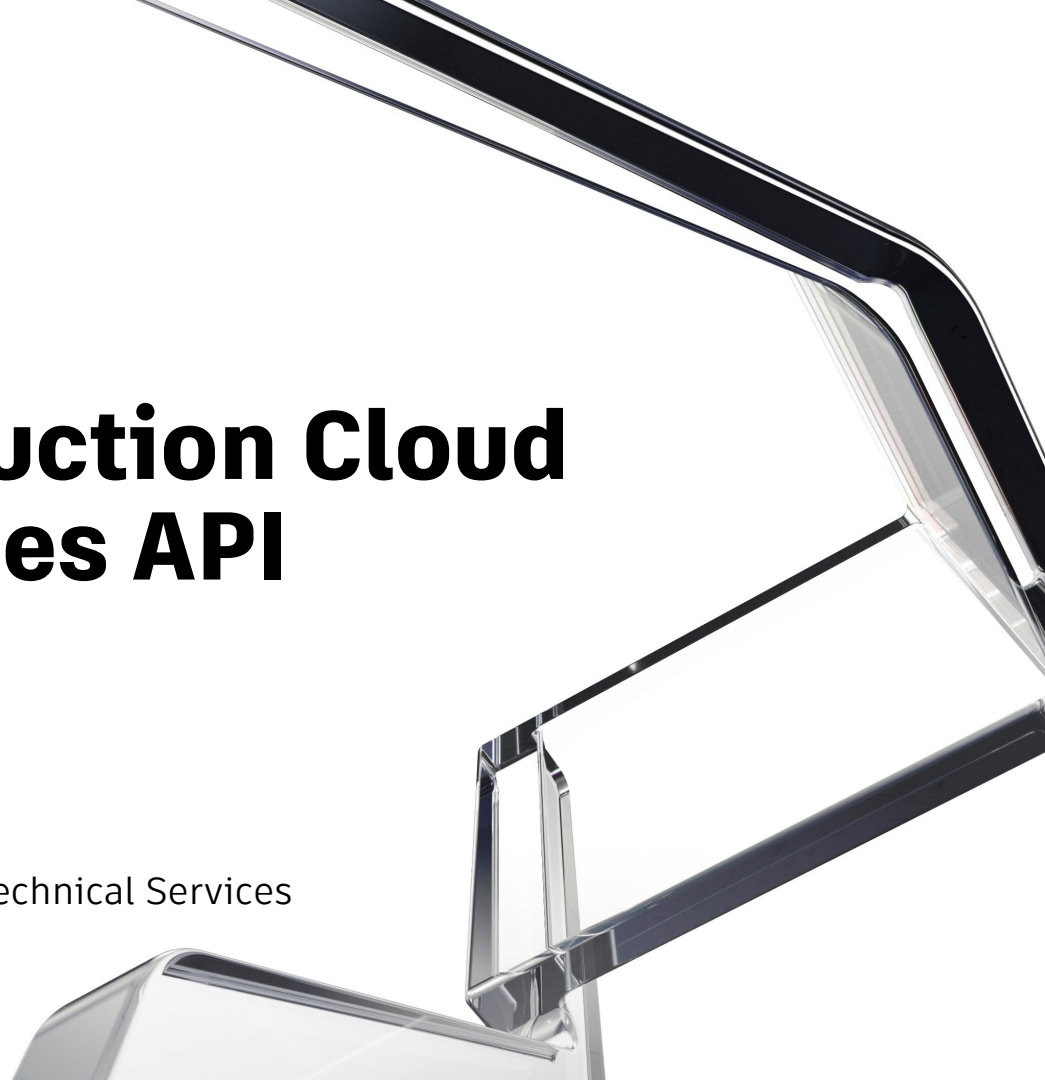
AUTODESK

Autodesk Construction Cloud & Model Properties API

Forge Data Days

原田 美佳子, Senior Manager AEC Developer Technical Services

小笠原 龍司, Developer Advocate



The presentations today may contain forward-looking statements about our strategies, products, future results, performance or achievements, financial, operational and otherwise, including statements about our strategic priorities, business model transition, and guidance for the fiscal year 2023 and beyond; our long term financial and operational goals; our M&A strategy; and our capital allocation initiatives. These statements reflect management's current expectations, estimates and assumptions based on the information currently available to us. These forward-looking statements are not guarantees of future performance and involve significant risks, uncertainties and other factors that may cause our actual results, performance or achievements to be materially different from results, performance or achievements expressed or implied by the forward-looking statements contained in these presentations, such as a failure to successfully integrate acquired businesses; developments in the COVID-19 pandemic and the resulting impact on our business and operations; general market, political, economic, and business conditions; complete transitions to new business model and markets; failure of the construction industry to grow as anticipated; failure to develop new products; failure to successfully expand adoption of our products; and failure of product changes to have the desired benefits.

A discussion of factors that may affect future results is contained in our most recent Form 10-K and Form 10-Q filings available at www.sec.gov, including descriptions of the risk factors that may impact us and the forward-looking statements made in these presentations. The forward-looking statements made in these presentations are being made as of the time and date of their live presentation. If these presentations are reviewed after the time and date of their live presentation, even if subsequently made available by us, on our website or otherwise, these presentations may not contain current or accurate information. We disclaim any obligation to update or revise any forward-looking statement based on new information, future events or otherwise.

Statements regarding planned or future development efforts for our products and services are not intended to be a promise or guarantee of future availability of products, services, or features but merely reflect our current plans and based on factors currently known to us. Purchasing decisions should not be made based upon reliance on these statements.

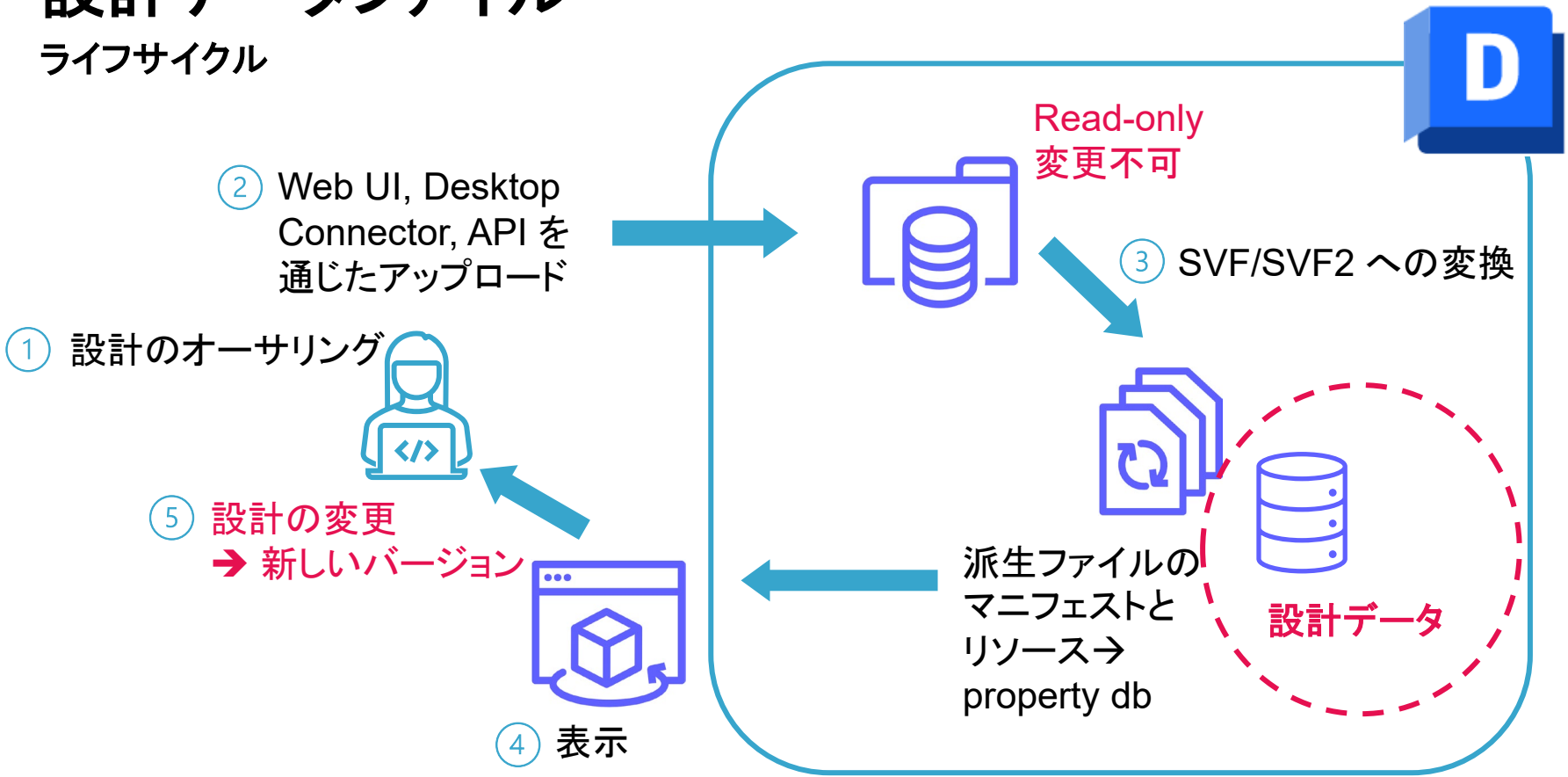
アジェンダ

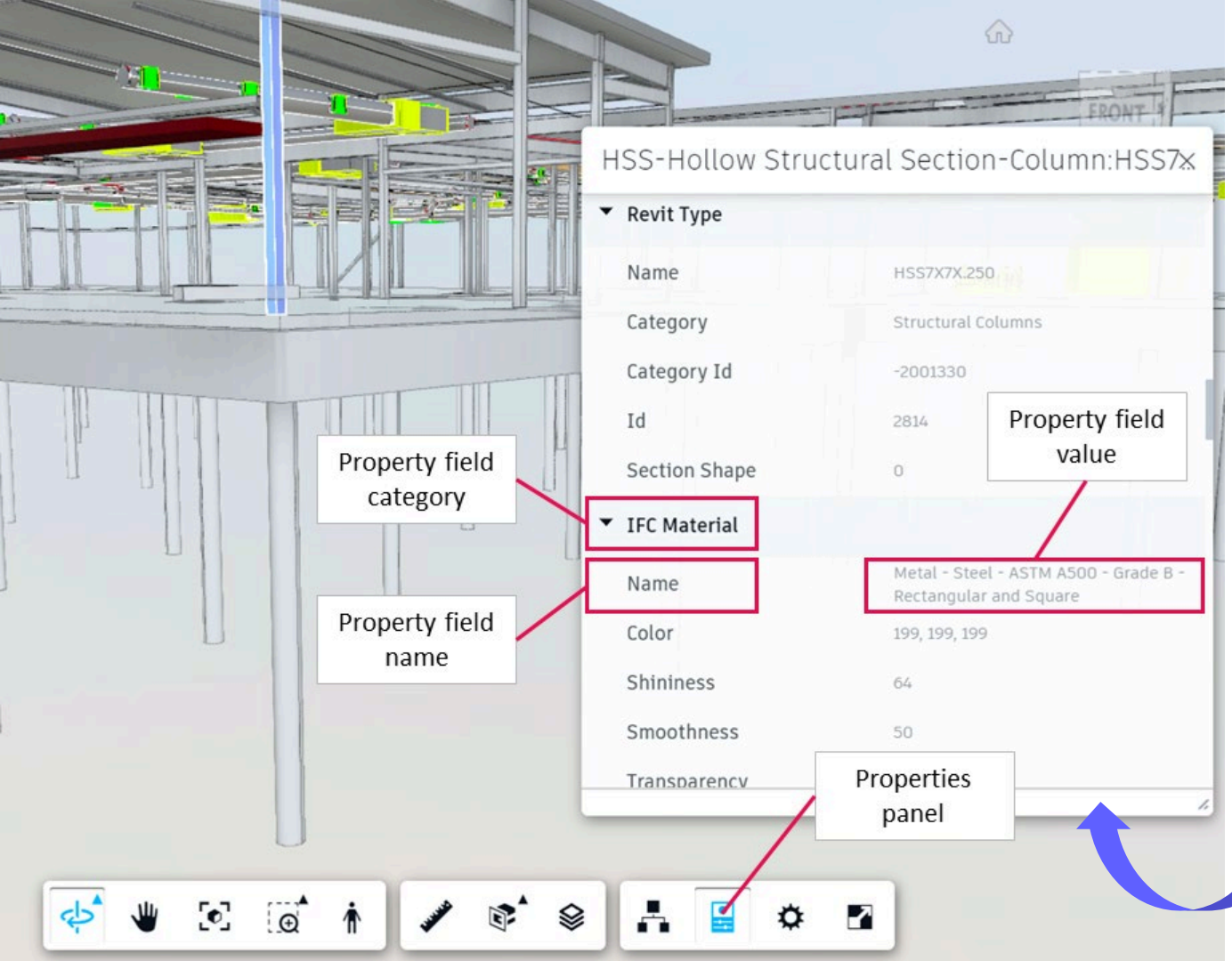
- Model Property API
- Data Connector API
- 今後の予定

Model Properties API

設計データファイル

ライフサイクル





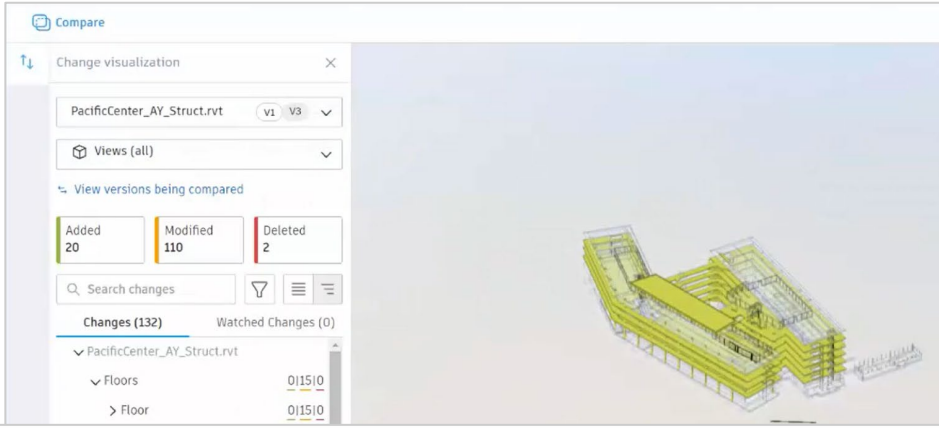
SVF2 モデルプロパティの分類（BIM）

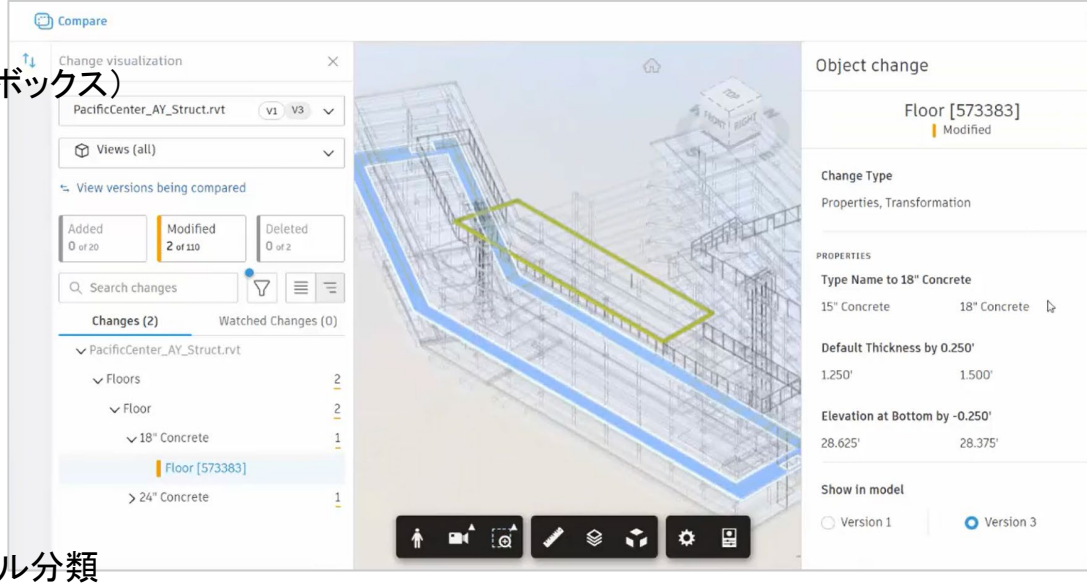
- カテゴリ
- 名前
- 型 (e.g., string, double)
- (オプショナル) 測定単位

propertyDb

Model Properties API

- 2022年2月リリース
- Autodesk/BIM 360 Docs 製品で動作。US と EMEA
- Model Derivative サービスで生成される PropertyDb に対して(svf2 変換後に)実行
  - **Index** – svf2 のプロパティ(+ バウンディングボックス)をクエリ, フィルタリング
  - **Diff** – Index + 2つのバージョンの比較
- 製品の機能でも利用されています。
  - Design Collaboration での変更内容の分析
- 今後のリリース予定
  - アセット – モデルとの統合
  - Model Coordination – 干渉チェック後のモデル分類



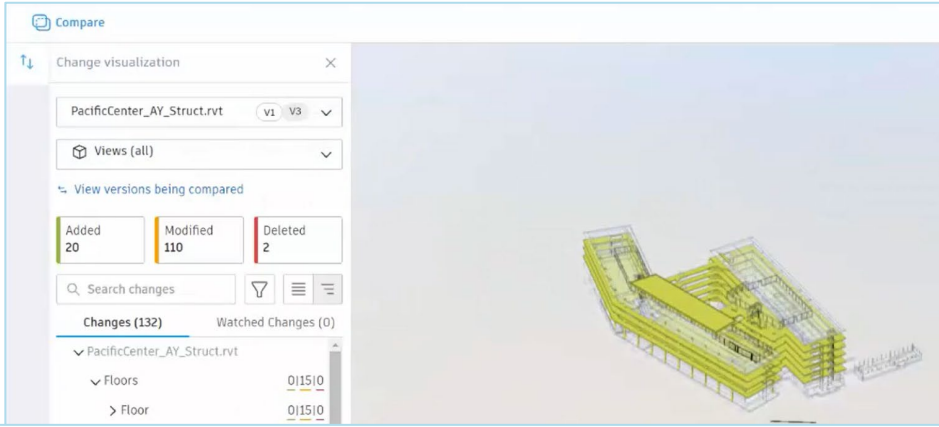


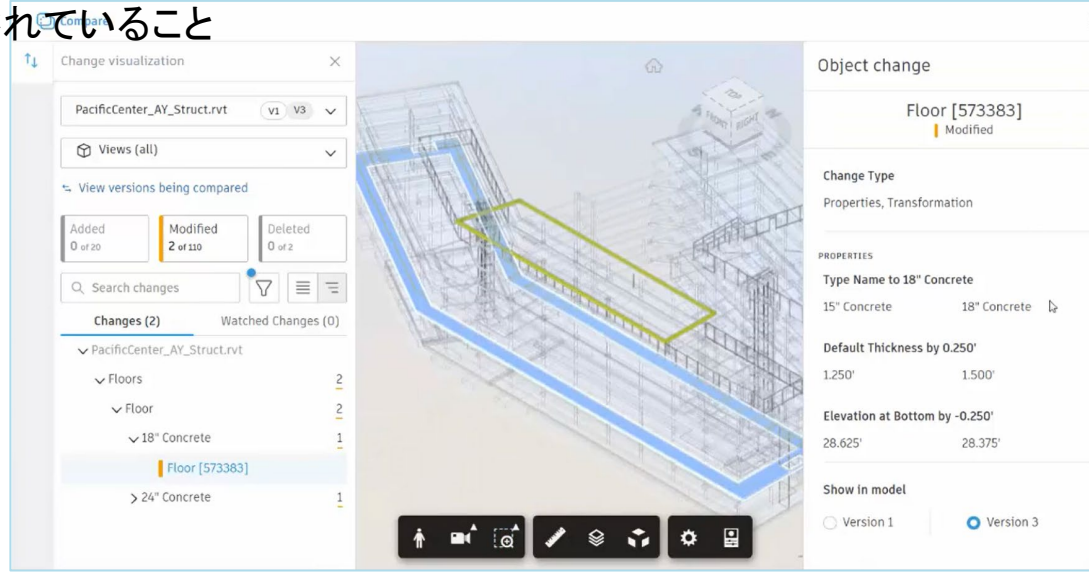
Model Properties API

サポートされているファイル形式

- **Index** – Model Derivative サービス/SVF2変換でサポートされているファイル

- **Diff** - 一意に識別可能な要素 ID が付与されていること
  - **RVT**
  - **DWG**
  - **NWC**:下記の製品から出力されたもの
    - Revit と AutoCAD ベースのバーチカル製品
  - **IFC**: 下記の製品から出力されたもの
    - AutoCAD Architecture, MEP, Civil 3D 2018+
    - ARCHCAD
    - Revit
    - MagiCAD for Revit
    - Tekla Structures





サンプル アプリケーション

Model Properties API

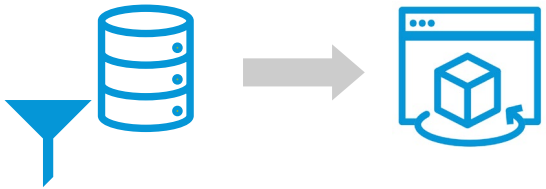
要素のフィルタリングとモデルの部分ロード

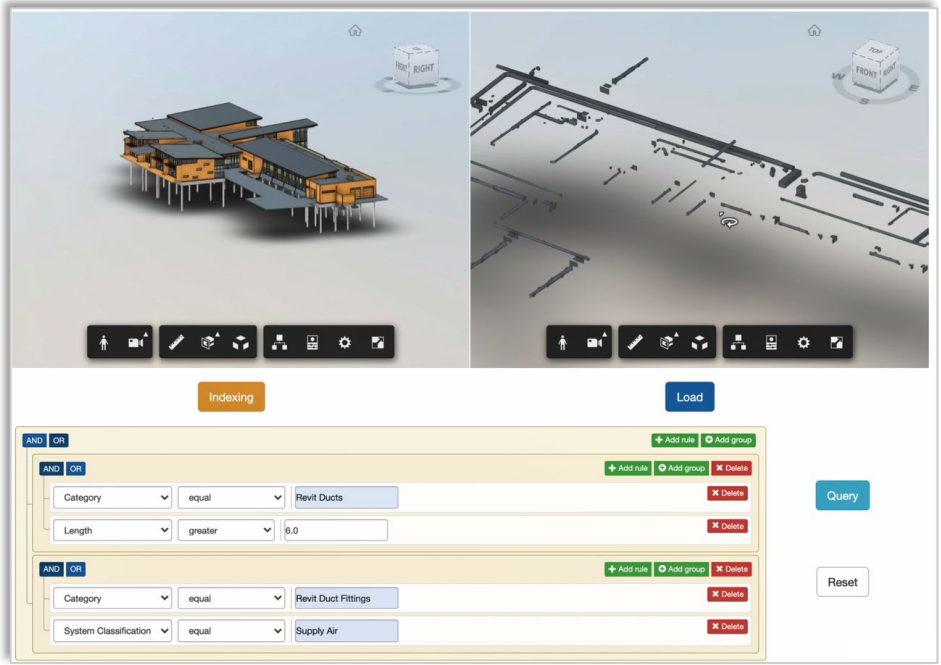
インデックス

概要: 要素をプロパティでフィルタリングします。プロパティは、長さや高さなどの幾何学的な情報も含みます。フィルタ条件は二項演算の形式で表現され、AND / ORで組み合わせることができます。結果はForge Viewerで可視化されます。

ソースコード: GitHub サンプル

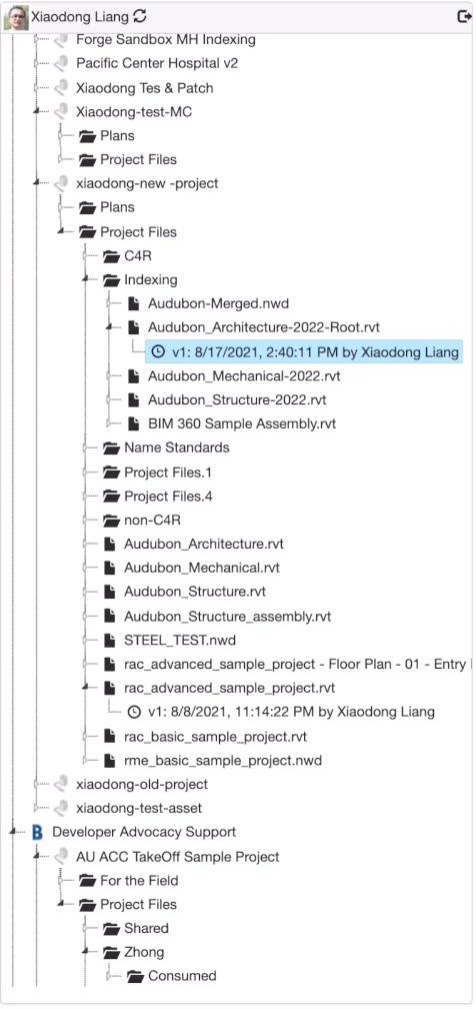
開発者: Xiaodong Liang, Autodesk

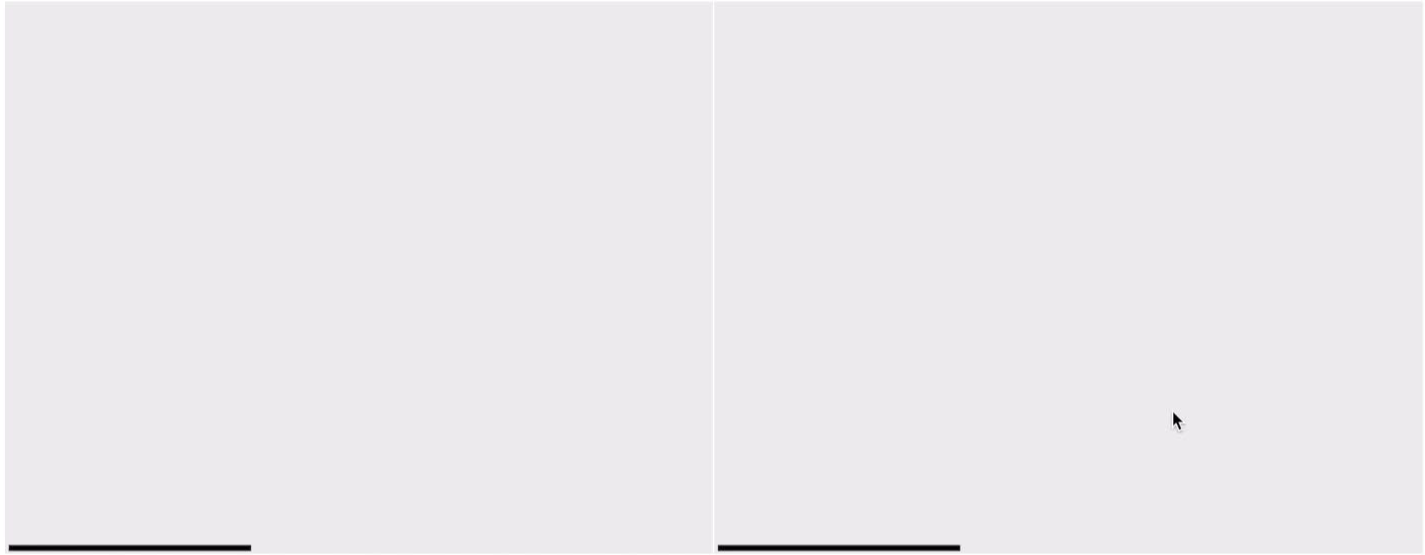




AUTODESK FORGE

Elements Filtering and Partial Loading





Indexing

Load

AND OR + Add rule + Add group

Level equal dummy ✖ Delete

AND OR + Add rule + Add group ✖ Delete

Category equal ✖ Delete

Right Arrow equal ○ False ● True ✖ Delete

Free Size less 0.11 ✖ Delete

Query

Reset

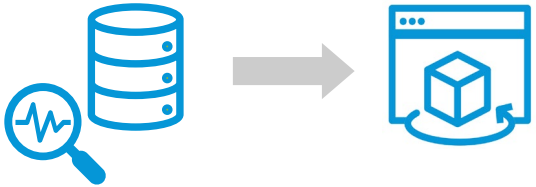
2つのバージョンの比較

差分比較

概要: モデルの2つのバージョンを比較し、違いをForge Viewer 上で可視化します。
ジオメトリとプロパティの違いを抽出します。
要素は、追加、削除、変更に分類されます。

ソースコード: GitHub サンプル

開発者: Xiaodong Liang, Autodesk



AUTODESK FORGE

Models Differences

Xiaodong Liang

- Autodesk Forge Partner Development
  - Daimler POC Project
  - Endymion
  - Flow Test
  - Forge Concert Hall
  - Forge Infraworks
  - Pacific Center Hospital v2
  - Xiaodong Tes & Patch
  - Xiaodong-test-MC
  - xiaodong-new -project
    - Plans
    - Project Files
      - C4R
      - Name Standards
      - Project Files.1
      - Audubon_Architecture.rvt
      - Audubon_Mechanical.rvt
        - v1: 8/4/2021, 7:16:41 PM by Xiaod
        - v2: 8/4/2021, 7:20:54 PM by Xiao
        - v3: 8/7/2021, 5:28:32 PM by Xiao
      - Audubon_Structure.rvt
        - v1: 5/27/2021, 8:27:30 PM by Xiac
        - v2: 8/9/2021, 3:43:06 PM by Xiao
      - STEEL_TEST.nwd
      - rac_advanced_sample_project.rvt
        - v1: 8/8/2021, 11:14:22 PM by Xiao
      - rac_basic_sample_project.rvt
        - v1: 8/9/2021, 11:57:47 AM by Xiac
        - v2: 8/9/2021, 12:02:38 PM by Xiac
        - v3: 8/9/2021, 12:38:16 PM by Xiac
      - rme_basic_sample_project.nwd
  - xiaodong-old-project

All Added

All Removed

All Changed

Added Items

name	category	level	RC	Free Size	Si
No matching records found					

Removed Items

name	category	level	RC	Free Size	Si
No matching records found					

Changed Items

name	geometry changed?	property change
No matching re		

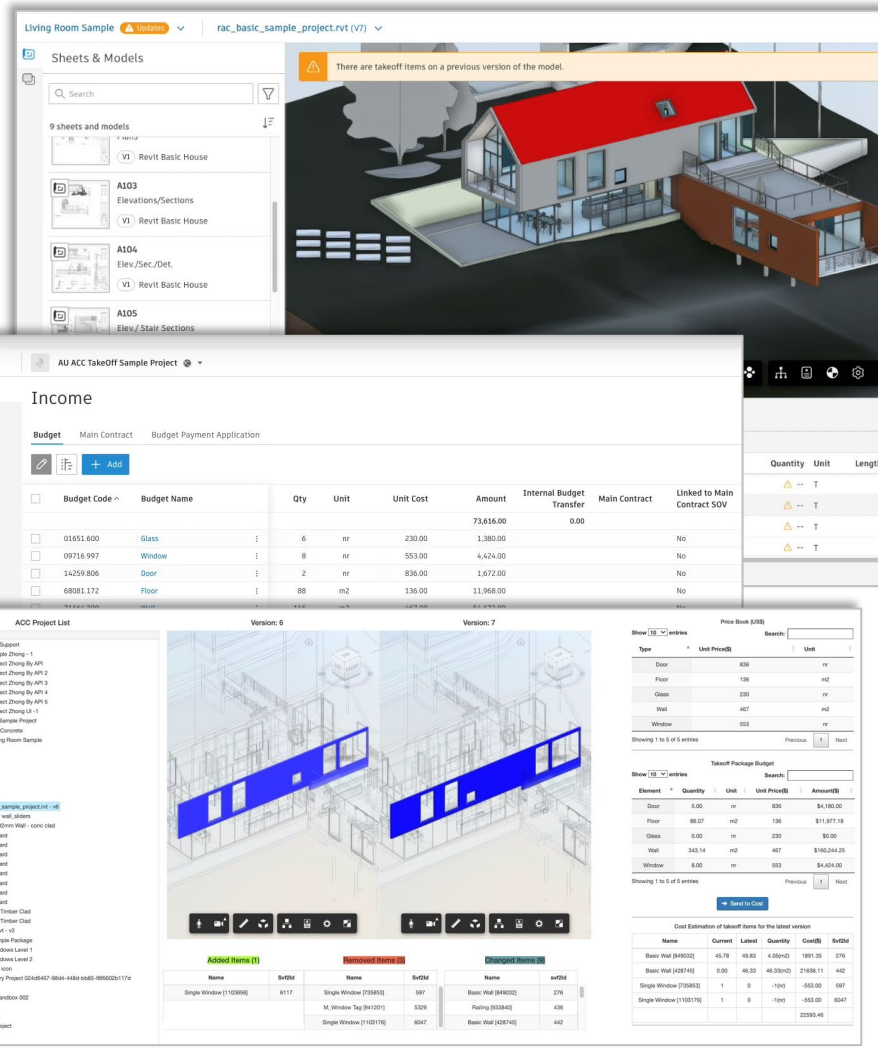
Takeoff / Costでの変更の解析

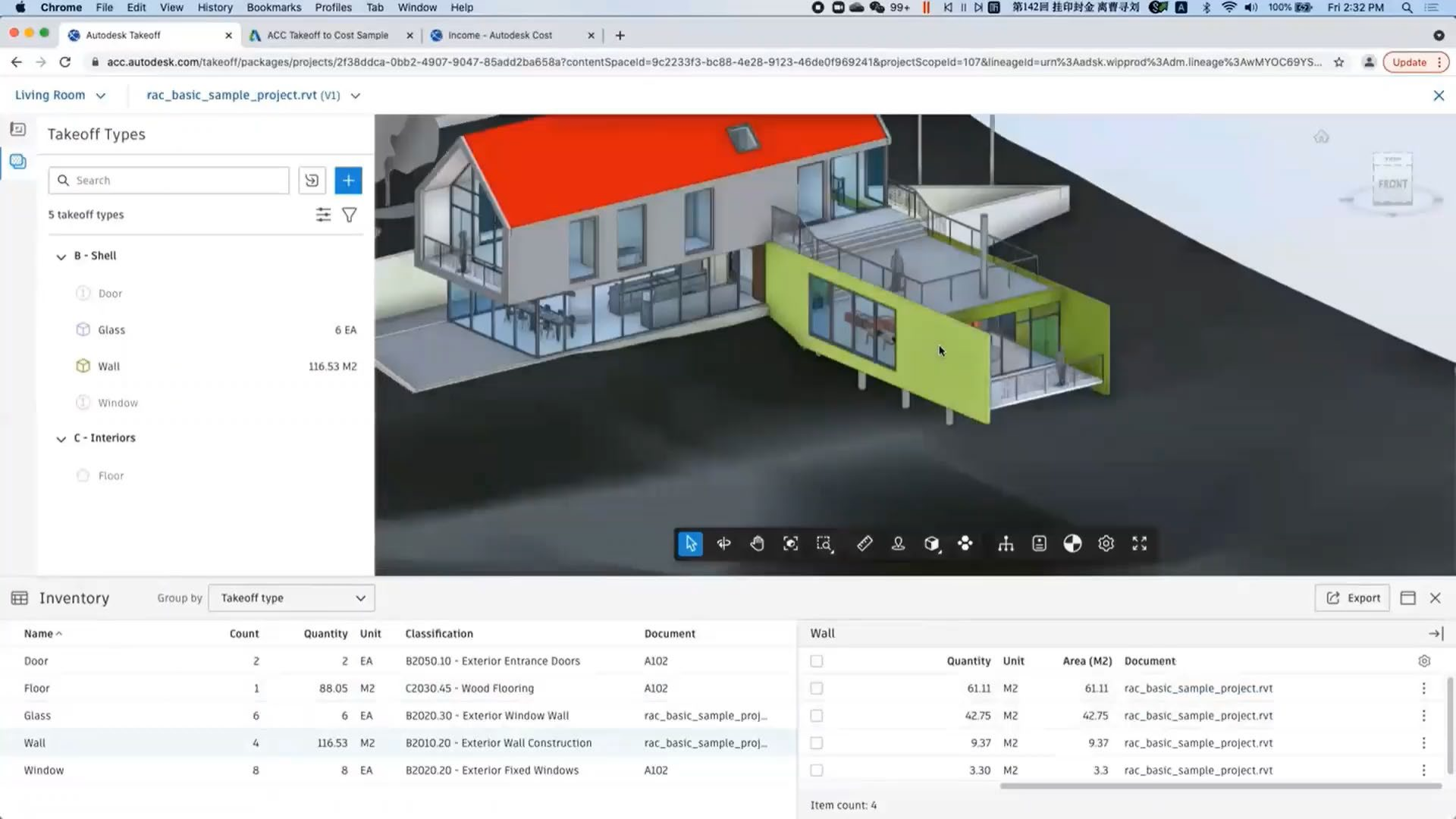
積算における差分解析アプリケーション

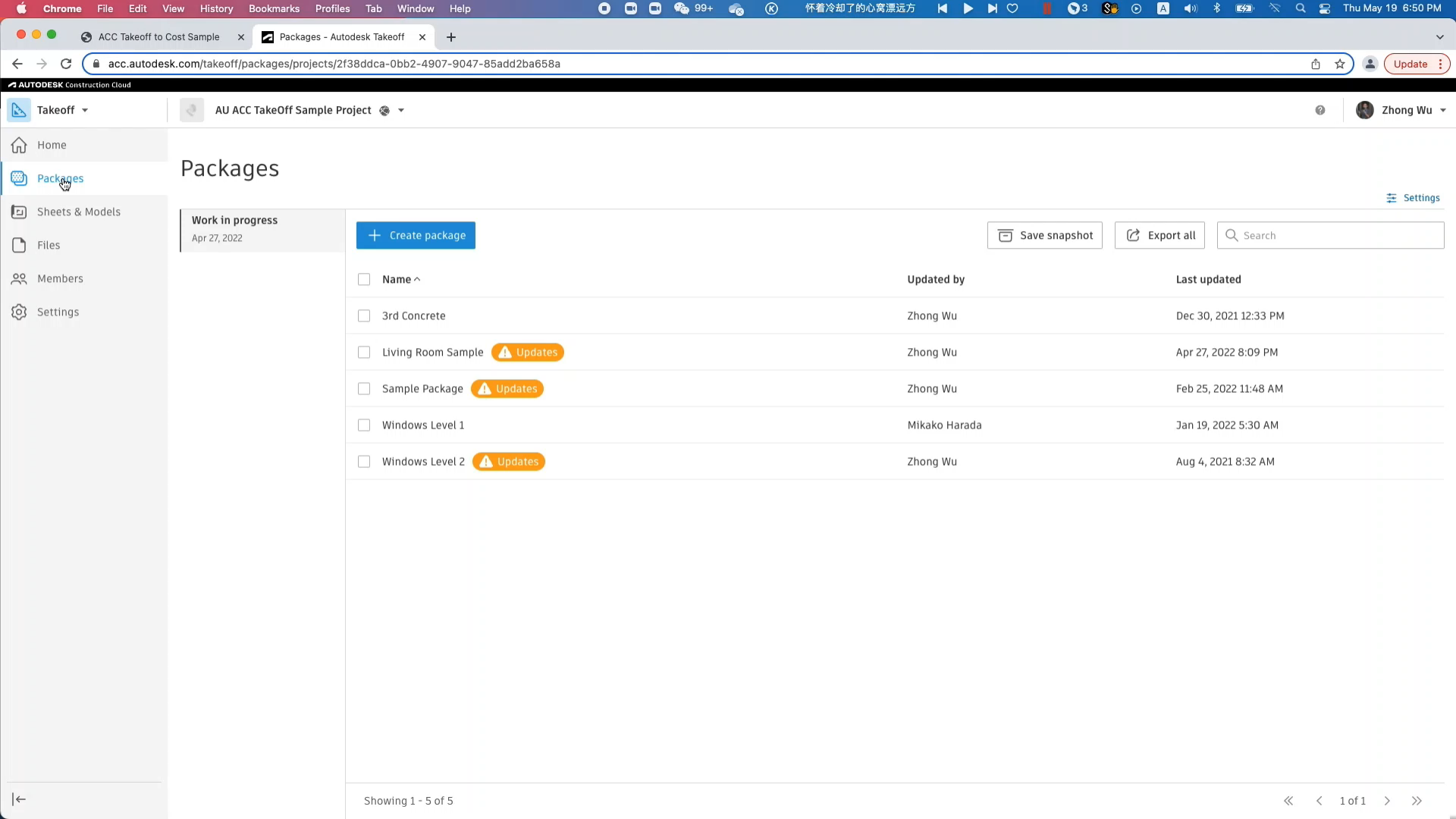
概要: 積算の項目の変更を特定し、モデルの現在と以前のバージョンを比較し、Forge Viewerで違いを可視化します。Cost モジュールで予算のデータを更新します。

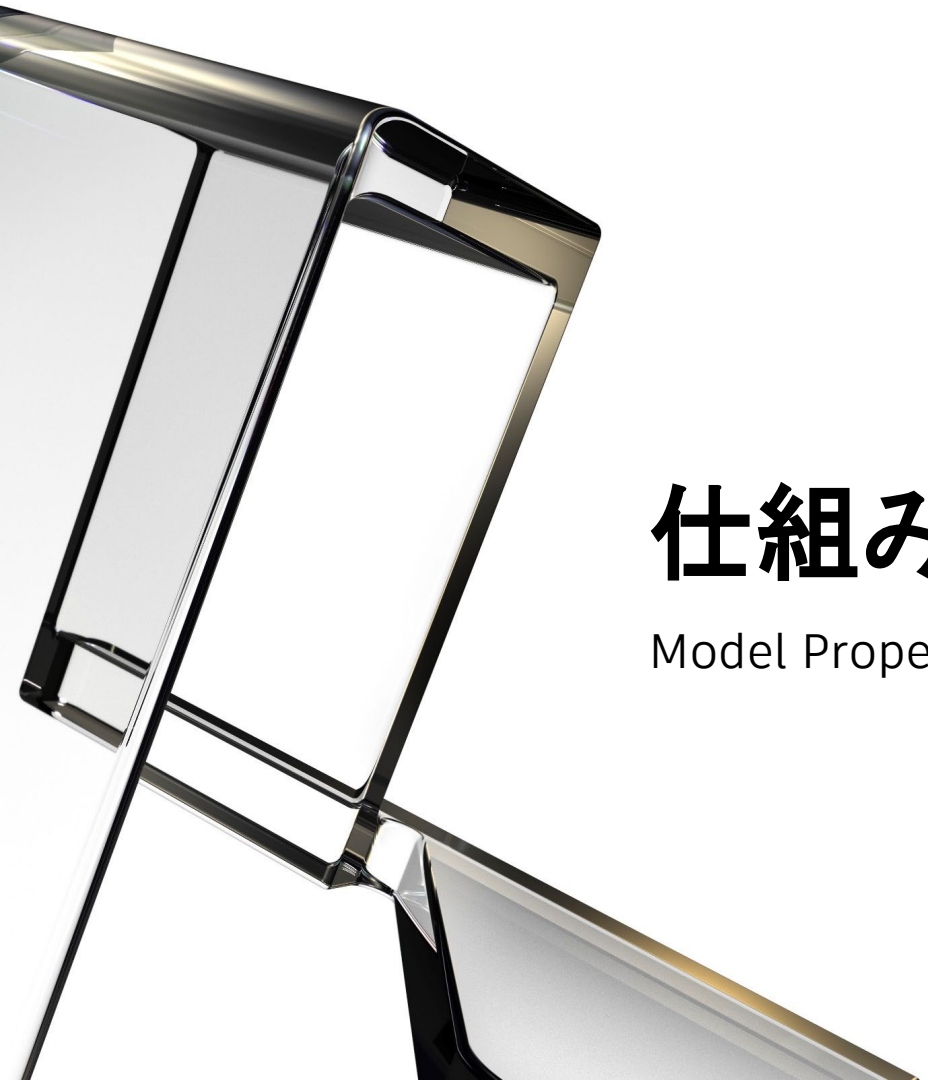
ソースコード: GitHub sample

開発者: Zhong Wu, Autodesk




Chrome File Edit View History Bookmarks Profiles Tab Window Help
Fri 2:32 PM
Autodesk Takeoff
ACC Takeoff to Cost Sample
Income - Autodesk Cost
acc.autodesk.com/takeoff/packages/projects/2f38ddca-0bb2-4907-9047-85add2ba658a?contentSpaceId=9c2233f3-bc88-4e28-9123-46de0f969241&projectScopeId=107&lineageId=urn%3Aadsk.wipprod%3Adm.lineage%3AwMYOC69YS...
Update
Living Room
rac_basic_sample_project.rvt (V1)
Takeoff Types
Search
5 takeoff types
B - Shell
Door
Glass 6 EA
Wall 116.53 M2
Window
C - Interiors
Floor
FRONT
Inventory
Group by Takeoff type
Export
Name Count Quantity Unit Classification Document
Door 2 2 EA B2050.10 - Exterior Entrance Doors A102
Floor 1 88.05 M2 C2030.45 - Wood Flooring A102
Glass 6 6 EA B2020.30 - Exterior Window Wall rac_basic_sample_proj...
Wall 4 116.53 M2 B2010.20 - Exterior Wall Construction rac_basic_sample_proj...
Window 8 8 EA B2020.20 - Exterior Fixed Windows A102
Wall
Quantity Unit Area (M2) Document
61.11 M2 61.11 rac_basic_sample_project.rvt
42.75 M2 42.75 rac_basic_sample_project.rvt
9.37 M2 9.37 rac_basic_sample_project.rvt
3.30 M2 3.3 rac_basic_sample_project.rvt
Item count: 4


Chrome File Edit View History Bookmarks Profiles Tab Window Help
Thu May 19 6:50 PM
ACC Takeoff to Cost Sample
Packages - Autodesk Takeoff
acc.autodesk.com/takeoff/packages/projects/2f38ddca-0bb2-4907-9047-85add2ba658a
Update
AUTODESK Construction Cloud
Takeoff
AU ACC TakeOff Sample Project
Zhong Wu
Home
Packages
Sheets & Models
Files
Members
Settings
Packages
Settings
Work in progress
Apr 27, 2022
Create package
Save snapshot
Export all
Search
Name
Updated by
Last updated
3rd Concrete
Zhong Wu
Dec 30, 2021 12:33 PM
Living Room Sample
Updates
Zhong Wu
Apr 27, 2022 8:09 PM
Sample Package
Updates
Zhong Wu
Feb 25, 2022 11:48 AM
Windows Level 1
Mikako Harada
Jan 19, 2022 5:30 AM
Windows Level 2
Updates
Zhong Wu
Aug 4, 2021 8:32 AM
Showing 1 - 5 of 5
1 of 1



仕組みの解説

Model Properties API

Model Properties API の仕組み

基本的なワークフロー

Model Derivative

DM/Docs

UI, Desktop Connector または API を通じてファイルを Docs にアップロード

Model Derivative

ファイルは svf/svf2 に変換されます。

Docs では変換は自動的にトリガされます。

設計データ

PropertyDb

プロパティのデータベース、マニフェスト、リソース

Model Properties API の仕組み

基本的なワークフロー

Model Derivative

Model Properties

DM/Docs

UI, Desktop Connector または API を通じてファイルを Docs にアップロード

Model Derivative

ファイルは svf/svf2 に変換されます。

Docs では変換は自動的にトリガされます。

設計データ

PropertyDb

プロパティのデータベース、マニフェスト、リソース

Index

インデックスを作成すると3つのダウンロード可能な JSON データが生成されます:

- Manifest
- Fields
- Properties

Query

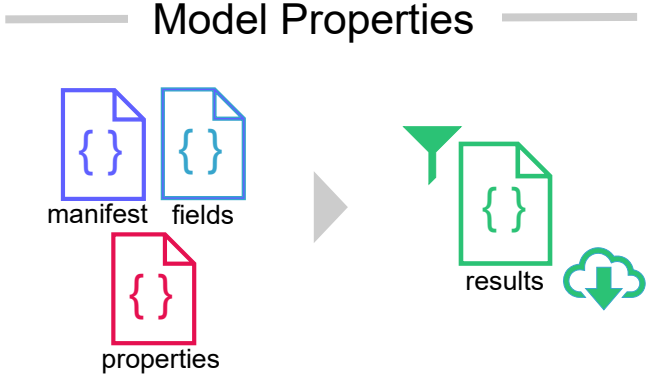
結果: クエリフィルタにマッチする JSON データが生成されます。

Model Properties API の仕組み

生成されるリソース

リソース	型	説明
manifest	JSON	シードファイルの詳細と、インデックス行の生成に使用される svf2 propertyDb の情報を含む、インデックスまたはクエリのためのマニフェスト
fields	NDJSON	インデックスやクエリのために抽出された一意のフィールド（プロパティタイプ）のセット
properties	NDJSON	インデックス化したオブジェクトプロパティの実際のデータ
results	NDJSON	クエリを実行した結果のオブジェクトプロパティ。

NDJSON = new-line delimited JSON（改行区切りの JSON）



インデックスを作成すると3つのダウンロード可能な JSON データが生成されます：

- Manifest
- Fields
- Properties

結果: クエリフィルタにマッチする JSON データが生成されます。

Model Properties API の仕組み

Index エンドポイント

	エンドポイント	
Index	POST	**indexes:batch-status**
	GET	indexes/**:indexId**
	GET	indexes/:indexId/**manifest**
	GET	indexes/:indexId/**fields**
	GET	indexes/:indexId/**properties**
Query	POST	indexes/:indexId/**queries**
	GET	indexes/:indexId/queries/**:queryId**
	GET	indexes/:indexId/queries/:queryId/**properties**




- 8 つのエンドポイント

Model Properties API の仕組み

Index 作成のエンドポイント

	エンドポイント	
Index	POST	**indexes:batch-status**
	GET	indexes/**:indexId**
	GET	indexes/:indexId/**manifest**
	GET	indexes/:indexId/**fields**
	GET	indexes/:indexId/**properties**
Query	POST	indexes/:indexId/**queries**
	GET	indexes/:indexId/queries/**:queryId**
	GET	indexes/:indexId/queries/:queryId/**properties**




- 基本的な Index の作成 – “遅延実行”
  - 初回 – インデックス作成ジョブを開始して結果をキャッシュします。
  - 2回目以降 – キャッシュを利用。
  - キャッシュは最後に使用した日から30日間保持されます。
- 進捗状況のポーリング
  - state: PROCESSING, FINISHED, FAILED
- レスポンスの JSON は同一です。
- stats: オブジェクトの数
- ダウンロード可能な3つのjson.gzリソースの作成
  - Manifest, fields, properties

例：インデックスの作成 – POST indexes:batch-status

```
curl --request POST 'https://developer.api.autodesk.com/construction/index/v2/projects/f83c      ...      /indexes:batch-status' \
     --header 'Authorization: Bearer ****' \
     --header 'Content-Type: application/json' \
     --data-raw '{
       "versions": [
           {
               "versionUrn": "urn:adsk.wipprod:fs.file:vf.DyTWutcvTcOLUNUARxcTzQ?version=4"
           }
       ]
     }'
```

リクエスト

```
[
    {
        "projectId": "f83cef12-deef-4771-9feb-4f85643e3c46",
        "indexId": "qTmPiKJZ7siqxkTNpWGANw",
        "type": "INDEX",
        "state": "PROCESSING",
        "selfUrl": "https://developer.api.autodesk.com/construction/index/v2/projects/f83cef12-deef-4771-9feb-4f8564
        "versionUrns": [
            "urn:adsk.wipprod:fs.file:vf.DyTWutcvTcOLUNUARxcTzQ?version=4"
        ],
        "updatedAt": "2021-08-19T08:21:13.8771187+00:00",
        "retryAt": "2021-08-27T14:28:28.8382067+00:00",
        "stats": null,
        "manifestUrl": null,
        "fieldsUrl": null,
        "propertiesUrl": null
    }
]
```

レスポンス

例：進捗状況のポーリング - GET indexes/:indexId

```
curl --request GET 'https://developer.api.autodesk.com/construction/index/v2/projects/  ···  /indexes/qTmPiKJZ7siqxkTNpWGANw' \
     --header 'Authorization: Bearer ****'
```

リクエスト

レスポンス


```
{
    "projectId": "f83cef12-deef-4771-9feb-4f85643e3c46",
    "indexId": "qTmPiKJZ7siqxkTNpWGANw",
    "type": "INDEX",
    "state": "FINISHED",
    "selfUrl": "https://developer.api.autodesk.com/construction/index/v2/proje ··· /indexes/qTmPiKJZ7siqxkTNpWGANw",
    "versionUrns": [
        "urn:adsk.wipprod:fs.file:vf.DyTWutcvTcOLUNUARxcTzQ?version=4"
    ],
    "updatedAt": "2021-08-19T08:21:13.8771187+00:00",
    "retryAt": "2021-08-27T14:31:55.1444684+00:00",
    "stats": {
        "objects": 33097
    },
    "manifestUrl": "https://developer.api.autodesk.com/construction/index/v2/pro ··· 46/indexes/qTmPiKJZ7siqxkTNpWGANw/manifest",
    "fieldsUrl": "https://developer.api.autodesk.com/construction/index/v2/proje ··· /indexes/qTmPiKJZ7siqxkTNpWGANw/fields",
    "propertiesUrl": "https://developer.api.autodesk.com/construction/index/v2/p ··· 3c46/indexes/qTmPiKJZ7siqxkTNpWGANw/properties"
}
```

Model Properties API の仕組み

Index ダウンロードのエンドポイント

	エンドポイント	
Index	POST	**indexes:batch-status**
	GET	indexes/**:indexId**
	GET	indexes/:indexId/**manifest**
	GET	indexes/:indexId/**fields**
	GET	indexes/:indexId/**properties**
Query	POST	indexes/:indexId/**queries**
	GET	indexes/:indexId/queries/**:queryId**
	GET	indexes/:indexId/queries/:queryId/**properties**

- (オプショナル) ダウンロード
  - manifest
  - fields
  - Properties

Manifest (.json)


```
{
    "schema": "2.0.0",
    "projectId": "f83cef12-deef-4771-9feb-4f85643e3c46",
    "status": "Succeeded",
    "createdAt": "2021-07-23T08:56:07.0868303+00:00",
    "seedFiles": [
        {
            "lineageId": "a19f7db",
            "lineageUrn": "urn:adsk.wipprod:dm.lineage:DyTWutcvTcOLUNUARxcTzQ",
            "versionUrn": "urn:adsk.wipprod:fs.file:vf.DyTWutcvTcOLUNUARxcTzQ?version=4",
            "databases": [
                {
                    "id": "3747dccf",
                    "offsets": "urn:adsk.viewing:fs.file:dXJuOmFkc2sud2l  ...  yc2lvbj04/output/Resource/objects_offs.json.gz",
                    "attributes": "urn:adsk.viewing:fs.file:dXJuOmFkc2su  ...  dmVyc2lvbj04/output/Resource/objects_attrs.json.gz"
                    "values": "urn:adsk.viewing:fs.file:dXJuOmFkc2sud2l  ...  c2lvbj04/output/Resource/objects_vals.json.gz",
                    "mapping": "urn:adsk.viewing:fs.file:dXJuOmFkc2sud2l  ...  yc2lvbj04/output/Resource/objects_avs.json.gz",
                    "ids": "urn:adsk.viewing:fs.file:dXJuOmFkc2sud2lwcHJ  ...  vbj04/output/Resource/objects_ids.json.gz"
                }
            ],
            "views": [
                {
                    "id": "e7fda9d5",
                    "urn": "urn:adsk.wipprod:fs.file:vf.DyTWutcvTcOLUNUARxcTzQ?version=4",
                    "is3d": true,
                    "viewableName": "{3D}",
                    "viewableId": "0935d8b2-149b-4a0d-b816-863f0d595a20-000bcd64",
                    "viewableGuid": "00cd2da3-fbfa-44a9-7a33-cad0bc4720cb"
                },
                {
                    "id": "12fcb372",
                    "urn": "urn:adsk.wipprod:fs.file:vf.DyTWutcvTcOLUNUARxcTzQ?version=4",
                    "is3d": true,
                    "viewableName": "New Construction",
                    "viewableId": "c884ae1b-61e7-4f9d-0001-719e20b22d0b-00120bb2",
                    "viewableGuid": "4a966c2a-ead6-65c3-4f98-273dd7543047"
                }
            ]
        },
    ],
    "errors": [],
    "stats": {
        "objects": 33097,
        "contentLength": 1881318
    }
}
```


Lineage と Version

SVF2 Prop DB リソース URNs

Viewables

Index のバイトサイズと行数

Index Fields (json.gz)


```
{"key":"p153cb174","category":"__name__","type":"String","name":"name","uom":null}
{"key":"p74a9a490","category":"__document__","type":"String","name":"schema_name","uom":null}
{"key":"p137c14f2","category":"__document__","type":"String","name":"schema_version","uom":null}
{"key":"p1490bcea","category":"__document__","type":"Boolean","name":"is_doc_property","uom":null}
{"key":"p5eddc473","category":"__category__","type":"String","name":"Category","uom":null}
{"key":"p00723fa6","category":"Identity Data","type":"String","name":"Design Option","uom":null}
{"key":"pe8094f29","category":"Other","type":"String","name":"Project Issue Date","uom":null}
{"key":"p50756a0d","category":"Other","type":"String","name":"Client Name","uom":null}
{"key":"p32791eb0","category":"Other","type":"String","name":"Project Address","uom":null}
{"key":"pbf75ced9","category":"Other","type":"String","name":"Project Name","uom":null}
{"key":"p8213f1ad","category":"Other","type":"String","name":"Project Number","uom":null}
{"key":"pa7275c45","category":"__categoryId__","type":"Integer","name":"CategoryId","uom":null}
{"key":"p93e93af5","category":"__parent__","type":"DbKey","name":"parent","uom":null}
{"key":"p1d45bc4f","category":"Dimensions","type":"Double","name":"Computation Height","uom":"ft"}
{"key":"pe01bd7ef","category":"Extents","type":"String","name":"Scope Box","uom":null}
{"key":"p9fffb245","category":"Materials and Finishes","type":"Integer","name":"Color","uom":null}
{"key":"p1b3b6224","category":"Materials and Finishes","type":"String","name":"Transparency","uom":null}
{"key":"pd9fcab30","category":"Materials and Finishes","type":"Boolean","name":"Glow","uom":null}
{"key":"pf62e5a3c","category":"Structural","type":"Double","name":"Structural Framing Length Roundoff","uom":"ft"}
```


フィールドキー → SQL カラム名	カテゴリ	型	名前	UOM (Unit of Measurement)

Basic Properties (json.gz)

Object ID, Index manifest JSONのキー
(diff Index の場合は Type と ChangeType)

インデックス フィールド プロパティの値

プロパティのハッシュ値 + ハッシュを計算するときに無視するプロパティ

SVF2 ジオメトリハッシュ

Viewable オブジェクトのバウンディングボックス min/max

オブジェクトを含む Viewables
(Viewables の manifest キー)

Model Properties API の仕組み

クエリのエンドポイント

	エンドポイント	
Index	POST	**indexes:batch-status**
	GET	indexes/**:indexId**
	GET	indexes/:indexId/**manifest**
	GET	indexes/:indexId/**fields**
	GET	indexes/:indexId/**properties**
Query	POST	indexes/:indexId/**queries**
	GET	indexes/:indexId/queries/**:queryId**
	GET	indexes/:indexId/queries/:queryId/**properties**

- クエリのビルドと実行
  - インデックスクエリは、カスタムJSONスキーマで記述。（AWS S3 Select文のフィルタ式に変換されます）
  - カラムを制限することができます。（別のヘッダーを持つ）エイリアスの使用も可能。
- 進捗状況のポーリング
  - state: PROCESSING, FINISHED, FAILED

Index Fields

```
// Forge viewer element display name field
{"key":"p153cb174","category":"__name__","type":"String","name":"name","uom":null}
// Revit category name field
{"key":"p20d8441e","category":"__category__","type":"String","name":"_RC","uom":null}
// Revit family name field
{"key":"p30db51f9","category":"__category__","type":"String","name":"_RFN","uom":null}
// Revit type name field
{"key":"p13b6b3a0","category":"__category__","type":"String","name":"_RFT","uom":null}
```

クエリのサンプル:
Revit の分類をカスタムのカラムで取得

クエリ

```
{
    "query": {
        "$and": [
            { "$notnull": "s.props.p20d8441e" },
            { "$notnull": "s.props.p30db51f9" },
            { "$notnull": "s.props.p13b6b3a0" },
            { "$gt": [{ "$count": "s.views" }, 0] }
        ]
    },
    "columns": {
        "s.svf2Id": true,
        "lmvName": "s.props.p153cb174",
        "revitCategory": "s.props.p20d8441e",
        "revitFamily": "s.props.p30db51f9",
        "revitType": "s.props.p13b6b3a0",
        "s.views": true
    }
}
```

Revit の分類を示す Row

0 以上のビュー配列

エイリアスを定義したカラム

対応する S3 SQL

```
select
    s.svf2Id,
    s.props.p153cb174 as lmvName,
    s.props.p20d8441e as revitCategory,
    s.props.p30db51f9 as revitFamily,
    s.props.p13b6b3a0 as revitType,
    s.views
from S3Object[*] s
where
    s.props.p20d8441e is not null and
    s.props.p30db51f9 is not null and
    s.props.p13b6b3a0 is not null and
    count(s.views) > 0
```

Model Properties API の仕組み

クエリの結果をダウンロードするエンドポイント

	エンドポイント	
Index	POST	**indexes:batch-status**
	GET	indexes/**:indexId**
	GET	indexes/:indexId/**manifest**
	GET	indexes/:indexId/**fields**
	GET	indexes/:indexId/**properties**
Query	POST	indexes/:indexId/**queries**
	GET	indexes/:indexId/queries/**:queryId**
	GET	indexes/:indexId/queries/:queryId/**properties**

- クエリの結果をダウンロード
  - クエリのリクエスト時の queryResultsUrl、またはクエリIDを使用して、送信されたクエリ式に一致するインデックス行をダウンロードします。
  - 結果: 改行区切りの NDJSON
  - プロパティインデックス行のサブセット
  - フォーマットは、先に見たプロパティと全く同じです。

Model Properties API の仕組み

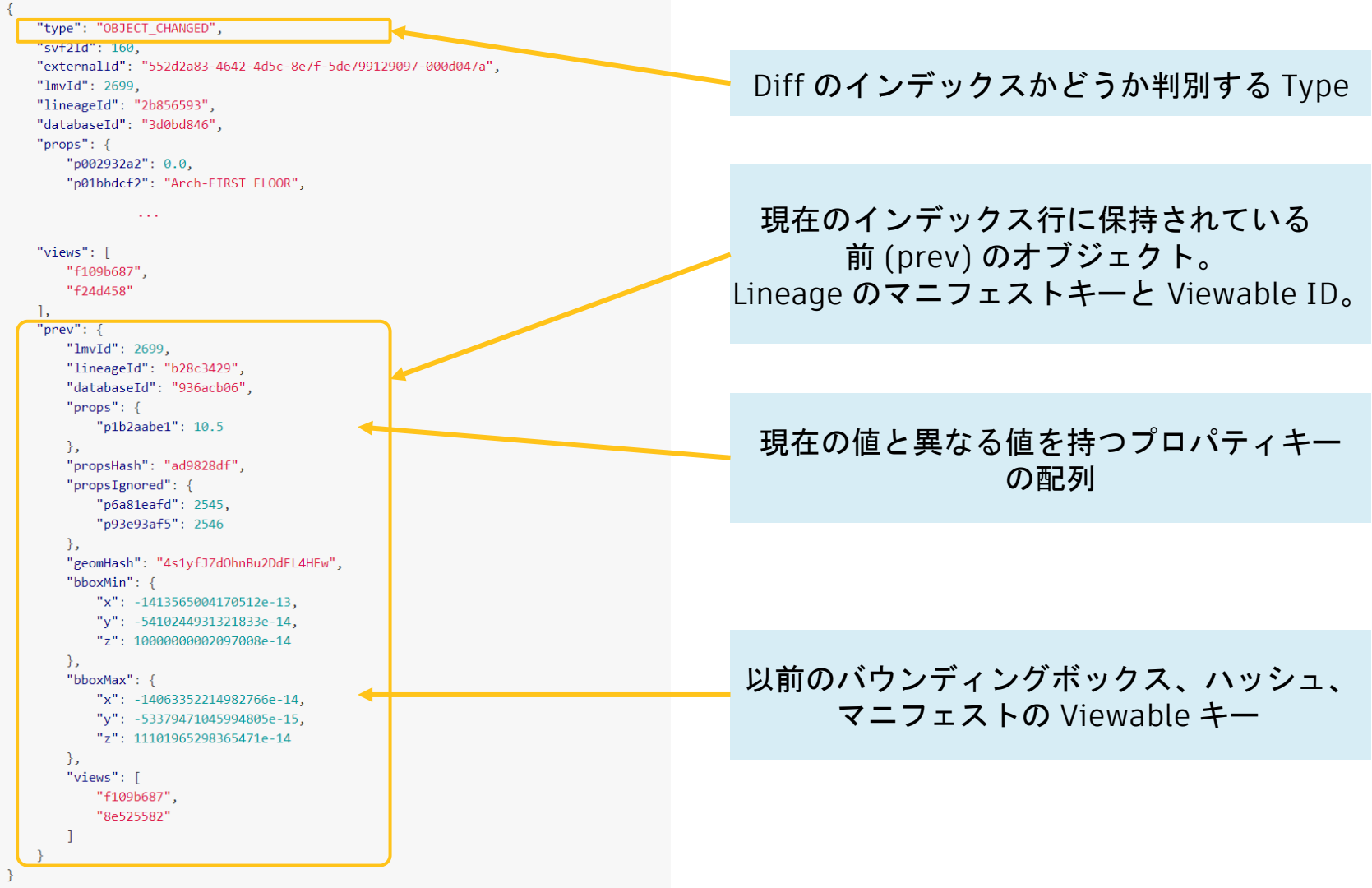
Diff エンドポイント

	エンドポイント	
Diff	POST	**diffs:batch-status**
	GET	diffs/**:diffId**
	GET	diffs/:diffId/**manifest**
	GET	diffs/:diffId/**fields**
	GET	diffs/:diffId/**properties**
Query	POST	diffs/:diffId/**queries**
	GET	diffs/:diffId/queries/**:queryId**
	GET	diffs/:diffId/queries/:queryId/**properties**

- Diff – Index 作成と同じ手順
- 比較する2つのバージョンの URN を指定:

```
{
  "diffs": [
    {
      "prevVersionUrn": "urn:adsk.w
      "curVersionUrn": "urn:adsk.wi
    }
  ]
}
```

- stats: add, removed, modified


{
"type": "OBJECT_CHANGED",
"svf2Id": 160,
"externalId": "552d2a83-4642-4d5c-8e7f-5de799129097-000d047a",
"lmvId": 2699,
"lineageId": "2b856593",
"databaseId": "3d0bd846",
"props": {
"p002932a2": 0.0,
"p01bbdcf2": "Arch-FIRST FLOOR",
...
"views": [
"f109b687",
"f24d458"
],
"prev": {
"lmvId": 2699,
"lineageId": "b28c3429",
"databaseId": "936acb06",
"props": {
"p1b2aabe1": 10.5
},
"propsHash": "ad9828df",
"propsIgnored": {
"p6a81eafd": 2545,
"p93e93af5": 2546
},
"geomHash": "4s1yfJZdOhnBu2DdFL4HEw",
"bboxMin": {
"x": -1413565004170512e-13,
"y": -5410244931321833e-14,
"z": 10000000002097008e-14
},
"bboxMax": {
"x": -14063352214982766e-14,
"y": -53379471045994805e-15,
"z": 11101965298365471e-14
},
"views": [
"f109b687",
"8e525582"
]
}
}
Diff のインデックスかどうか判別する Type
現在のインデックス行に保持されている
前 (prev) のオブジェクト。
Lineage のマニフェストキーと Viewable ID。
現在の値と異なる値を持つプロパティキー
の配列
以前のバウンディングボックス、ハッシュ、
マニフェストの Viewable キー

Basic インデックス行 vs. Diff インデックス行

インデックス行に含まれるフィールドの一覧

説明	現在のバージョン	前のバージョン
IDs	s.svf2Id s.**externalId**	
Change type, previous vs. current		s.**type** s.**changeType**
lineage version info, SVF2 database URNs	s.lmvId s.lineageId s.databaseId	s.**prev**.lmvId s.prev.lineageId s.prev.databaseId
Property values	s.props.* s.propsHash s.propsIgnored.*	s.prev.props.* s.prev.propsHash s.prev.propsIgnored.*
Geometry hash and bounding box values IF viewable	s.geomHash s.bboxMin.x s.bboxMin.y s.bboxMin.z s.bboxMax.x s.bboxMax.y s.bboxMax.z	s.prev.geomHash s.prev.bboxMin.x s.prev.bboxMin.y s.prev.bboxMin.z s.prev.bboxMax.x s.prev.bboxMax.y s.prev.bboxMax.z
Viewable keys IF viewable	s.views s.views[*i*]	s.prev.views s.prev.views[*i*]

JSON 抽象構文木 → S3 Select(AWS)

```
$not
$and
$or
$gt
$lt
$eq
$le
$ge
```

```
$like
$between
$in
$contains
$isnull
$notnull
$add
$sub
$mul
$div
```

```
$cat
$coalesce
$mod
$cast
$nullif
$date_add
$date_diff
$extract
$substring
$to_string
$to_timestamp
```

```
$char_length
$lower
$upper
$count
$sum
$avg
$min
$max
$trim
$utcnow
$case
```

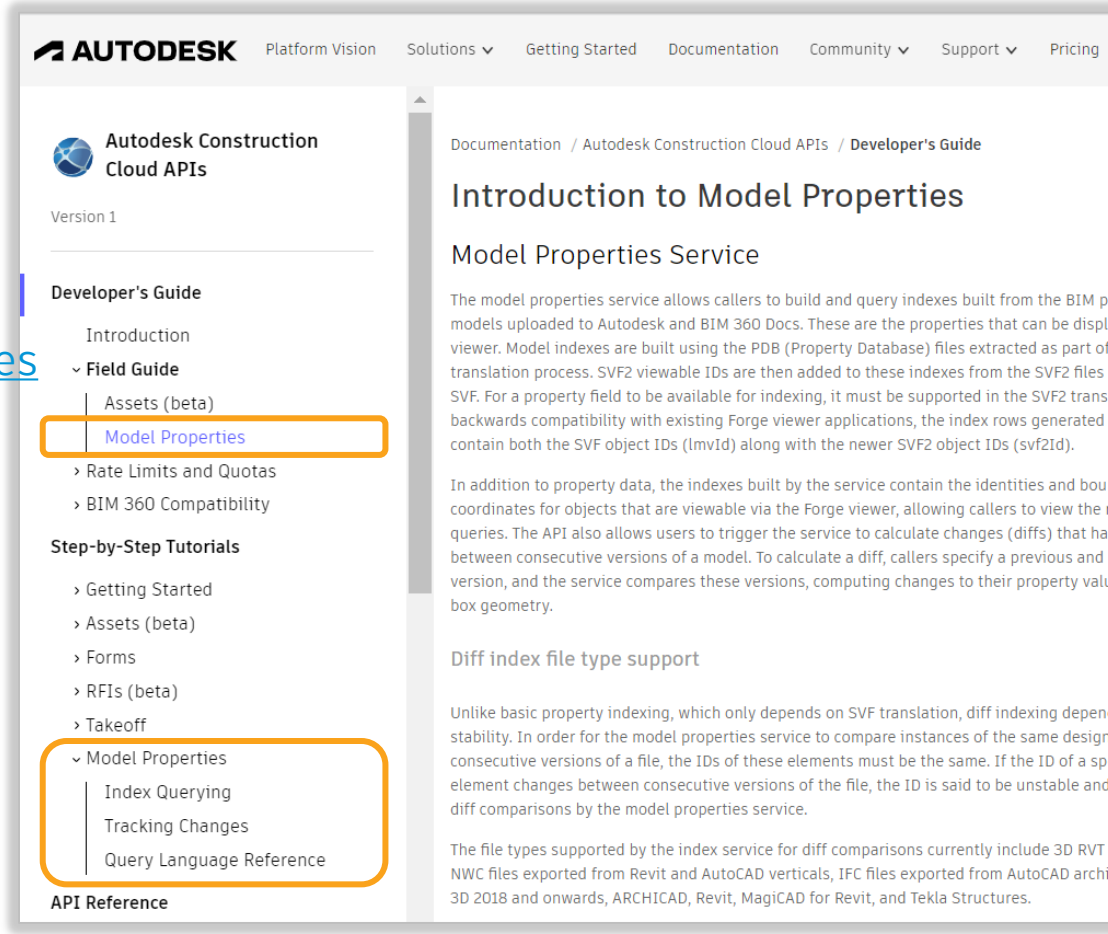
開発者リソース

Model Properties API

開発者リソース

ドキュメンテーション

- フィールドガイド
  - Introduction to Model Properties
- Step-by-Step チュートリアル
  - Index Querying
  - Tracking Changes
  - Query Language Reference
- レファレンスガイド
  - Index
  - Diff



開発者リソース

- GitHub コードサンプル
  - Postman Collection (3つの Step-by-Stepチュートリアルに対応)
  - Model Properties API Walkthrough in PowerShell Core (クエリ言語の理解)
  - Element Filtering and Partial Model Load (Viewer 統合)
  - Compare Two Versions (Viewer 統合)

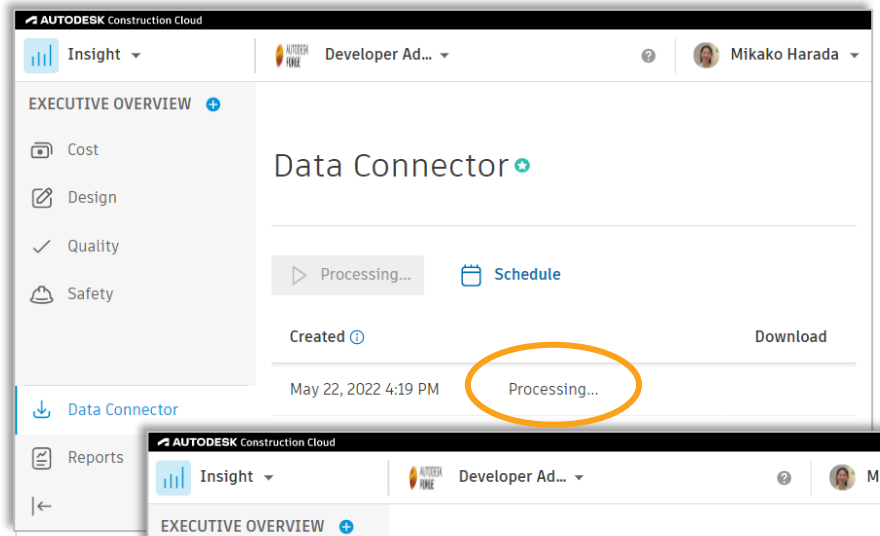
- ブログ記事
  - “BIM 360/ACC Model Properties API”
    https://forge.autodesk.com/blog/bim-360acc-model-properties-api

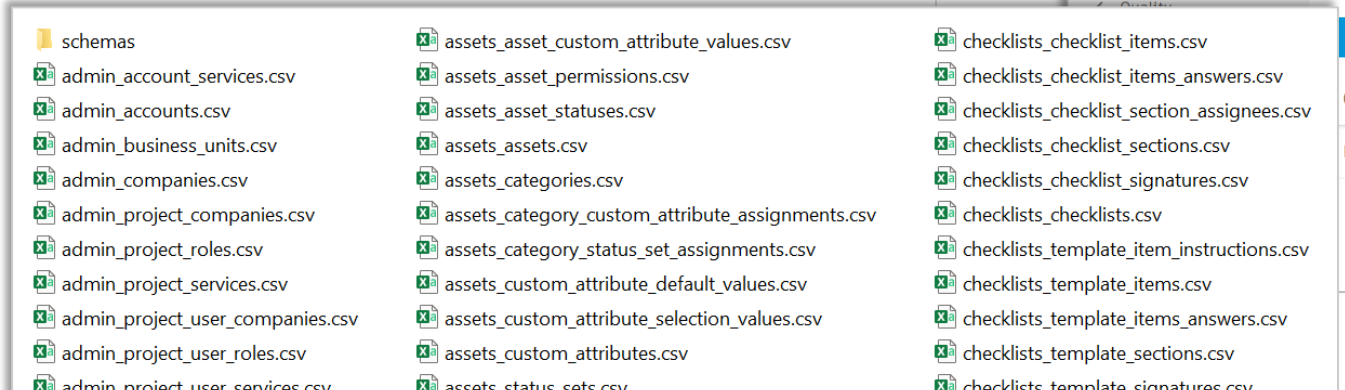
  - キーワード “Model Properties” でページ内検索
    https://forge.autodesk.com/blog/model-properties-api-vs-model-derivative-api

Data Connector API

Data Connector API

- Insight モジュール, アカウントレベル
- データのダンプが可能。
- パブリック (2021年1月~)
- API 自体は B360/ACC 互換
- データは製品によって異なります。
- Tip: **Executive Overview** permission
- プロジェクトレベル(Project Admin)

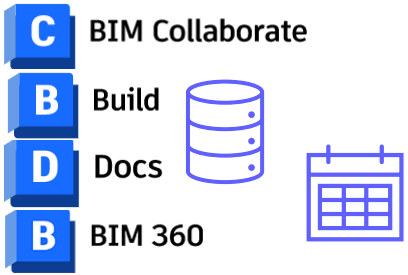


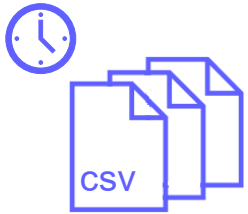


基本的なワークフロー

Data Connector API

Customization





リクエストの登録

データ抽出をリクエストします。スケジュール：ワンタイム、毎日、毎週、毎月、毎年。サービスを選びます（admin、issues など）

ジョブの実行

スケジュールされた時刻にジョブが実行されます。ステータスの確認とジョブIDの取得ができます。データは30日間保存されます。

抽出データのダウンロード

抽出したデータ（CSV）をダウンロードします。全部でも一部でも可。

解析

任意のアプリケーションでデータを利用します。

Data Connector API の仕組み

	エンドポイント	
Requests	POST	**requests**
	GET	requests
	GET	requests/:requestId
	PATCH	requests/:requestId
	DELETE	requests/:requestId
Jobs	GET	requests/:requestId/**jobs**
	GET	jobs/
	GET	jobs/:jobId
	DELETE	jobs/:jobId
Data	GET	jobs/:jobId/**data-listing**
	GET	jobs/:jobId/**data/:name** 

- 11 のエンドポイント

Data Connector API の仕組み

	エンドポイント	
	POST	**requests**
Requests	GET	requests
	GET	requests/:requestId
	PATCH	requests/:requestId
	DELETE	requests/:requestId
Jobs	GET	requests/:requestId/**jobs**
	GET	jobs/
	GET	jobs/:jobId
	DELETE	jobs/:jobId
Data	GET	jobs/:jobId/**data-listing**
	GET	jobs/:jobId/**data/:name**

- データ抽出リクエストの作成
  - 認証されたユーザーのみ
  - 抽出のスケジュール
    - ワンタイム
    - インターバル - 毎日、毎週、毎月、毎年
  - サービスグループ
    - admin, assets, checklists, cost, daily logs, issues, locations, meeting minutes, relationships, rfis, submittals, …
    - さらにサービスが追加されています。
  - (オプショナル) Callback URL - ジョブが実行された際に呼び出されます。

Data Connector API の仕組み

	エンドポイント	
Requests	POST	**requests**
	GET	requests
	GET	requests/:requestId
	PATCH	requests/:requestId
	DELETE	requests/:requestId
Jobs	GET	requests/:requestId/**jobs**
	GET	jobs/
	GET	jobs/:jobId
	DELETE	jobs/:jobId
Data	GET	jobs/:jobId/**data-listing**
	GET	jobs/:jobId/**data/:name**

- データ抽出リクエストの情報を Get/Update/Delete

Data Connector API の仕組み

	エンドポイント	
Requests	POST	**requests**
	GET	requests
	GET	requests/:requestId
	PATCH	requests/:requestId
	DELETE	requests/:requestId
Jobs	GET	requests/:requestId/**jobs**
	GET	jobs/ ★ 7/20 追加！
	GET	jobs/:jobId
	DELETE	jobs/:jobId
Data	GET	jobs/:jobId/**data-listing**
	GET	jobs/:jobId/**data/:name**

- 実行された抽出ジョブの情報の取得
  - status: queued, running, complete
  - completionStatus: success, failed, cancelled
- Delete

Data Connector API の仕組み

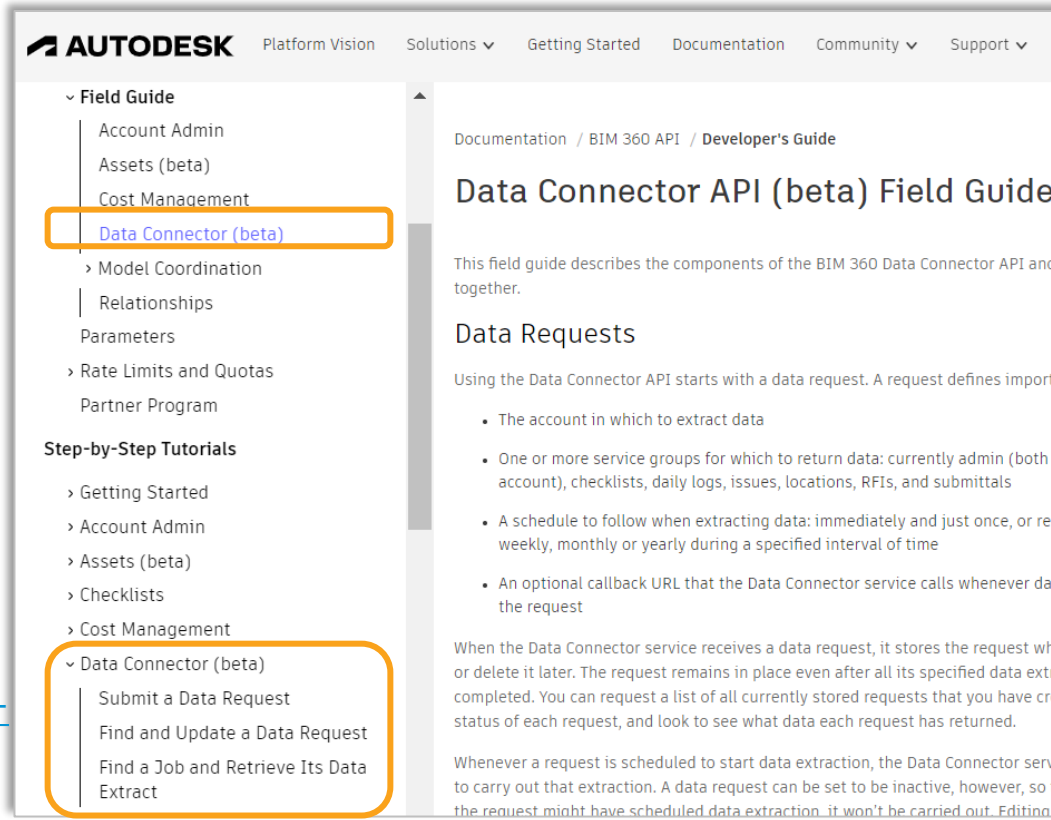
	エンドポイント	
Requests	POST	**requests**
	GET	requests
	GET	requests/:requestId
	PATCH	requests/:requestId
	DELETE	requests/:requestId
Jobs	GET	requests/:requestId/**jobs**
	GET	jobs/
	GET	jobs/:jobId
	DELETE	jobs/:jobId
Data	GET	jobs/:jobId/**data-listing**
	GET	jobs/:jobId/**data/:name**

- 抽出したファイル名のリストを取得
  - README.html
  - autodesk_data_extract.zip
    - (全ファイルを .zip 圧縮、UI と同じ)
  - meta.csv – ジョブの日時とリージョン
  - .csv files
    - 各データに対して1つ以上のファイル
    - e.g., admin_companies.csv
  - Schema フォルダ内にスキーマファイル
- ダウンロード: GET signed URL

開発者リソース

Data Connector

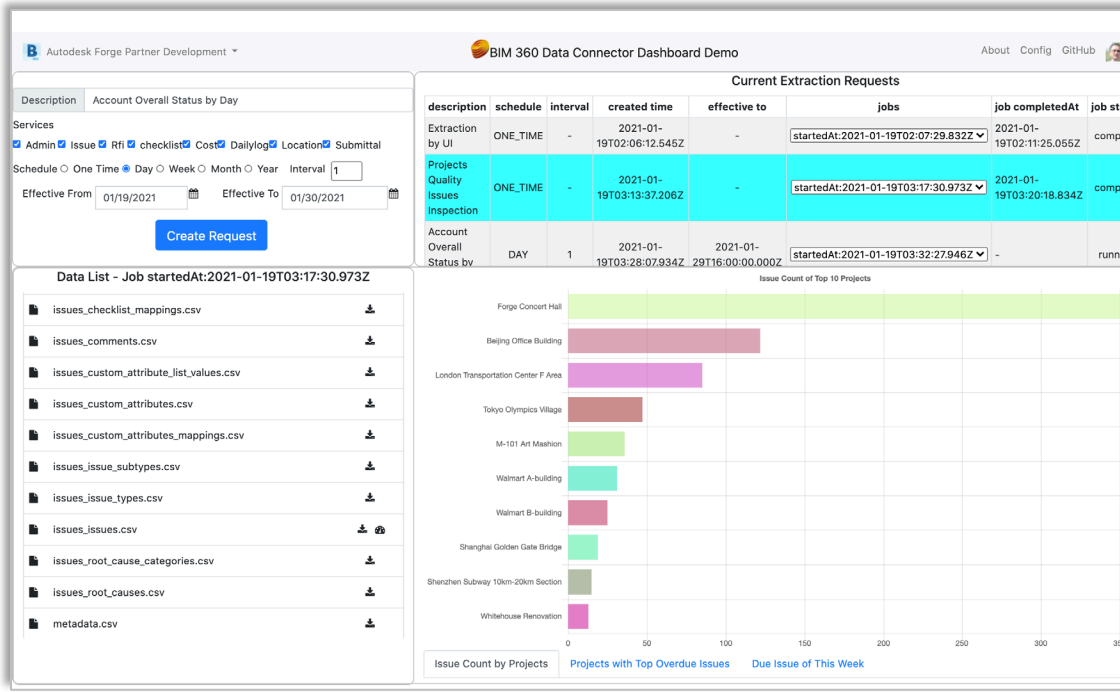
- ドキュメンテーション:
  - Field Guide
  - Step-by-Step Tutorials
  - Reference Guide
- GitHub コードサンプル:
  - Postman collection
  - Export/create requests, get jobs, data and make dashboard
- Blog: https://forge.autodesk.com/blog/bim-360-data-connector-api-available-public-beta

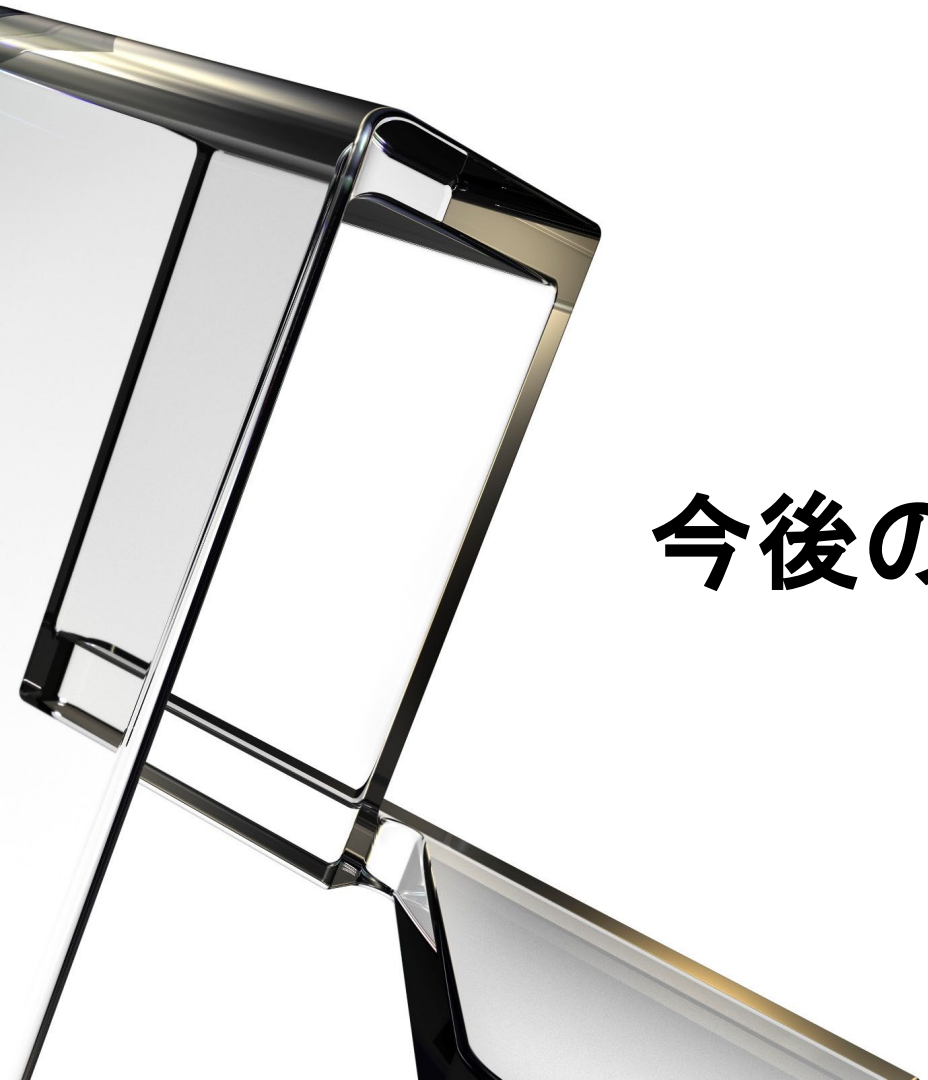


ダッシュボードサンプル

Data Connector

- 概要: スケジューリング（左上）、ジョブのリストアップ（右上）、指定したジョブの利用可能なデータのリストアップ（左下）、棒グラフでの指摘事項のカウント数の表示（右下）。
- ソースコード: GitHub サンプル [Export/create requests, get jobs, data and make dashboard]
- 開発者: Xiaodong Liang, Autodesk





今後の予定

今後の予定

Coming Soon

- Locations write
- Build: シート
- Cost Webhooks
- Forms 取得の改善点
  (割り当てられた担当者の
  会社/役割をサポート)

デザインレビュー

- Docs: ファイル – pdf 書き出し
  (w/ マークアップ)
- Data Connector – プロジェクト
  レベルのアクセス
- AutoSpec (Pype)

高い優先度

- Admin - プロジェクトの作成、
  ユーザー・会社・役割の追加
  - テンプレート
  - 全てのサービスを有効化
- Issues – ベータ版からパブリッ
  クリリース

Thank you!

@autodeskforge