



SD471532

# ヒントやコツ、 Forge Model Derivative Service の将来

伊勢崎 俊明

Developer Advocate : @AutodeskForge

# アジェンダ

- Model Derivative API の役割
- SVF 変換時の変換オプション
- SVF 変換で得られるメタデータの利用
- Model Derivative API の将来



伊勢崎 俊明

Chuo-ku, Tokyo, JP

---

Developer Advocacy & Support  
Autodesk Japan  
Software Development

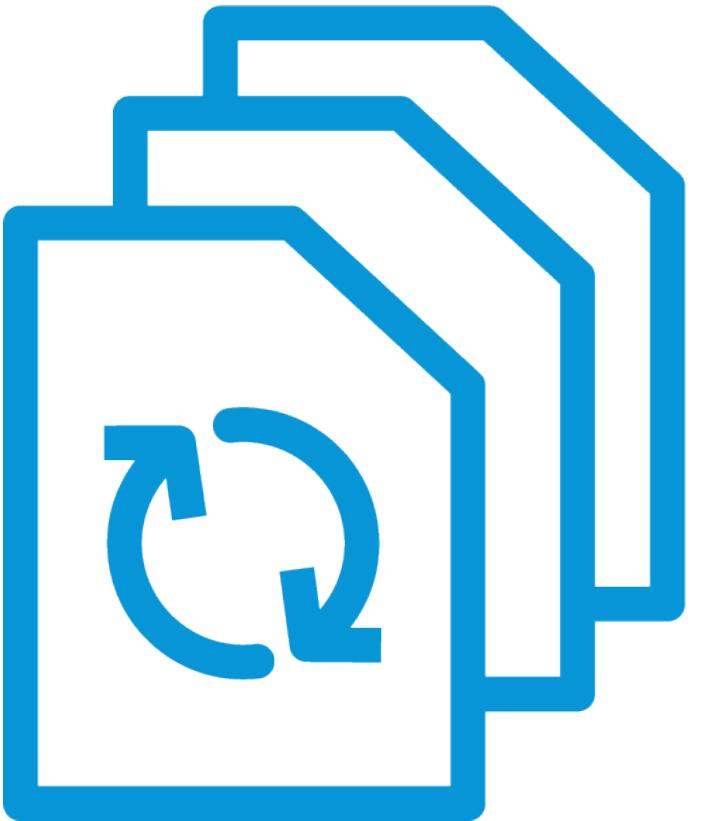
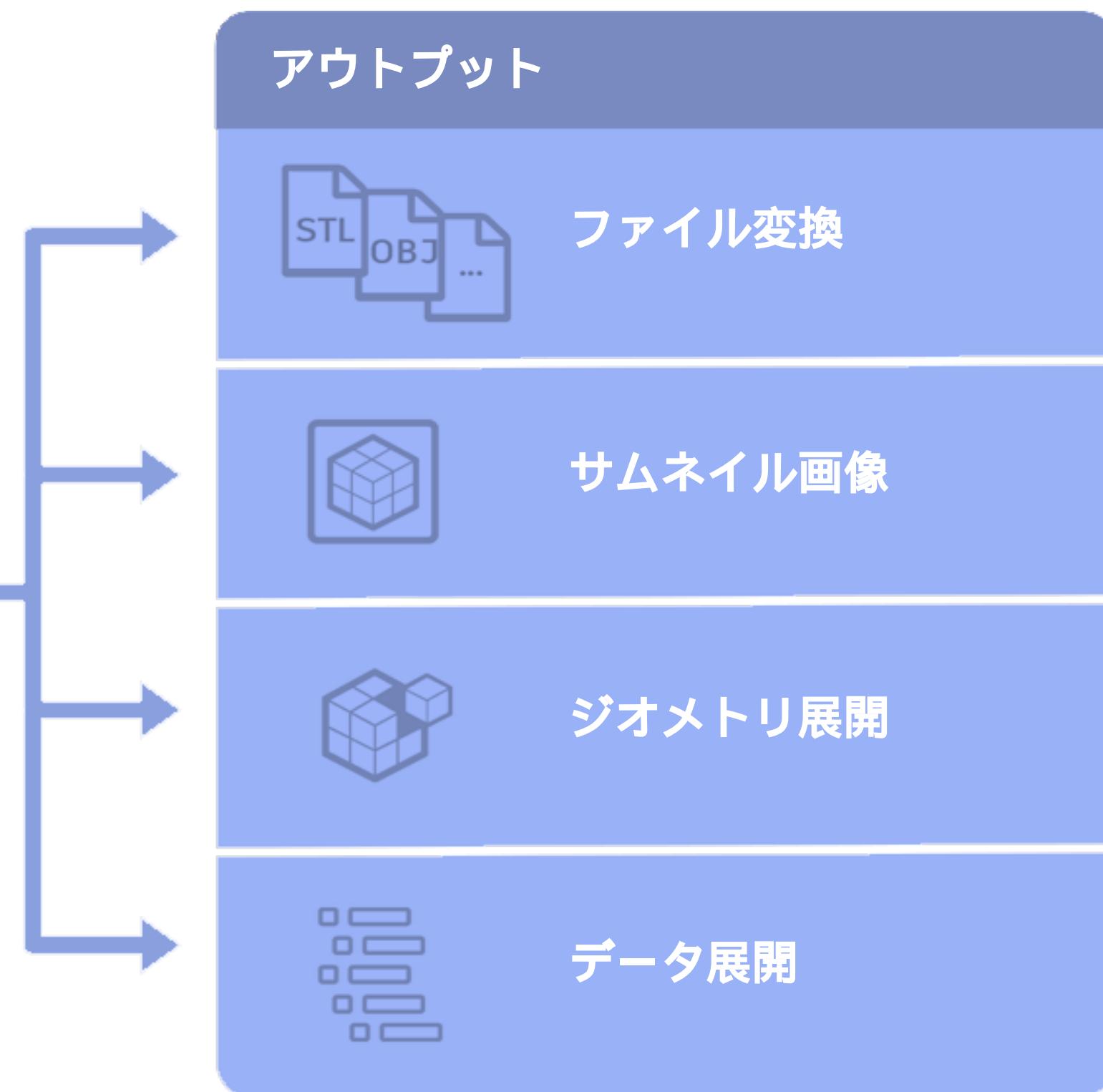
CAD / CAM 業界の様々な企業での製品教育、製品/開発サポート、アプリケーション開発を経て、現在、オートデスクに 20 年超在籍。セールス エンジニアを経て、過去から最新までの技術変遷を踏まえて、Forge を啓蒙するエバンジェリストとして活動中。

# Model Derivative API の役割

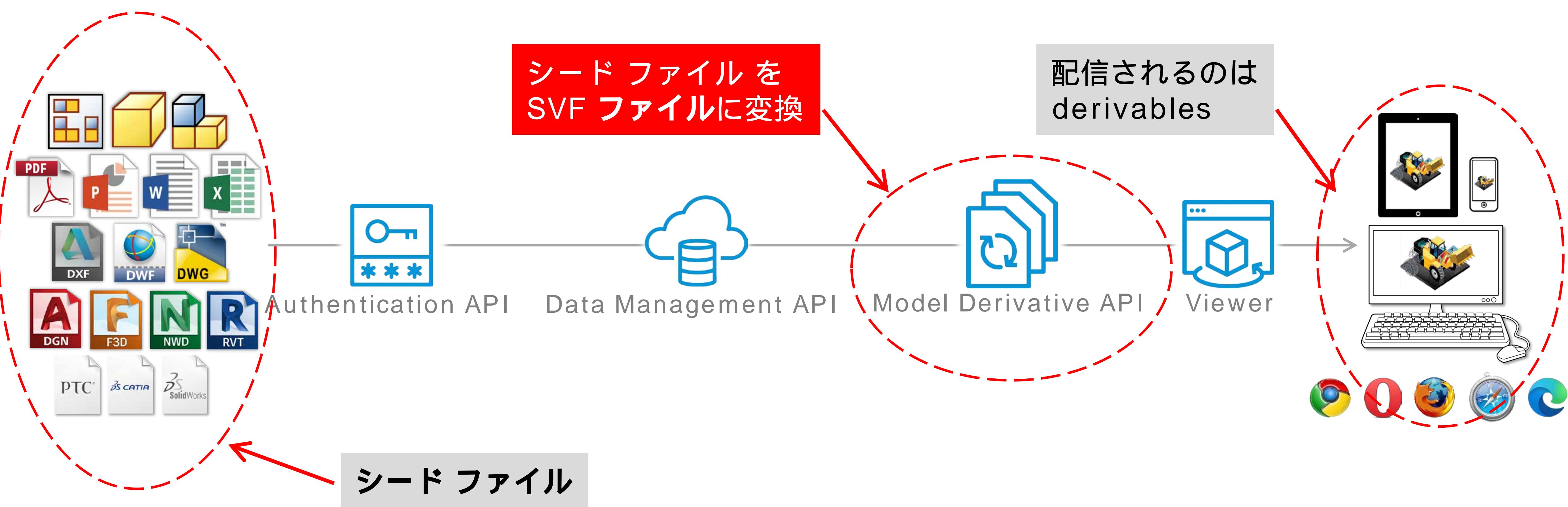
- ・ デザイン ファイルを変換
  - § 他のデザイン ファイル形式に変換
  - § Viewer 用に SVF 形式に変換してブラウザで表示
  - § ジオメトリ データやモデル階層の展開
  - § サムネイル画像の生成
- § RESTful API



ソース ファイル



# Forge Viewer ソリューションの流れ



# SVF 変換時の変換オプション

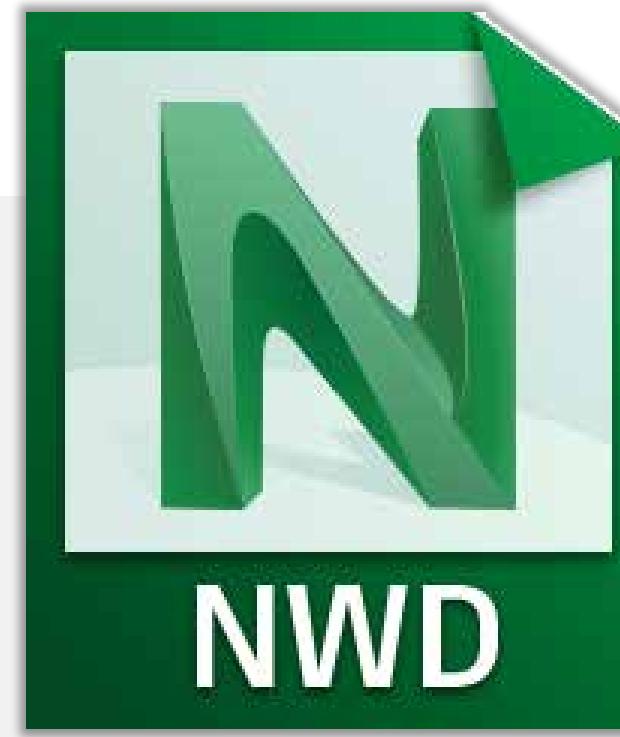
The screenshot shows the Model Derivative API v2 developer documentation. On the left, there's a sidebar with links like 'Developer's Guide', 'Step-by-Step Tutorials', 'Code Samples', 'API Reference' (which is expanded), and 'HTTP Specification'. The main content area is titled 'Attributes by Output Type' and 'Case 3: Input file type is Navisworks'. It details various options for transforming input files into SVF derivatives. A red arrow points to the 'timelinerProperties' section.

Attribute	Type	Description
advanced	object	A set of special options, which you must specify only if the input file is IFC, Revit, or Navisworks.
materialMode	string	Specifies the materials to apply to the generated SVF derivatives. Available options are: <ul style="list-style-type: none"><li>- <b>auto</b> - (Default) Use the current setting of the default view of the file.</li><li>- <b>basic</b> - Use basic materials.</li><li>- <b>autodesk</b> - Use Autodesk materials.</li></ul>
hiddenObjects	bool	<b>true</b> : Extract hidden objects from the input file. <b>false</b> : (Default) Do not extract hidden objects from the input file.
basicMaterialProperties	bool	<b>true</b> : Extract properties for basic materials. <b>false</b> : (Default) Do not extract properties for basic materials.
autodeskMaterialProperties	bool	<b>true</b> : Extract properties for Autodesk materials. <b>false</b> : (Default) Do not extract properties for Autodesk materials.
timelinerProperties	bool	<b>true</b> : Extract timeliner properties. <b>false</b> : (Default) Do not extract timeliner properties.

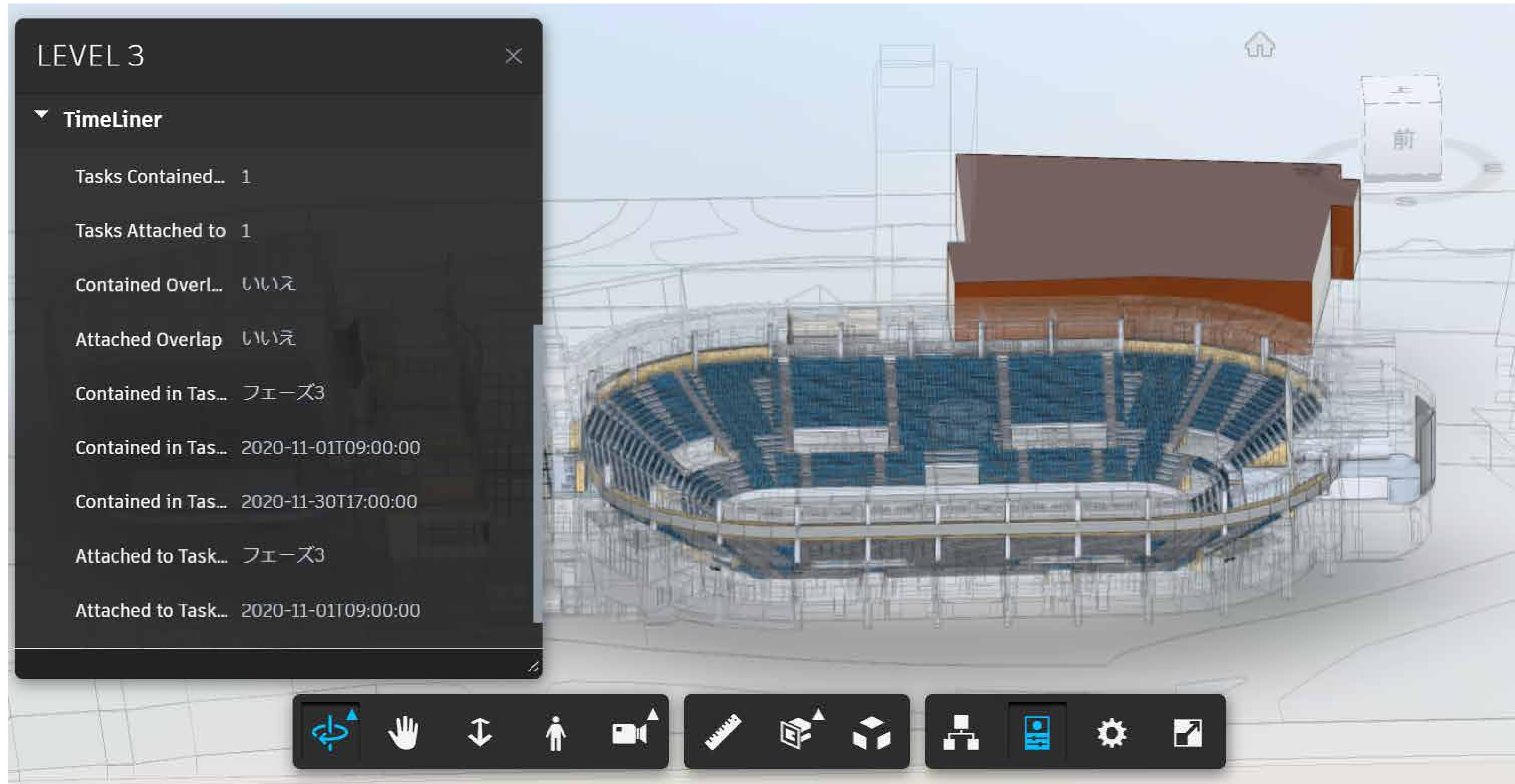
# リクエストパラメータ timelinerProperties

- POST job のリクエスト ボディ

```
{  
  "input": {  
    "urn": "<urn>  
  },  
  "output": {  
    "formats": [  
      {  
        "type": "svf",  
        "views": [  
          "2d",  
          "3d"  
        ],  
        "advanced": {  
          "timelinerProperties": true  
        }  
      }  
    ]  
  }  
}
```



# Timeliner (NWD ファイルのみ)



# SVF 変換時の変換オプション

Model Derivative API v2

- > Developer's Guide
- > Step-by-Step Tutorials
- > Code Samples
- ▽ API Reference
  - ▽ HTTP Specification
    - ▽ Derivatives
      - GET formats
      - POST job
      - POST references
      - GET :urn/thumbnail
      - GET :urn/manifest
      - DELETE :urn/manifest
      - GET :urn/manifest/:derivativeURN
      - HEAD :urn/manifest/:derivativeURN
      - GET :urn/metadata
      - GET :urn/metadata/:guid

## Attributes by Output Type

### SVF Output

#### Case 2: Input file type is Revit

[Expand all](#)

formats \* A JSON object representing the requested output types.

object

views \* Required options for SVF type. Possible values: 2d , 3d

array: string

advanced A set of special options, which you must specify only if the input is Revit, or Navisworks.

object

generateMasterViews Generates master views when translating from the Revit input format. This option is ignored for all other input formats. This attribute default

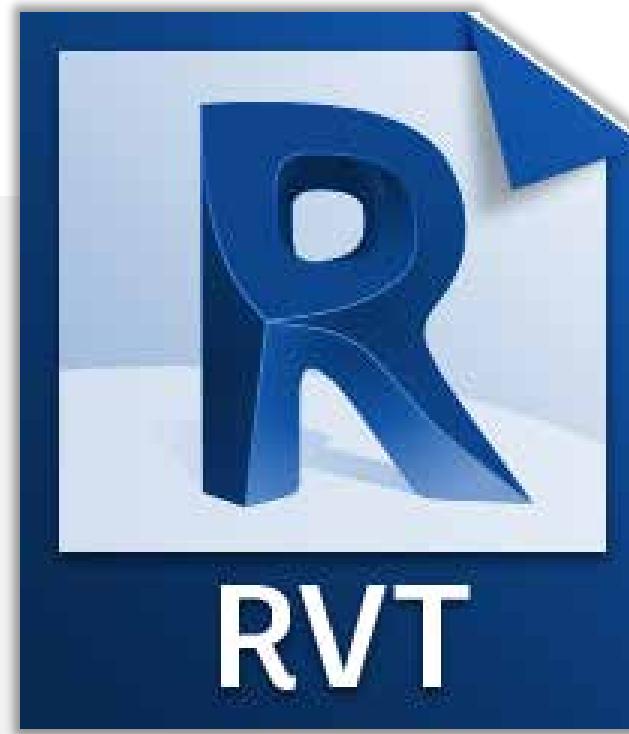
bool

Master views are 3D views that are generated for each phase of the model. A master view contains all elements (including "room" elements) present in the model for that phase. The display name of a master view defaults to the phase it is generated from. However, if a view with that name already exists, the Model Derivative service appends a suffix to the default display name.

# リクエスト パラメータ generateMasterViews

- POST job のリクエスト ボディ

```
{  
  "input": {  
    "urn": "<urn>  
  },  
  "output": {  
    "formats": [  
      {  
        "type": "svf",  
        "views": [  
          "2d",  
          "3d"  
        ],  
        "advanced": {  
          "generateMasterViews": true  
        }  
      }  
    ]  
  }  
}
```



# 部屋とスペース (RVT ファイルのみ)

モデル

検索

植栽

造作工事

特殊設備

部屋

洋室(1) 101 [593983]

洋室(2) 101 [593990]

LDK 101 [593993]

LDK 102 [593996]

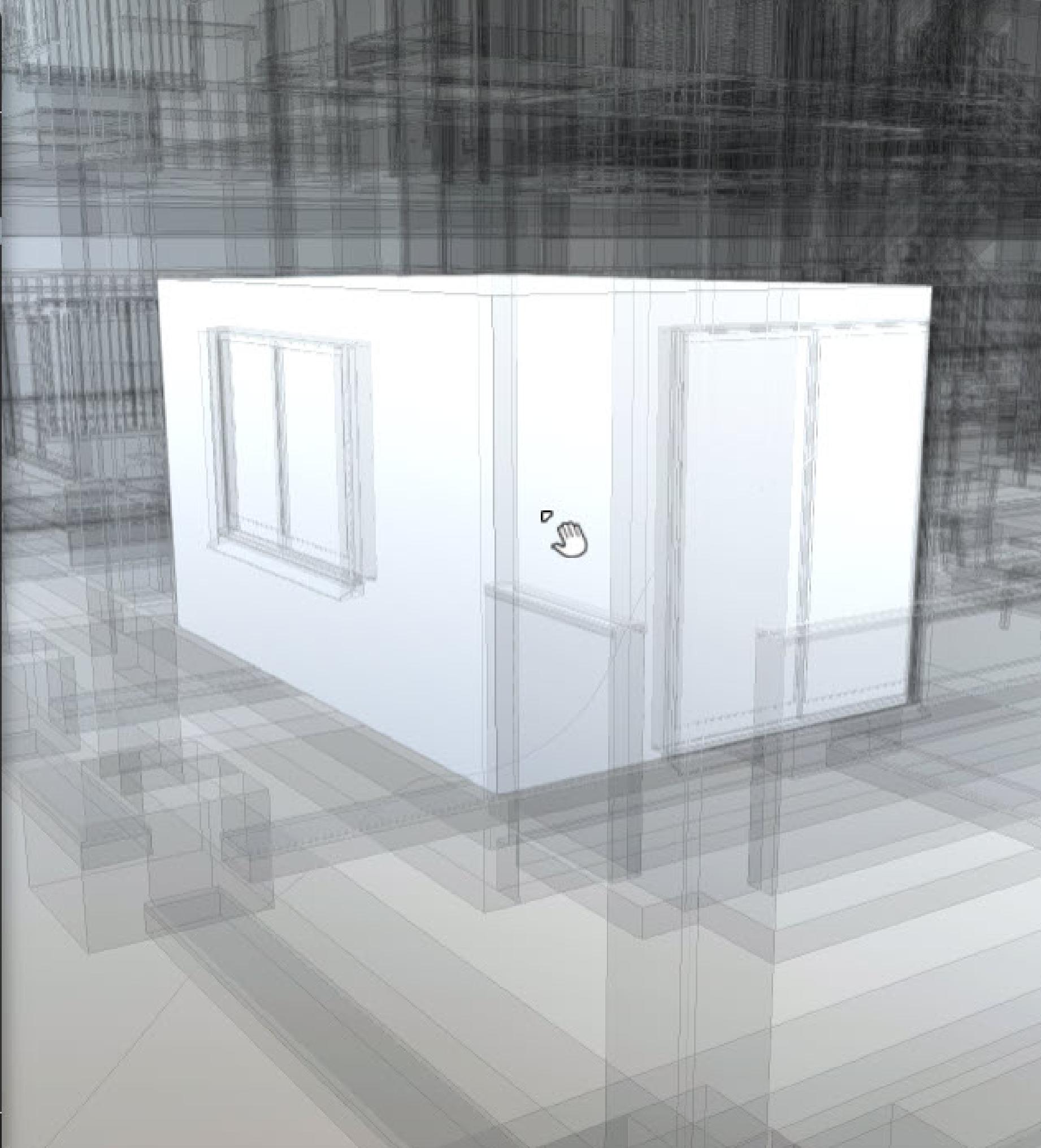
和室 102 [593999]

玄関 101 [594054]

ホール 101 [594057]

トイレ 101 [594060]

PS 101 [594063]



A hand icon is positioned over the door handle of a room in the 3D model.

洋室(1) 101 [593983]

Name 洋室(1) 101 [593983]

拘束

レベル 1FL

上部レベル 1FL

オフセット(上部...) 3200 mm

基準レベルオフ... 0 mm

寸法

面積 13.46999999999995 m<sup>2</sup>

周長 15400.000000000004 mm

部屋高さ(レベル...) 3200 mm

容積 29.84030460000006 m<sup>3</sup>

坪/帖換算

算定高さ 0 mm



# SVF 変換で得られるメタデータの利用

 Model Derivative API v2

Developer's Guide

- Overview

API Basics

- About this API
- Translate Models
- Extract Metadata** 
- Extract Geometry
- Extract Thumbnails

GDPR Compliance

Webhooks

Field Guide

Supported Translations

> Rate Limits

> Step-by-Step Tutorials

> Code Samples

## Extract Metadata

When you translate a model into the SVF format, the Model Derivative service saves information about the derivatives (metadata) in the manifest. When the source model is large, the manifest can become difficult to parse. As such, the Model Derivative API provides endpoints to specifically query metadata.

[GET :urn/metadata](#) lets you extract information about the 3D Views and 2D sheets/views referenced in the manifest. These derivatives are the Viewables that you can typically display in a browser using the Forge Viewer SDK. See the tutorial on [Extract Metadata from a Source Model](#) to see how you can extract the names of Viewables and their metadata GUIDs (Global Unique Identifier).

Source models from applications such as Autodesk Inventor and Fusion 360 produce only one Viewable per model. However, source models from applications such as Autodesk Revit can contain multiple Viewables.

Once you obtain the GUID of a Viewable, you can use [GET :urn/metadata/:guid](#) to obtain the object/component hierarchy of the model. In addition to the hierarchy, the list provides the `objectid` of each object. See the tutorial on [Extract Geometry from a Source File](#) for a demonstration on how the object hierarchy and objectsids are used to uniquely identify geometry and thereafter extracted as OBJ files.

Using [GET :urn/metadata/:guid/properties](#) you can obtain a flat list of objects in that Viewable. It also returns the properties of each object. Using a query parameter, you can filter the results to provide the properties of one specific object.

The following image shows the object hierarchy and the properties of a selected object, as displayed in a browser using the Forge Viewer SDK.

# 派生データの取得 - その 1

- ビュー上のオブジェクト階層メタデータの GUID を取得

GET

:urn/metadata

Returns a list of model view (metadata) IDs for a design model. The metadata ID enables end users to select an object tree and properties for a specific model view.

Although most design apps (e.g., Fusion and Inventor) only allow a single model view (object tree and set of properties), some apps (e.g., Revit) allow users to design models with multiple model views (e.g., HVAC, architecture, perspective).

Note that you can only retrieve metadata from an input file that has been translated into an SVF file.

# 派生データの取得 - その 2

- GUID からツリー構造やオブジェクト毎の情報を取得

GET

`:urn/metadata/:guid`

Returns an object tree, i.e., a hierarchical list of objects for a model view.

To call this endpoint you first need to call the [GET :urn/metadata](#) endpoint, to determine which model view (object tree and set of properties) to use.

GET

`:urn/metadata/:guid/properties`

Returns a list of properties for each object in an object tree. Properties are returned according to object ID and do not follow a hierarchical structure.

# Revit 派生データでのバージョン比較例

\* VS Code Forge Tools エクステンションを利用

• Untitled-1 - Visual Studio Code

• Untitled-2

```
{ "objectid": 8692, "name": "鋼製_一般枠_引違い-二枚 [563046]", "externalId": "d9b3f277-b695-4a6f-a1a5-8d30a...", "properties": { "IFC パラメータ": { "開勝手": "" }, "その他": { "上枠の高さ": "2100.000 mm", "垂直距離": "0.000 mm", "天井高さ": "0.000 mm", "天端高さ": "0.000 mm", "抱き見込": "60.000 mm", "排煙有効高さ": "0.000 mm", "既定値の敷居の高さ": "0.000 mm", "有効幅": "1800.000 mm", } } }
```

```
{ "objectid": 8683, "name": "鋼製_一般枠_引違い-二枚 [563046]", "externalId": "d9b3f277-b695-4a6f-a1a5-8d30a...", "properties": { "IFC パラメータ": { "開勝手": "" }, "その他": { "上枠の高さ": "3000.000 autodesk.unit", "垂直距離": "0.000 autodesk.unit.unit", "天井高さ": "0.000 autodesk.unit.unit", "天端高さ": "0.000 autodesk.unit.unit", "抱き見込": "60.000 autodesk.unit.uni", "排煙有効高さ": "0.000 autodesk.unit.", "既定値の敷居の高さ": "100.000 autode", "有効幅": "1800.000 autodesk.unit.uni" } } }
```

Snoop Objects

FamilyInstance  
└ < 1800x1200 563046 >

Field	Value
UniqueId	d9b3f277-b695-4a6f-a1a5-8d30ae36ba4e-00089766
VersionGuid	940b4090-a921-4f99-a97e-14b064df1b30
ViewSpecific	False
WorksetId	< WorksetId >

# Model Derivative API の将来

- パフォーマンス向上の計画：

- § 大規模モデルへの対応拡大

- § モバイル デバイス等、低メモリ環境での利便性改善

- ⇒ 新しい SVF2 形式の導入

# Private Beta : SVF2 形式

- Streaming Vector Format の正常進化バージョン
- 当初 OTG ( Oscar The Grouch's ) で紹介
- BIM 360 Docs 上で超大規模モデル表示用途で利用
- ジオメトリの共有利用、キャッシュ再利用による効果
- 少メモリ、Socket 通信による高速ロード、etc
- Model Derivative API で SVF2 を指定して変換が必要
- Forge Viewer 7.25 以上のバージョンで表示可能
- Private Beta として一部デベロッパの方による評価中 (2020年9月現在)

SVF : isSVF2()=undefined

Total geometry size: 108.47375679016113 MB

Number of meshes: 27796

Num Meshes on GPU: 10000

Net GPU geom memory used: 53333904



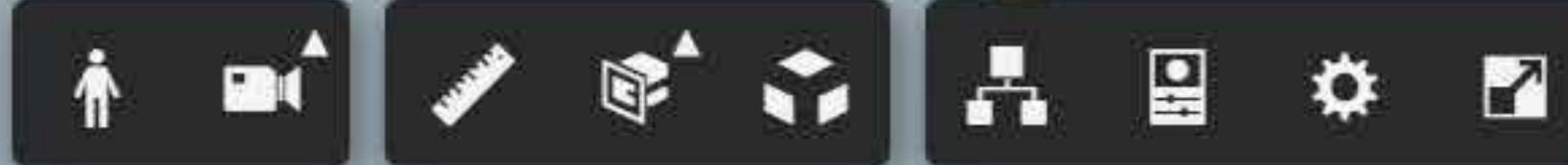
SVF2 : isSVF2()=true

Total geometry size: 16.585140228271484 MB

Number of meshes: 3040

Num Meshes on GPU: 3040

Net GPU geom memory used: 16369340



# 関連セッションのご案内

SD473709



Kevin Vandecar

Manchester, New Hampshire, US

Principal Developer Advocate  
Autodesk  
Software Development

## More Tips, Tricks, and the Future of the Forge Model Derivative Service

Kevin Vandecar

Last year we brought you 'Tips, Tricks, and the Future for Forge Model Derivative Services.' We're bringing even more goodies to this year's presentation. We will include more details around the Revit flag to generate rooms and spaces for viewing. We will cover how you can use the model derivative data to do model version comparison. Finally, again we will preview some of the coming features in Model Derivative Service, including the improved SVF2 Forge Viewer format.

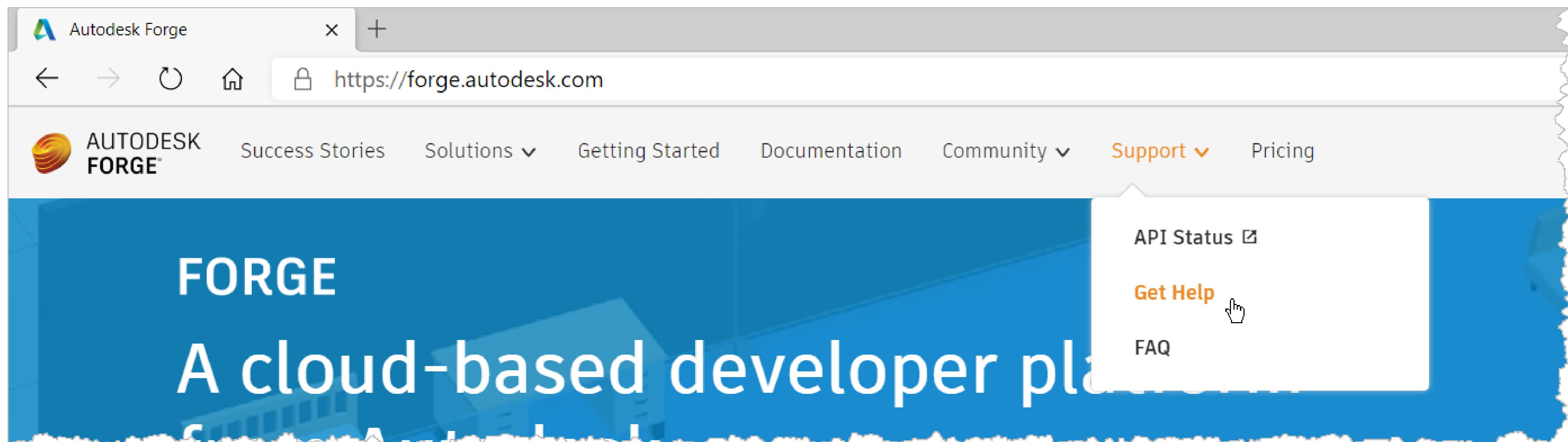
### - Hide key learnings

- Learn details about the Revit generateMasterViews flags to get room and space information.
- Learn how to compare the data sets coming from the Model Derivative service.
- See a preview of coming new features in Model Derivative service.
- Learn about how and when to use the new SVF2 format.

ご不明な点は

Email : [forge.help@autodesk.com](mailto:forge.help@autodesk.com)

または





Autodesk およびオートデスクのロゴは、米国およびその他の国々における Autodesk, Inc. およびその子会社または関連会社の登録商標または商標です。その他のすべてのブランド名、製品名、または商標は、それぞれの所有者に帰属します。オートデスクは、通知を行うことなくいつでも該当製品およびサービスの提供、機能および価格を変更する権利を留保し、本書中の誤植または図表の誤りについて責任を負いません。

© 2020 Autodesk. All rights reserved.

