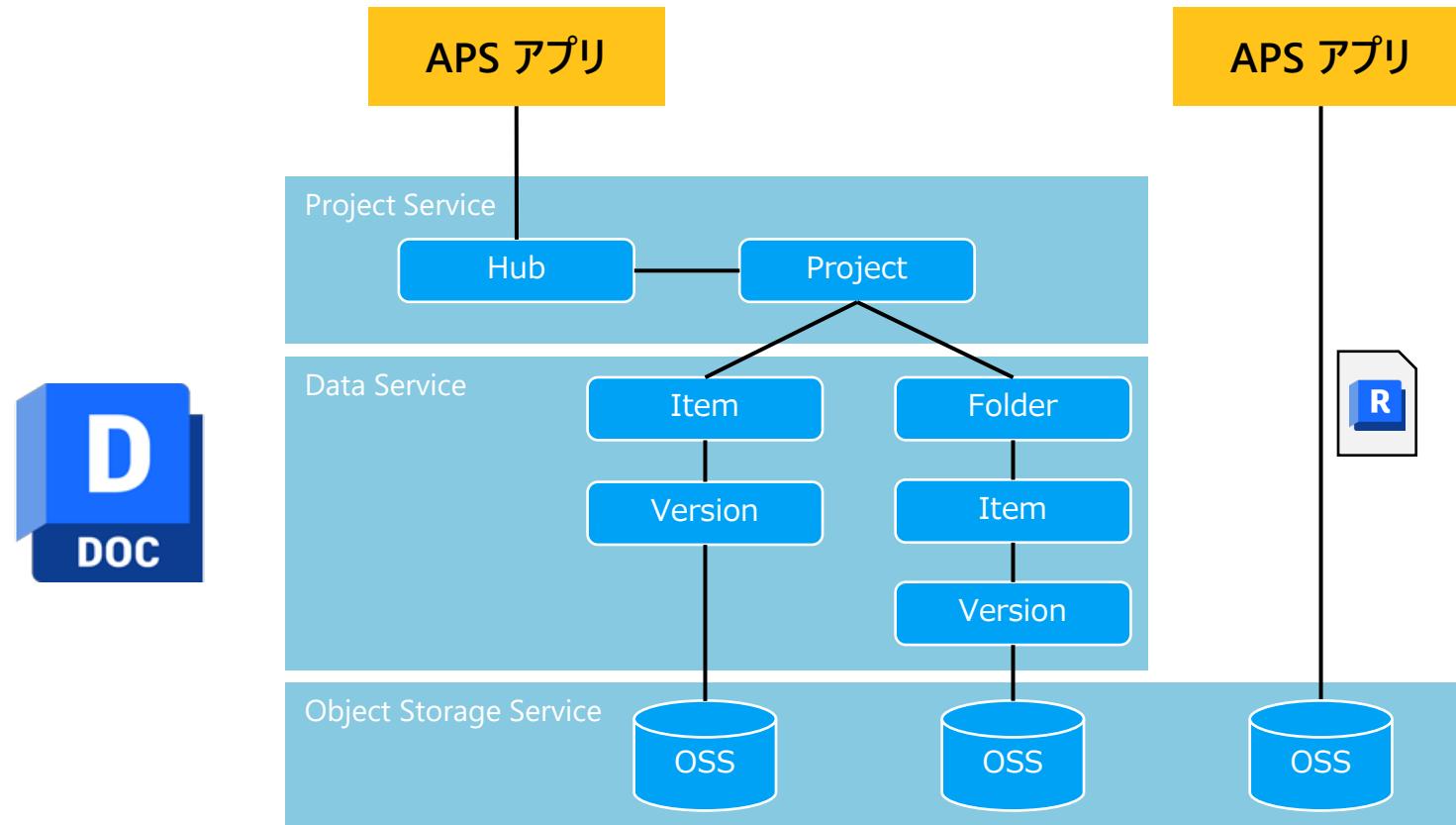




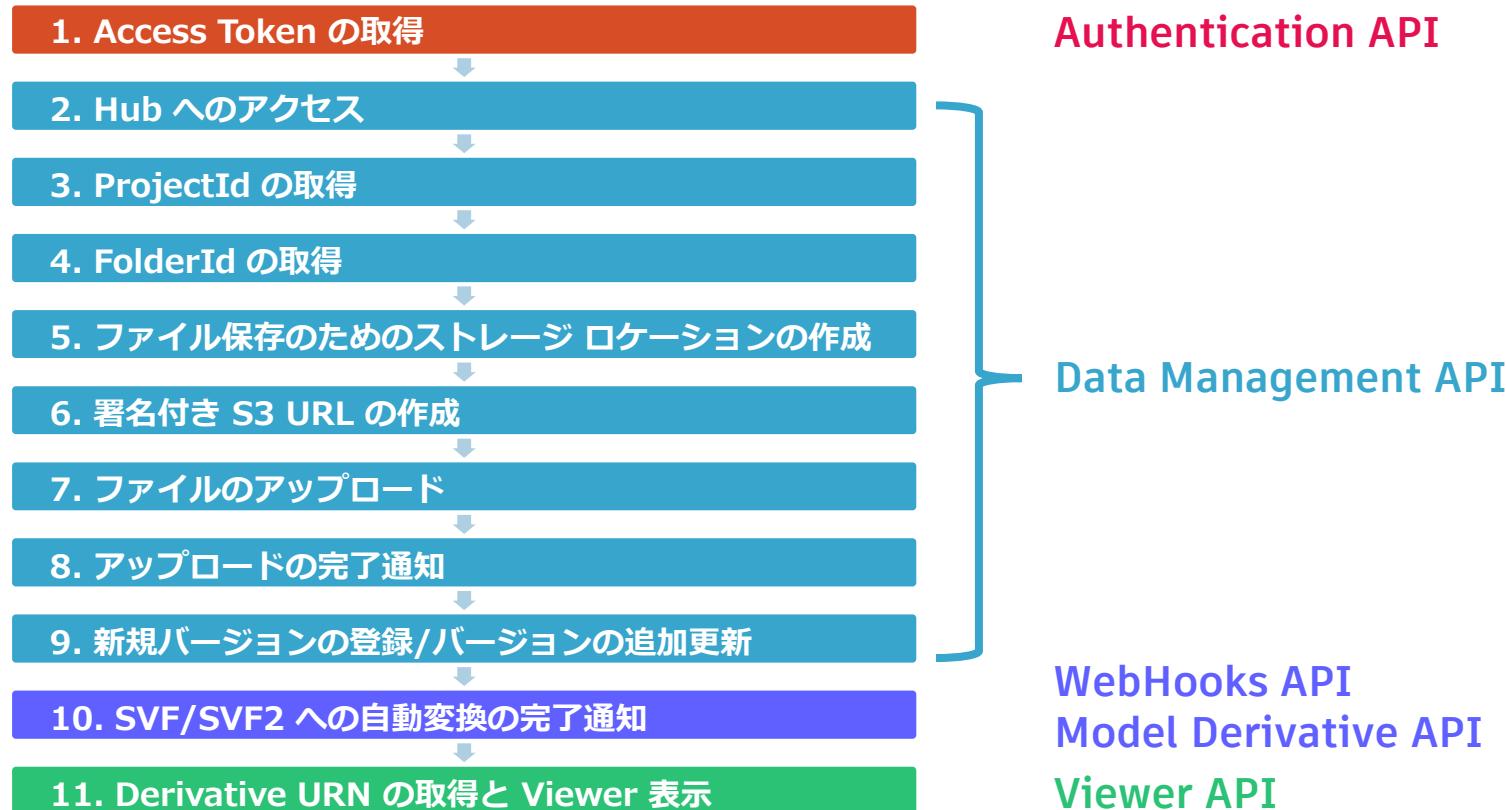
APS アプリからのアプローチ



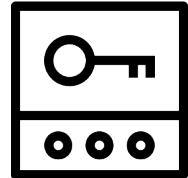
シナリオ/目的に応じた使い分けが必要



アップロードと表示のフロー (Autodesk Docs)



Access Token (アクセス トークン)



- 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
 - Authentication API – OAuth2, V2 リリース済み。
 - APS の各 API リクエスト時に使用します。
 - フォーマットは JSON ウェブ トークン (JWT)。
 - 有効期限が設定されます。
- 取得の際に必要なパラメータ
 - デベロッパキー
 - Client ID と Client Secret のペア
 - APS ポータルで App (アプリ) 作成時に取得
 - Scope
 - リソースへのアクセス権限範囲を指定
 - 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

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
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)
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

Access Token の種類

- **2-legged Authentication**
 - サーバー間通信で利用することを想定したアクセストークン。
 - アプリは APS サーバーからの認証を得ればアクセス可能（2者間通信）
 - Autodesk Docs のユーザストレージ領域・アプリ専用のストレージ（OSS）にアクセス
- **3-legged Authentication**
 - アプリがユーザーからの認可を必要とするリソースを利用するためのアクセストークン。
 - Autodesk ログインページに一旦リダイレクトして、ユーザーが Autodesk アカウントでログインし、ユーザーからそのリソースへのアクセスの同意が得られた場合に取得できます。
 - Autodesk Docs のユーザストレージ領域にアクセス
 - V2 で PKCE ワークフロー（デスクトップおよび Single-Page Web アプリ用）をサポート
- 認証・認可後に Access Token（アクセストークン）を発行

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。

The screenshot displays three API endpoint pages from the Autodesk developer documentation:

- projects/:project_id/items/:item_id**:
 - Method: GET
 - Description: Retrieves metadata for a specified item. Items represent word documents.
 - Notes:
 - The tip version for the item is included in the `included` array.
 - To retrieve metadata for multiple items, see the [ListItems](#) command.
 - New! Autodesk Construction Cloud platform (ACC). Note that this endpoint is part of the Autodesk Construction Cloud APIs, see the [Autodesk Construction Cloud documentation](#).
 - Resource Information
 - Method and URI: GET https://developer.api.autodesk.com/projects/:project_id/items/:item_id
 - Authentication Context: user context optional
 - Required OAuth Scopes: data:read
- users/me**:
 - Method: GET
 - Description: Retrieves the profile information about an end user, including the user's name, email address, and profile picture.
 - Notes:
 - This endpoint does not currently provide full information about a user, such as their roles or permissions.
 - It requires `create`, or `full control` permissions.
 - New! Autodesk Construction Cloud platform (ACC). Note that this endpoint is part of the Autodesk Construction Cloud APIs, see the [Autodesk Construction Cloud documentation](#).
 - Resource Information
 - Method and URI: GET <https://developer.api.autodesk.com/users/me>
 - Authentication Context: user context required
 - Required OAuth Scopes: account:read
- buckets/:bucketKey/objects**:
 - Method: GET
 - Description: List objects in a bucket. It is only available to the bucket creator.
 - New! Autodesk Construction Cloud platform (ACC). Note that this endpoint is part of the Autodesk Construction Cloud APIs, see the [Autodesk Construction Cloud documentation](#).
 - Resource Information
 - Method and URI: GET <https://developer.api.autodesk.com/oss/v2/buckets/:bucketKey/objects>
 - Authentication Context: app-only
 - Required OAuth Scopes: data:read

Hub と Project

- ACC のアカウント ≈ Hub (AccountId = 'b.' + HubId)
 - https://acc.autodesk.com/account-admin/projects/accounts/489*****210b/

The screenshot shows the Autodesk Construction Cloud interface. At the top, there's a navigation bar with the Autodesk logo, 'AUTODESK Construction Cloud', and dropdown menus for 'Account Admin' and 'Developer Advocacy Support'. The 'Developer Advocacy Support' menu is highlighted with a yellow box. Below the navigation is a sidebar with icons for 'プロジェクト' (Project), 'メンバー' (Members), '会社' (Company), '役割' (Roles), 'プロジェクトテンプレ...' (Project Templates), 'ライブラリ' (Library), 'アプリ' (Apps), and '設定' (Settings). The main content area has a title 'プロジェクト' (Project) and tabs for 'アクティブ' (Active) and 'アーカイブ済み' (Archived). A blue button '+プロジェクトを作成' (Create Project) is visible. Below this is a table with columns: '名前' (Name), '番号' (Number), 'タイプ' (Type), 'メンバー' (Members), and '会社' (Company). Two projects are listed: 'ACAD-PushToDocs' (highlighted with a red box) and 'ACC API Review EK'. The 'ACAD-PushToDocs' row also has a red box around it.

名前	番号	タイプ	メンバー	会社
ACAD-PushToDocs	-	⊕	1	1
ACC API Review EK	-	⊕	4	1

Folder と Item, Version

- プロジェクトのルートフォルダ配下に[プロジェクトファイル]フォルダがあります。

The screenshot shows the Autodesk Construction Cloud interface. At the top, there's a navigation bar with the 'AUTODESK Construction Cloud' logo, a 'Docs' dropdown, the project name 'Ryuji ACC 2023 Test Project' (which is highlighted with a red box), and user profile icons.

The main area is titled 'Files'. On the left, there's a sidebar with various options: 'File' (selected and highlighted with a blue box), 'Review', 'File Transfer', 'Task Item', 'Report', 'Member', 'Bridge', and 'Setting'. Below the sidebar, a green box highlights the 'Project Files' folder under the 'Folders' tab.

The 'Project Files' folder contains several sub-folders: 'Shared' (with 'Data Exchange' and 'In Progress'), 'Collaboration Test...', '意匠' (selected and highlighted with a blue box), 'Structures', and 'Equipment'.

In the center, there's a file upload section with a 'Upload file' button and a table for managing files. The table has columns: 'Name' (with a checkbox), 'Version' (with a dropdown), 'Indicator' (with a status icon), 'Size' (with a size indicator), 'Last Updated' (with a date), 'Updater' (with a user icon and name), 'Mark' (with a gear icon), and 'More' (with a three-dot menu). Two files are listed:

Name	Version	Indicator	Size	Last Updated	Updater	Mark	More
サンプル意匠.rvt	V8	Consumed	--	2023年10月15日 16:39	Ryuji Ogasawara	--	⋮
サンプル意匠.rvt	V8	Consumed	117.4 MB	2023年10月18日 23:21	Ryuji Ogasawara	SS	⋮

Data Management API のアクセストークン

- 3-legged トークン
- 2-legged トークン
 - APS アプリは、アカウントの管理者と同様に、すべてのユーザーにアクセスできてしまいます。
- **2-legged トークン + x-user-id ヘッダ**
 - リクエストヘッダに x-user-id を追加することで、API コールは指定されたユーザーの代理としてのみ動作するように制限されます。
 - X-user-id は、ACC Admin API – GET Project Users のレスポンス内 autodeskId プロパティ。

GET projects/{projectId}/users

ストレージロケーションの作成

- プロジェクト上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します。
 - ProjectId、FolderId、ファイル名

```
public async Task<dynamic> CreateStorage(string projectId, string folderId, string objectName)
{
    var projectsApi = new ProjectsApi();

    StorageRelationshipsTargetData storageRelData = new StorageRelationshipsTargetData(StorageRelationshipsTargetData.TypeEnum.Folders, folderId);
    CreateStorageDataRelationshipsTarget storageTarget = new CreateStorageDataRelationshipsTarget(storageRelData);
    CreateStorageDataRelationships storageRel = new CreateStorageDataRelationships(storageTarget);
    BaseAttributesExtensionObject attributes = new BaseAttributesExtensionObject(string.Empty, string.Empty, new JsonApiLink(string.Empty), null);
    CreateStorageDataAttributes storageAtt = new CreateStorageDataAttributes(objectName, attributes);
    CreateStorageData storageData = new CreateStorageData(CreateStorageData.TypeEnum.Objects, storageAtt, storageRel);
    CreateStorage storage = new CreateStorage(new JsonApiVersionJsonapi(JsonApiVersionJsonapi.VersionEnum._0), storageData);

    try
    {
        DynamicJsonResponse storageCreated = await projectsApi.PostStorageAsync(projectId, storage);
        return storageCreated;
    }
    catch (Exception e)
    {
        Debug.Print("Exception when calling ProjectsApi.PostStorage: " + e.Message);
        return null;
    }
}
```

object ID (bucket key + object key)
urn:adsk.objects:os.object:wip.dm.prod/2a6d61f2-49df-4d7b-9aed-439586d61df7.rvt

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

日本語解説: https://adndevblog.typepad.com/technology_perspective/2022/03/data-management-oss-object-storage-service-migrating-to-direct-to-s3-approach.html

https://adndevblog.typepad.com/technology_perspective/2022/11/considering-direct-s3-approach.html

SDK ではヘルパーソッドもサポート

- 旧 Forge SDK : `ObjectsApi.uploadResources()` メソッドが公開済み。
- 新 SDK でも対応予定。



```
public async Task<ObjectDetails> UploadModel(string objectKey, Stream content, Tokens tokens)
{
    var objApi = new ObjectsApi();
    objApi.Configuration.AccessToken = tokens.InternalToken;

    var results = await objApi.uploadResources("wip.dm.prod", new List<UploadItemDesc> {
        new UploadItemDesc(objectKey, content)
    });
    if (results[0].Error)
    {
        throw new Exception(results[0].completed.ToString());
    }
    else
    {
        var json = results[0].completed.ToJson();
        return json.ToObject<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。

<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

1. Autodesk Docs にアップロード完了

dm.version.added

dm.version.modified

2. 自動的に SVF に変換

extraction.updated

extraction.finished

3. SVF 変換後、自動的に SVF 2 に変換

extraction.updated

extraction.finished

Model Derivative API

変換の完了

GET:Manifest
Polling

- pending
- inprogress
- success
- failed
- timeout

Viewer 表示: URN の取得

- ソースファイルの URN を取得
 - Data management API で VersionId を取得して Base64 エンコード

Version Id : urn:adsk.wipprod:fs.file:vf.GT7D9cshRGS7GsP1YBQNVQ?version=2

Base64 エンコード（誤）：

dXJuOmFkc2sud2lwchHJvZDpmcy5maWxlOnZmLkdUN0Q5Y3NoUkdTN0dzUDFZQIFOVIE/dmVyc2lvbj0y

正しい URN は、「？」で Split して、それぞれを「_」で連結した結果の文字列になります。

Base64 エンコード（正）：

dXJuOmFkc2sud2lwchHJvZDpmcy5maWxlOnZmLkdUN0Q5Y3NoUkdTN0dzUDFZQIFOVIE_dmVyc2lvbj0y

- 下記のエンドポイントのレスポンスから data::relationships::derivatives::data::id の値を取得
 - GET: projects/:project_id/folders/:folder_id/contents
 - GET: projects/:project_id/items/:item_id
 - GET: projects/:project_id/items/:item_id/versions
 - GET: projects/:project_id/versions/:version_id

Viewer 表示: ドキュメントのロード

```
var documentId = 'urn:dXJu0mFkc2sub2JqZWN0czpvcy5vYmp1Y3Q6bXktYnVja2V0L215LWF3ZXNvbWUtZm9yZ2UtZmlsZS5ydnQ';
Autodesk.Viewing.Document.load(documentId, onDocumentLoadSuccess, onDocumentLoadFailure);

function onDocumentLoadSuccess(viewerDocument) {
    var defaultModel = viewerDocument.getRoot().getDefaultGeometry();
    viewer.loadDocumentNode(viewerDocument, defaultModel);
}

function onDocumentLoadFailure() {
    console.error('Failed fetching Forge manifest');
}
```



サンプルコード：チュートリアルサイト

- <https://tutorials.autodesk.io/>
- APS 開発者支援チームによる **旧 Forge SDK** を使用。

- Hubs Browser

- Sidebar コンポーネント
- Viewer 表示



Viewer 拡張: AggregatedView クラス

- 複数モデルの BubbleNode (2D または 3D の Viewable) を重ねてロード。
- それぞれのビューを切り替えて表示、またはモデルを動的にトグル表示 (ON/OFF) 可能。
- diffOptions の設定でビューの差分表示が可能。
- 次の Extension が自動的にロードされます。
 - Autodesk.CrossFadeEffects
 - Autodesk.AEC.LevelsExtension
 - Autodesk.ModelStructure
 - Autodesk.AEC.HyperlinkExtension
 - Autodesk.AEC.Minimap3DExtension
 - Autodesk.AEC.CanvasBookmarkExtension
 - Autodesk.AEC.DropMeExtension
 - 一人称視点モードも有効



Viewer 拡張: AggregatedView クラス

- 複数モデルを重ねて統合表示する例。



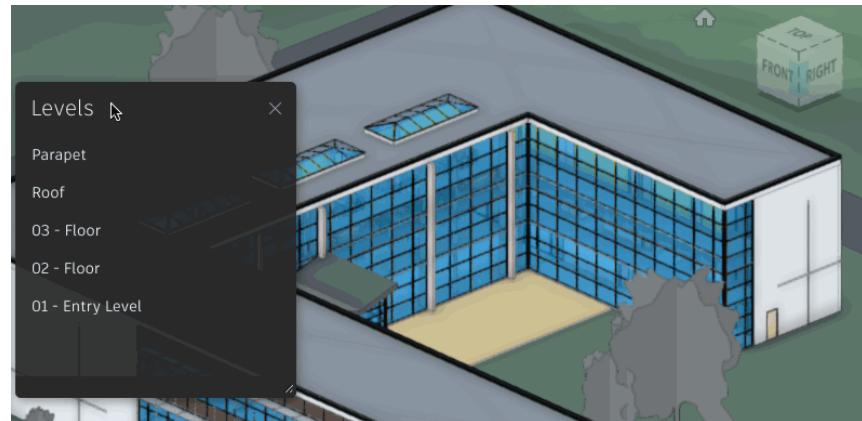
Viewer 拡張: AggregatedView クラス

Levels Extension で 3D モデルを指定のレベルのみ表示し、Minimap3D Extension で、2D 図面ビューと連動して視点の移動ができます。

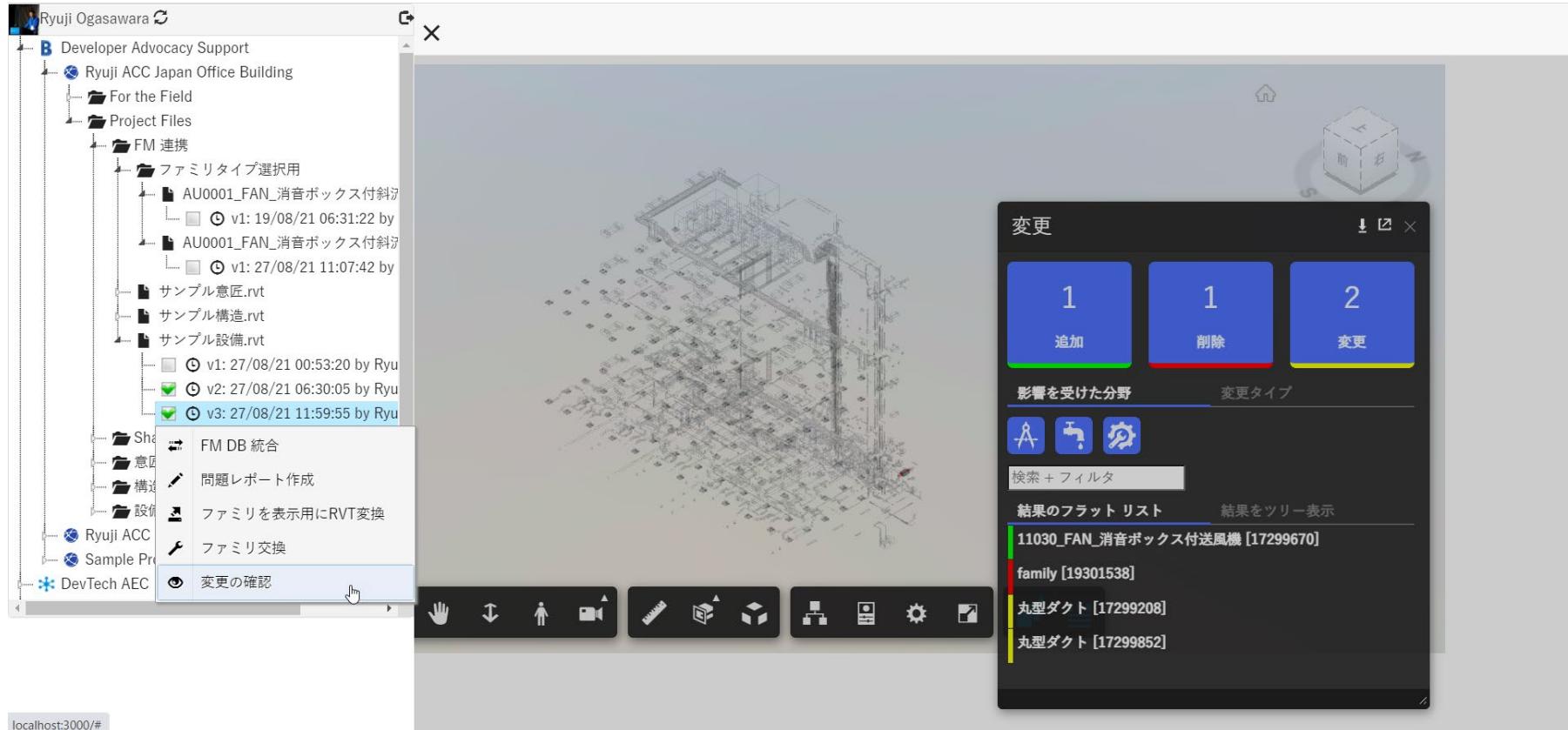


Viewer 拡張: AggregatedView クラス

1. ドキュメントを Viewer にロード。
2. onDocumentLoadSuccess(doc) 処理内で、doc.downloadAecModelData() をコール。
 - 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。
3. 一人称視点モードを起動します。



Viewer 拡張: Autodesk.DiffTool Extension



Viewer 拡張: Autodesk.DiffTool Extension

1. 2つの派生ファイルを Viewer にロードします。
2. ロード完了後に ‘Autodesk.DiffTool’ Extension をロードします。
3. ロード時にオプション（すべて必須）を指定します。

primaryModels : ロード済みモデルの配列でプライマリです。

diffModels : ロード済みモデルの配列で差分比較対象のモデルです。

versionA : プライマリモデルのバージョン識別子です。

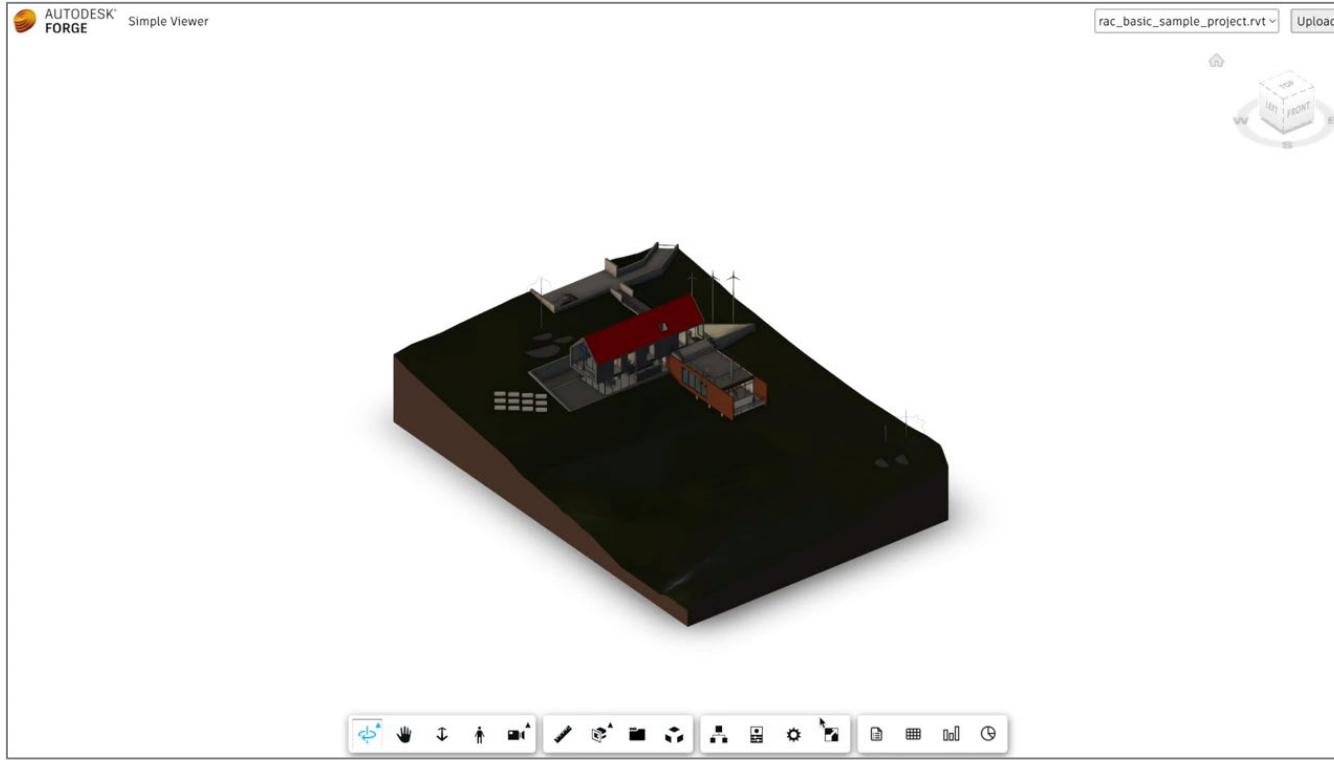
versionB : 差分比較対象モデルのバージョン識別子です。

mimeType : タイプに応じてロジックが最適化されます。

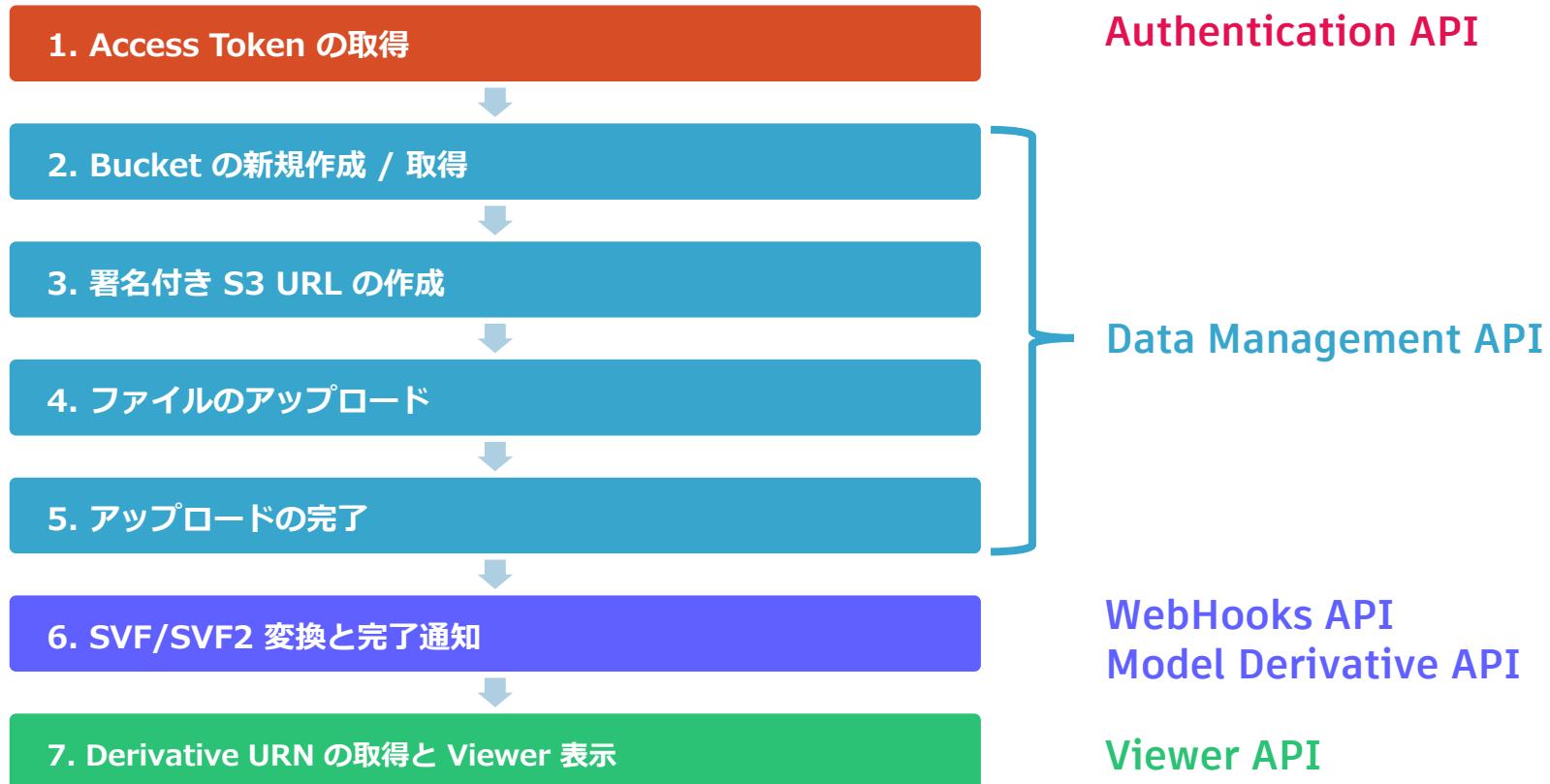
- ‘application/vnd.autodesk.revit’: Revit
- ‘application/vnd.autodesk.r360’: Revit
- ‘application/vnd.autodesk.fusion360’: For Fusion 360
- ‘application/vnd.autodesk.f3d’: For Fusion 360
- ‘application/vnd.autodesk.inventor.assembly’: For Inventor (IAM)
- ‘application/vnd.autodesk.navisworks’¹: For Navisworks (NWD)
- ‘application/vnd.autodesk.cad’: For IFC
- ‘application/vnd.autodesk.dxf’: For DXF
- ‘application/vnd.autodesk.autocad.dwg’: For DWG

Viewer 拡張: Extension の作成

- チュートリアルサイト
 - Dashboard (<https://tutorials.autodesk.io/tutorials/dashboard/>)

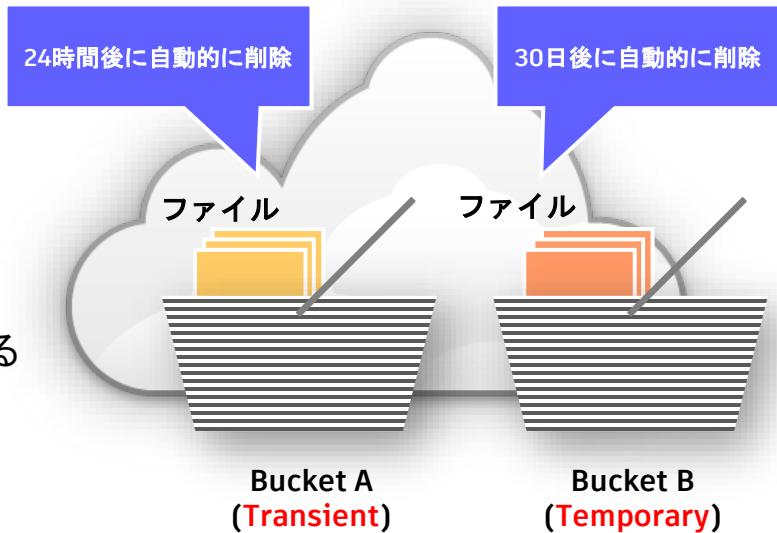


アップロードと表示のフロー (OSS)



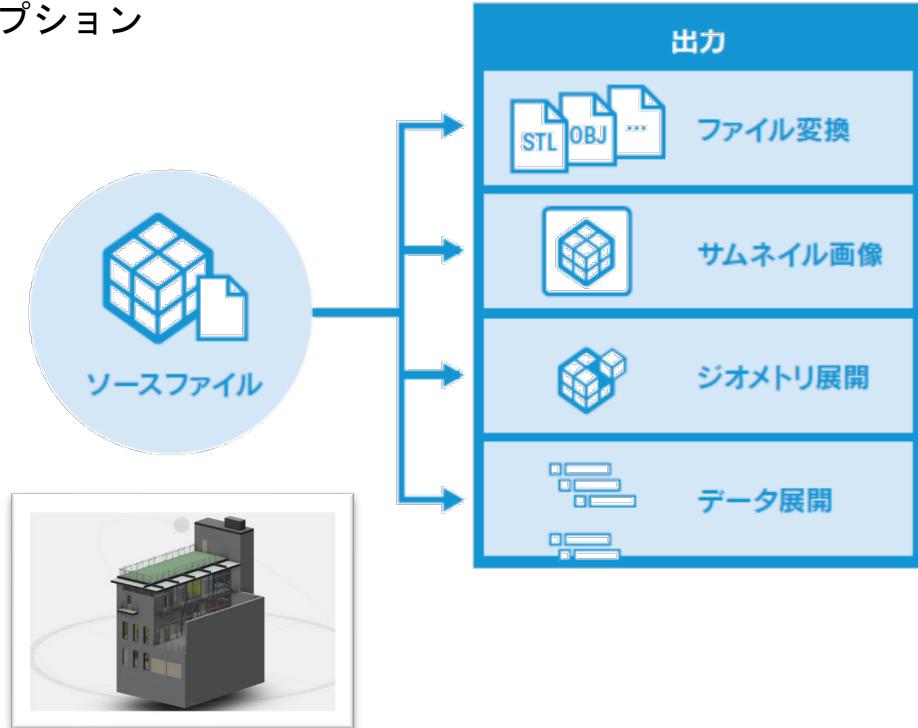
OSS Bucket と Object

- Bucket : OSS 上に作成するファイル保存のためのストレージ領域
 - グローバルで一意な名前を指定
 - DELETE メソッドもサポート済み
- Bucket ポリシー
 - transient : 保存されるファイルは24時間で自動消去
 - temporary : 保存されるファイルは30日で自動消去
 - persistent : 保存されるファイルは消去まで維持される
- Object : ファイルのこと



Model Derivative API で SVF / SVF2 に変換

- SVF/SVF2 形式に変換して Viewer で表示
 - POST: job エンドポイントの advanced オプション
 - generateMasterViews : true
 - 部屋、スペース、ゾーンの情報も変換して表示
 - timelineProperties : true
 - Navisworks Timelinear の情報も出力
- 他のデザインファイル形式に変換
- ジオメトリデータやモデル階層の展開
- サムネイル画像の生成



サンプルコード：チュートリアルサイト

- <https://tutorials.autodesk.io/>
- APS 開発者支援チームによる **Forge SDK** を使用。
- Simple Viewer
 - 50MB 以上のファイルをアップロードしたい場合
 - [RequestSizeLimit(500_000_000)] を追記で 500MB に拡張

```
[HttpPost()]
[RequestSizeLimit(500_000_000)]
public async Task<BucketObject> UploadAndTranslateModel([FromForm] UploadModelFor
{
    using (var stream = new MemoryStream())
    {
        await form.File.CopyToAsync(stream);
        stream.Position = 0;
        var obj = await _aps.UploadModel(form.File.FileName, stream);
        var job = await _aps.TranslateModel(obj.ObjectId, form.Entrypoint);
        return new BucketObject(obj.ObjectKey, job.Urn);
    }
}
```

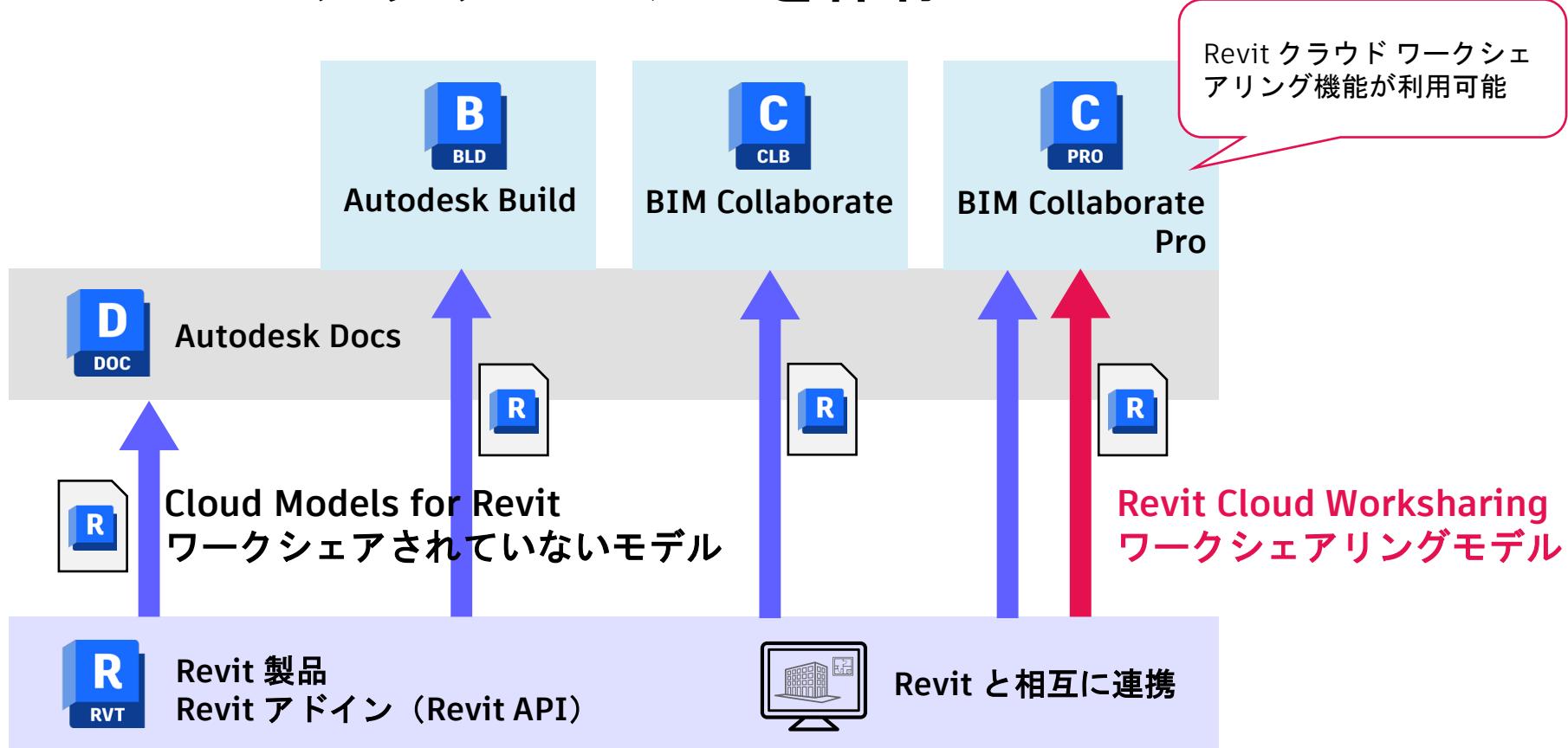




Revit API からのアプローチ



Revit からクラウドモデルを保存



Document.SaveAsCloudModel()

- 現在のモデルを Autodesk Docs にクラウド モデルとして保存します。
 - Cloud Models for Revit (ワークシェアされていないモデル)
 - Revit Cloud Worksharing (ワークシェアリングモデル)

C#

```
public void SaveAsCloudModel(  
    Guid accountId,  
    Guid projectId,  
    string folderId,  
    string modelName  
)
```

Authentication API
Data Management API

Web ブラウザから情報を取得する場合

- ACC プロジェクトの Insight ページに移動して URL をコピー。



https://acc.autodesk.com/insight/accounts/489c5*****1210b/projects/c99db*****fdce6/my-dashboard

AccountId

ProjectId

Web ブラウザから情報を取得する場合

- Autodesk Docs のターゲットのフォルダに移動して URL をコピー。

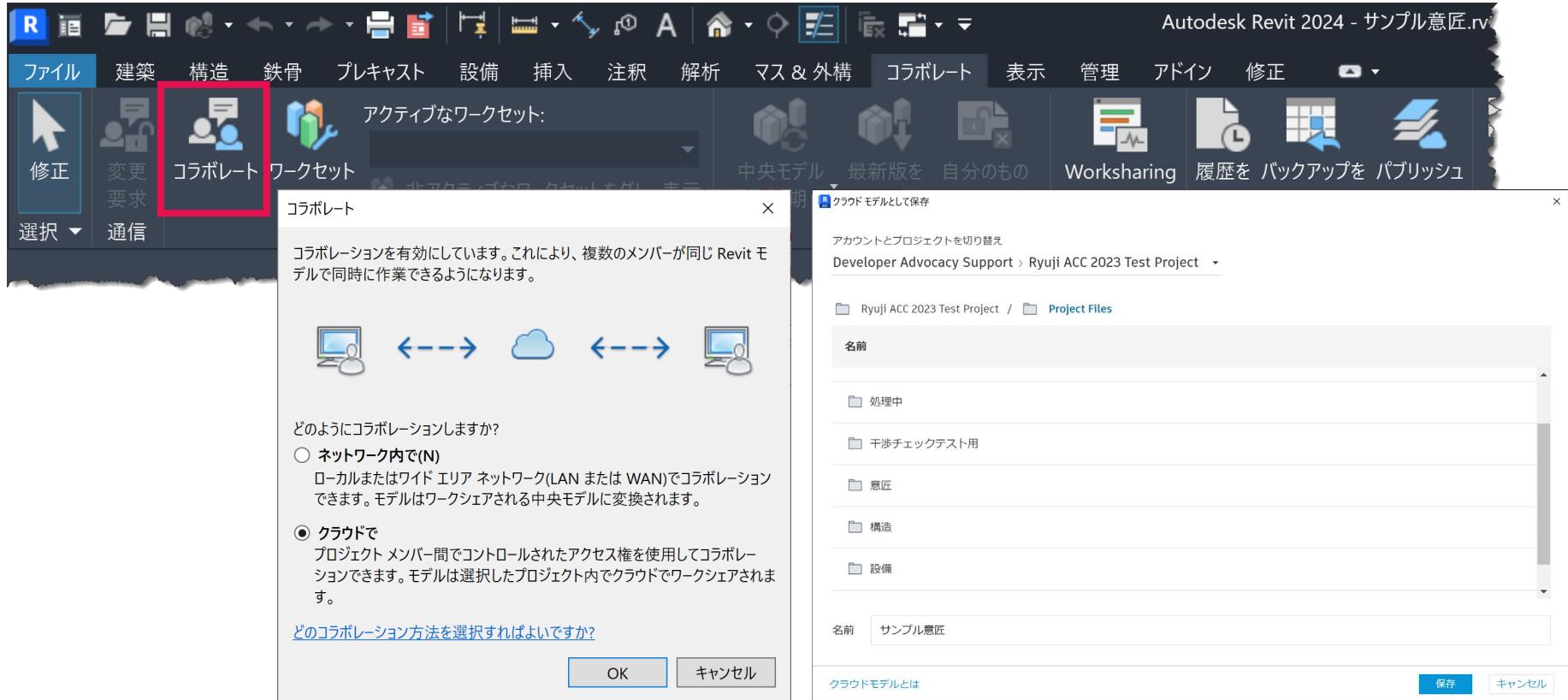


`https://acc.autodesk.com/docs/files/projects/c99db*****fdce6?`

`folderUrn=urn%3Aadsk.wipprod%3Afs.folder%3Aco.05yND7sIScayYFQTDRcg7g` FolderId

`&viewModel=detail&moduleId=folders`

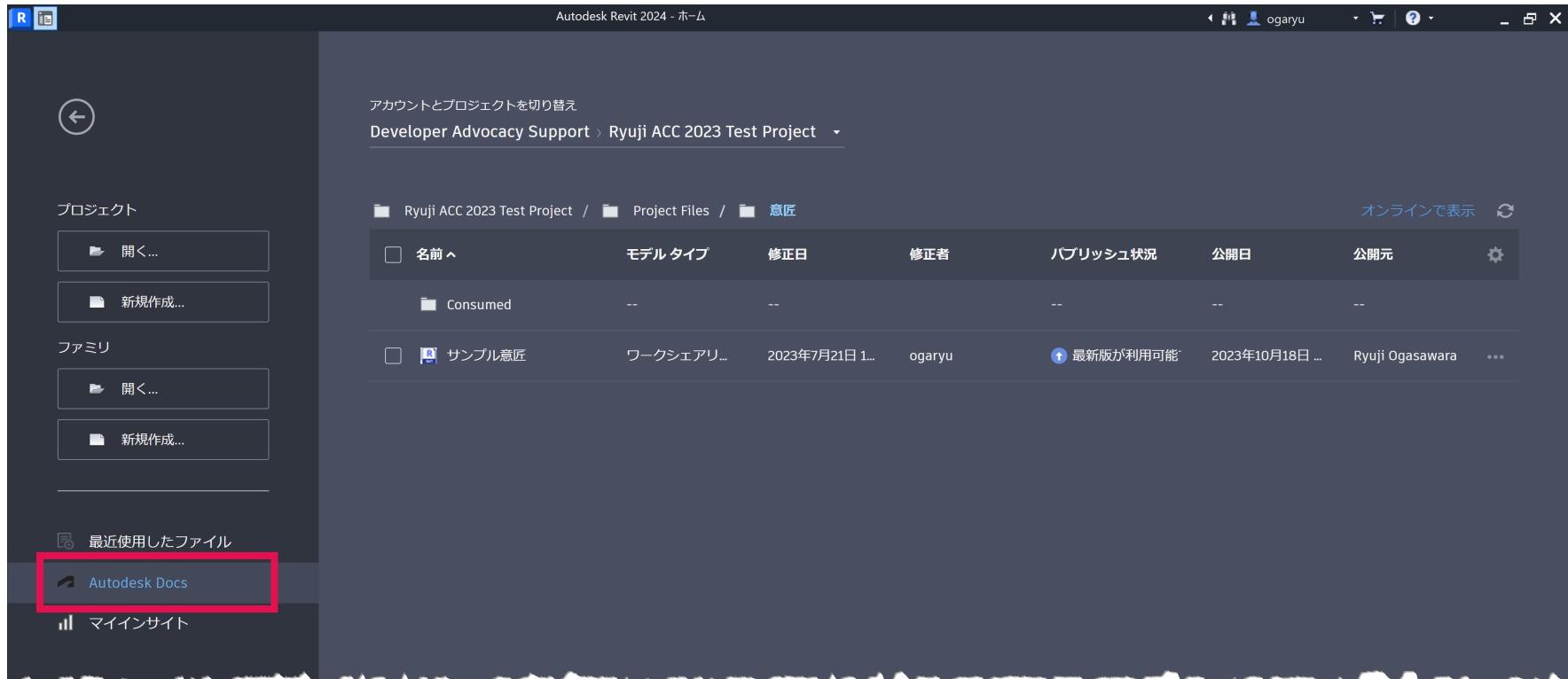
Revit でクラウドワークシェアリングを有効化する



Revit API でクラウドワークシェアリングを有効化

- Document.IsWorkshared
 - ワークシェアされているか判別
- Document.EnableWorksharing()
 - ローカルのモデルのワークシェアリングを有効化します。
- Document.EnableCloudWorksharing()
 - 既存のクラウド モデルをワークシェアリングモデルに変換します。
- Document.CanEnableCloudWorksharing()
 - この操作が指定されたモデルで有効かどうかを確認できます。

Revit でクラウドモデルを開く



Revit API でクラウドモデルを開く

- クラウドモデルを Application.OpenDocumentFile() で開くには、ModelPath が必要。
- ModelPath の取得方法
 - ModelPathUtils.ConvertCloudGUIDsToCloudPath(string region, Guid projectGuid, Guid modelGuid)
- プロジェクトの GUID とモデルの GUID の取得方法
 - Data Management API (+ Authentication API)

GET

projects/:project_id/folders/:folder_id/contents

https://help.autodesk.com/view/RVT/2024/JPN/?guid=Revit_API_Revit_API_Developers_Guide_Introduction_Application_and_Document_CloudFiles_html

Revit でクラウドモデルのリンクを再設定する

リンクを管理

Revit IFC CAD DWF 点群 地盤面 コーディネーション モデル PDF イメージ

リンク名	状態	参照タイプ	共有座標 保存	保存パス	パス タイプ	ローカル エイリアス
サンプル構造.rvt	ロード済み	アタッチ	<input type="checkbox"/>	サンプル構造.rvt	相対	

リンクを管理

Revit IFC CAD DWF 点群 地盤面 コーディネーション モデル PDF 1 / 1 ページ

リンク名	保存パス	パス タイプ	ローカル エイリアス
サンプル構造.rvt	Autodesk Docs://Ryuji ACC 2023 Test Project/Project Files/構造/サンプル構造.rvt	クラウド	

Revit API でクラウドモデルのリンクを再設定する

- RevitLinkType.Create(Document, ModelPath, RevitLinkOptions)
 - Revit リンクをクラウドモデルのモデルパスを指定して作成
- RevitLinkType.LoadFrom(ModelPath, WorksetConfiguration)
 - Revit リンクのロード先を再指定→クラウドモデルのモデルパスに変更

中央モデルと同期する

- Revit API 開発者用ガイド
 - ワークシェアリングされているファイルの管理
 - Document.SynchronizeWithCentral() メソッド
 - ・ 中央モデルからの変更を再ロードして、現在のセッションを最新の状態して、ローカルの変更を中央ファイルに戻して保存します。変更を行わない場合でも、中央ファイルへの保存は実行されます。
 - SynchronizeWithCentralOptions パラメータ
 - ・ 現在のユーザが所有している要素またはワークセットを同期中に放棄する必要があるかどうかなど、実際の同期のオプションを設定します。

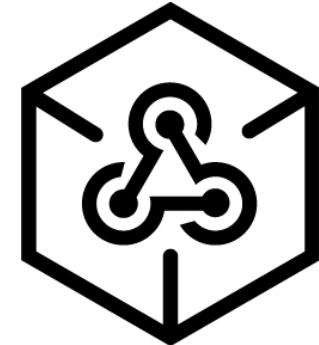
パブリッシュする (Data Management API)

POST https://developer.api.autodesk.com/data/v1/projects/:project_id/commands

```
POST "/data/v1/projects/b.c8112490-4e08-435c-994b-64fe60fea507/commands" -H 'content-type: application/vnd.api+json' -d '  
{  
  "jsonapi": { "version": "1.0" },  
  "data": {  
    "type": "commands",  
    "attributes": {  
      "extension": {  
        "type": "commands:autodesk.bim360:C4RModelPublish", "version": "1.0.0"  
      }  
    },  
    "relationships": {  
      "resources": {  
        "data": [ { "type": "items", "id": "urn:adsk.wipprod:dm.lineage:hPW2B1BbQG2L5HjCOh7Z8Q" } ]  
      }  
    }  
  }  
}
```

パブリッシュの完了通知

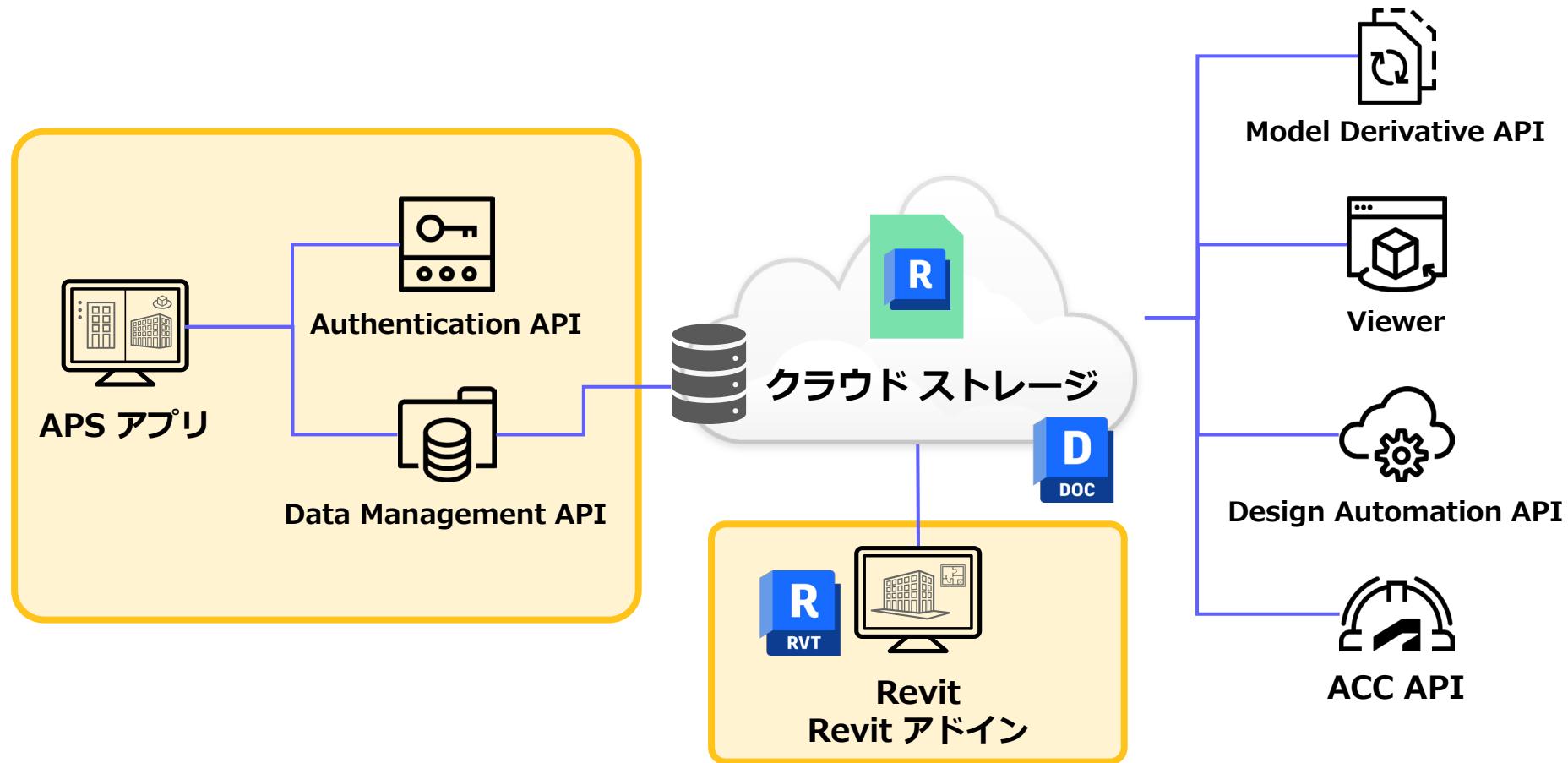
- Data Management API
 - GetPublishModelJob コマンドをポーリング
- Webhooks API
 - Revit Cloud Worksharing Events
 - model.sync : 中央モデルと同期後に通知。
 - model.publish : モデルがパブリッシュ開始・パブリッシュ中のキューにあるかどうか通知。
 - ACC/BIM 360 にモデルをアップロードした際、自動的に Model Derivative サービスが実行されて SVF/SVF2 変換が実行されます。この処理の完了通知を受け取れるようになりました。
 - extraction.finished : 変換処理終了後に通知
 - extraction.updated : 変換処理過程を通知
 - SVF 変換後、SVF2 変換後の 2 回呼び出されます。



Webhook now supports translation progress of BIM360 or ACC files

Forge Online - WebHooks : イベント通知と活用

Revit モデルをクラウドに移行する方法





Design & Make Platform