AUTODESK UNIVERSITY

Unleash the Power of Design Data with Autodesk Forge API

Mikako Harada
Senior Manager AEC, Developer Technical Services, Autodesk
@mikako_harada

Don Whittle
Distinguished Architect, Construction Solutions, Autodesk

Safe Harbor Statement

The presentations during this event may contain forward-looking statements about our outlook, future results and related assumptions, total addressable markets, acquisitions, products and product capabilities, and strategies. These statements reflect our best judgment based on currently known factors. Actual events or results could differ materially. Please refer to our SEC filings, including our most recent Form 10-K and Form 10-Q filings available at www.sec.gov, for important risks and other factors that may cause our actual results to differ from those in our forward-looking statements.

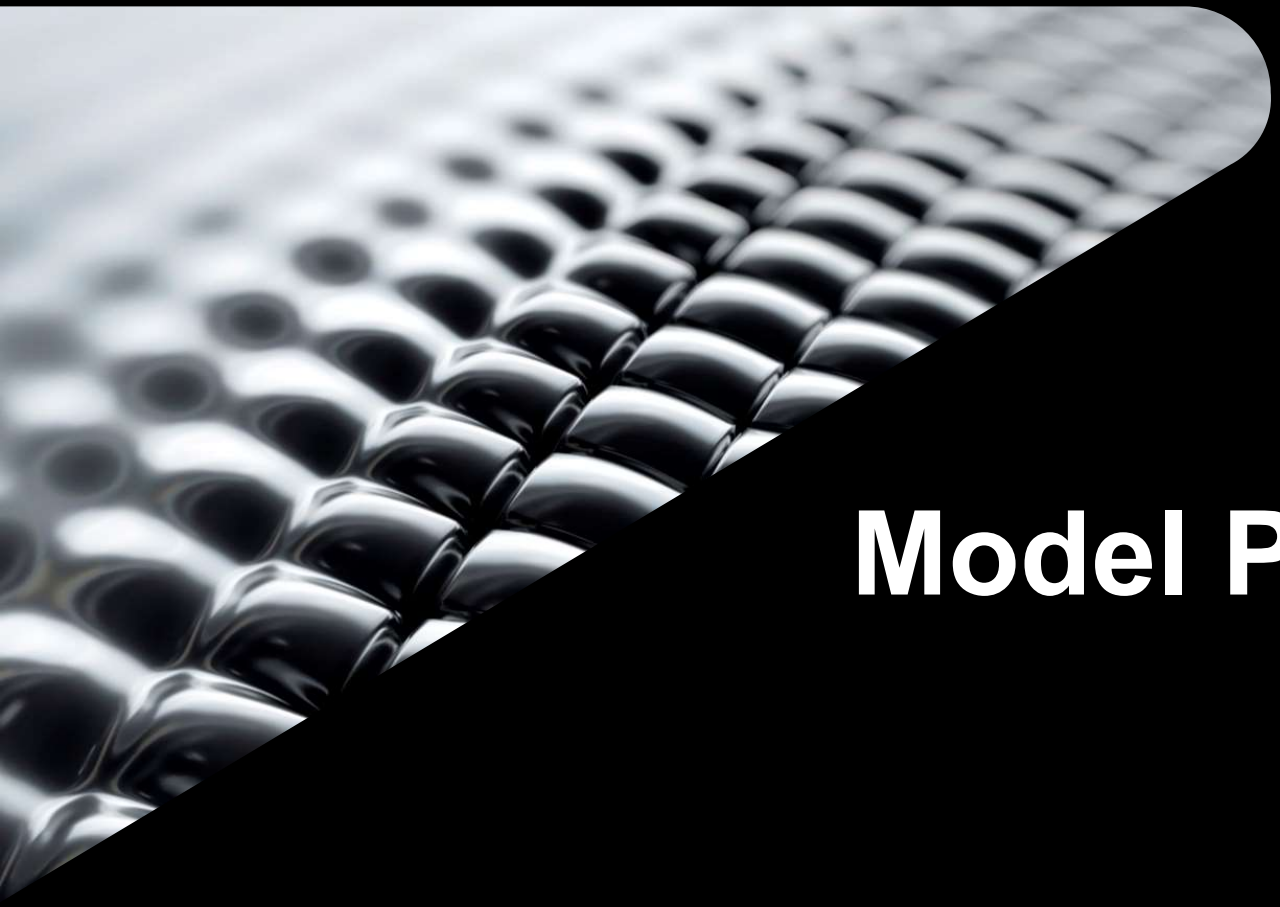
The forward-looking statements made in these presentations are being made as of the time and date of their live presentation. If these presentations are reviewed after the time and date of their live presentation, even if subsequently made available by us, on our website or otherwise, these presentations may not contain current or accurate information. We disclaim any obligation to update or revise any forward-looking statements.

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.

PLEASE NOTE: All Autodesk content is proprietary. Please Do Not Copy, Post or Distribute without authorization.

Agenda

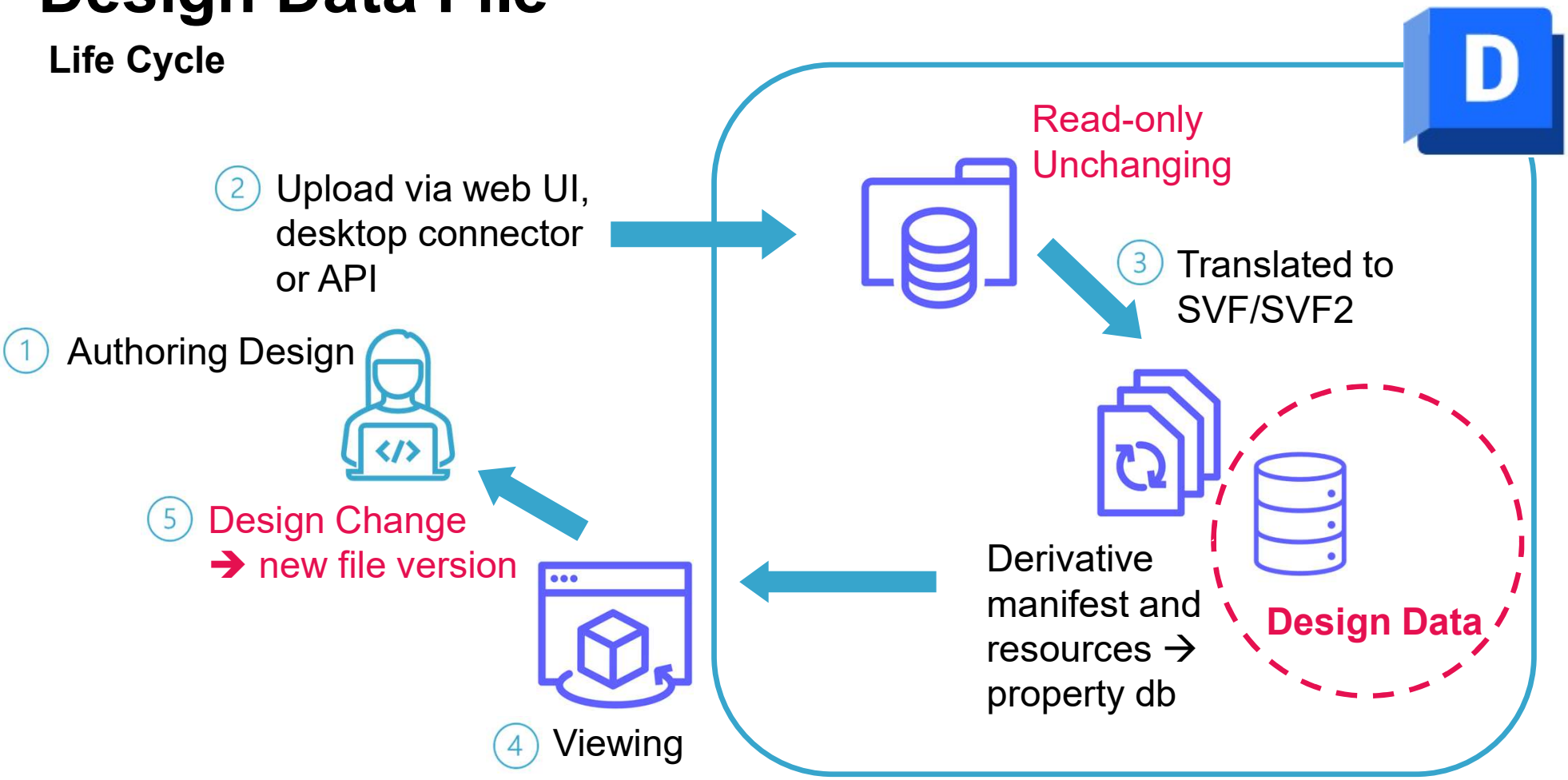
- Model Property API
- Data Exchange API (beta)
- Parameters Service API (in pipeline)

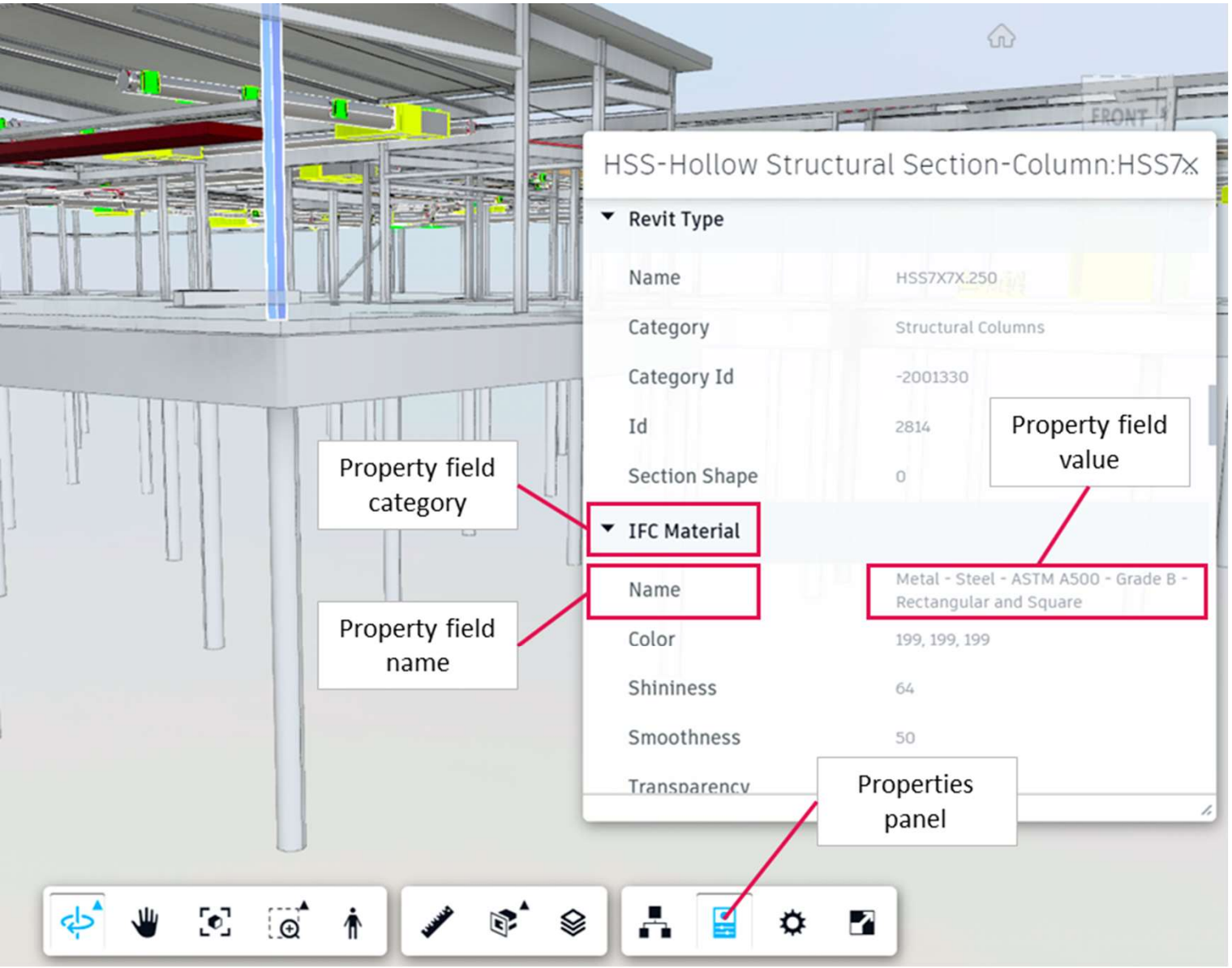


Model Properties API

Design Data File

Life Cycle



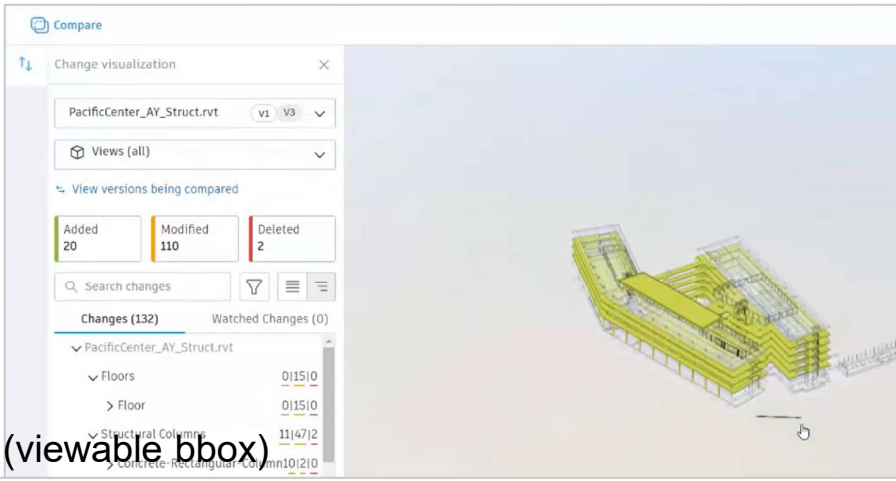


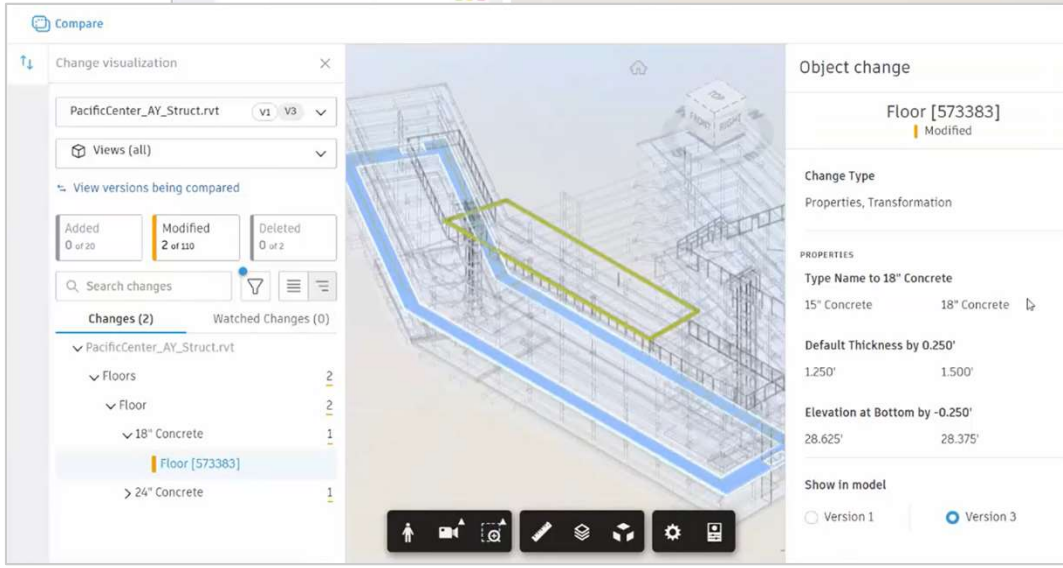
Classifying (BIM) SVF2 model properties

- Category
- Name
- Type (e.g., string, double)
- (Optional) Unit of Measure

Model Properties API

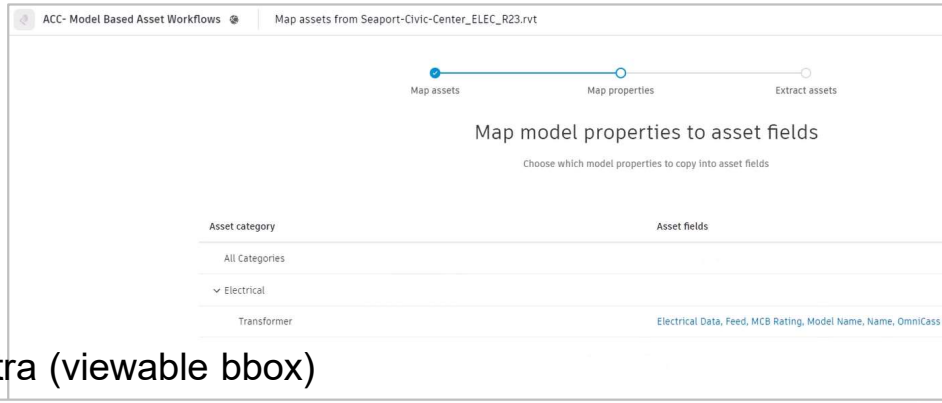
- Released Feb 2022
- Autodesk/BIM 360 Docs based products. US and EMEA
- Built on PropertyDb from Derivative Services (svf translation)
  - **Index** (Base) – query, filter properties of svf2 objects + extra (viewable bbox)
  - **Diff** – Index + compare two versions
- Used in product
  - Design Collaboration - Change analysis
  - Assets – integrating with models *new!*
- Feature release in pipeline
  - Model Coordination – model property breakdown

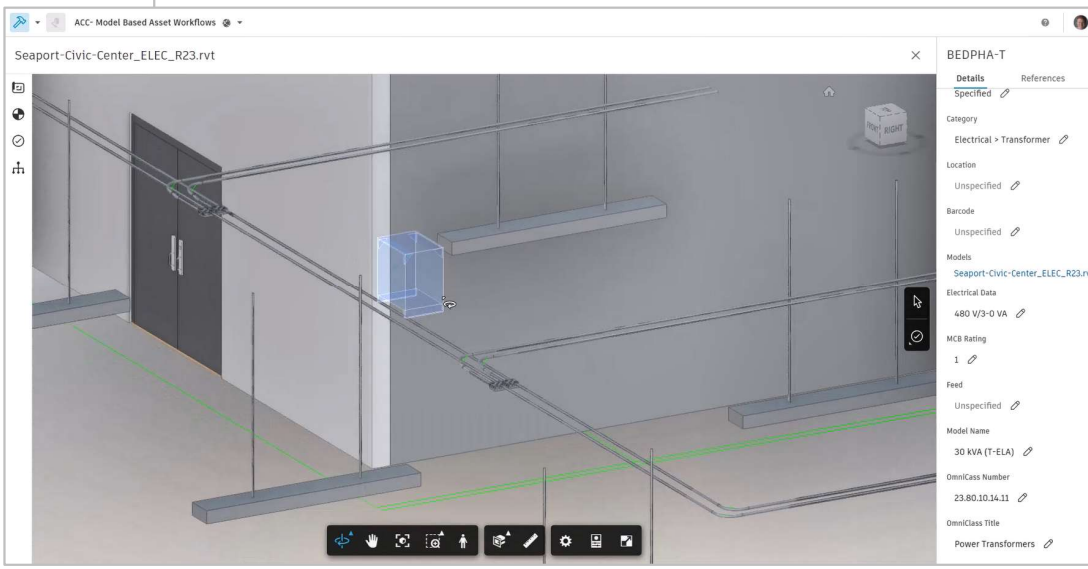




Model Properties API

- Released Feb 2022
- Autodesk/BIM 360 Docs based products. US and EMEA
- Built on PropertyDb from Derivative Services (svf translation)
  - **Index** (Base) – query, filter properties of svf2 objects + extra (viewable bbox)
  - **Diff** – Index + compare two versions
- Used in product
  - Design Collaboration - Change analysis
  - Assets – integrating with models *new!*
- Feature release in pipeline
  - Model Coordination – model property breakdown

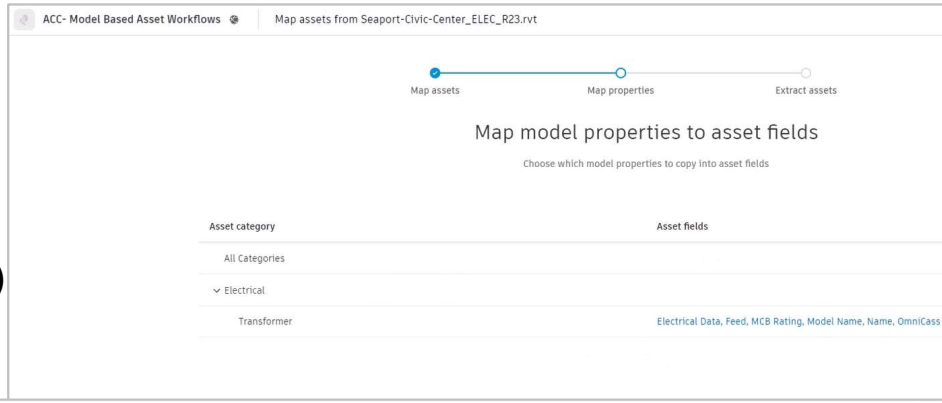


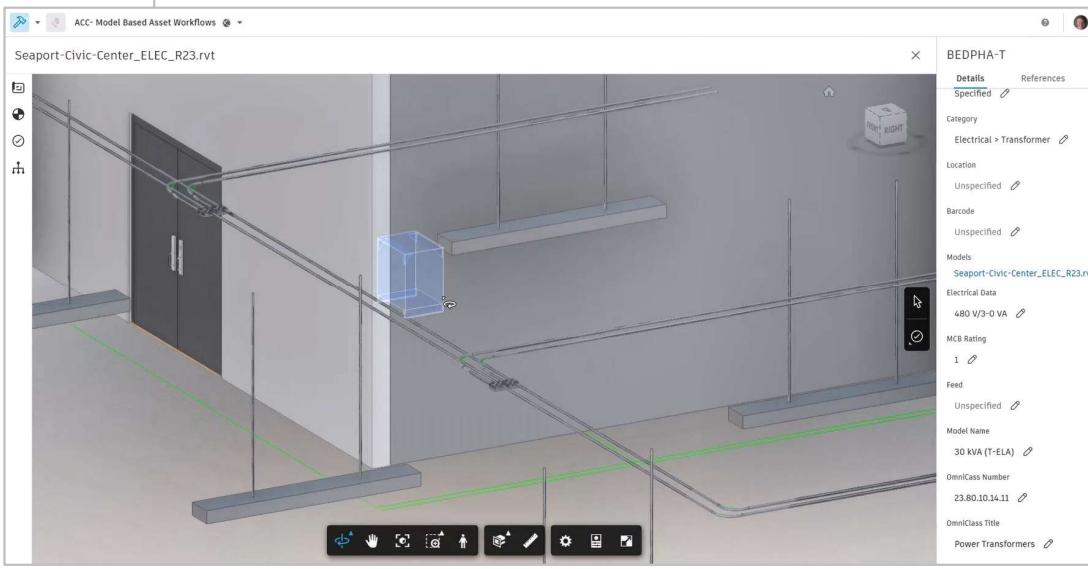


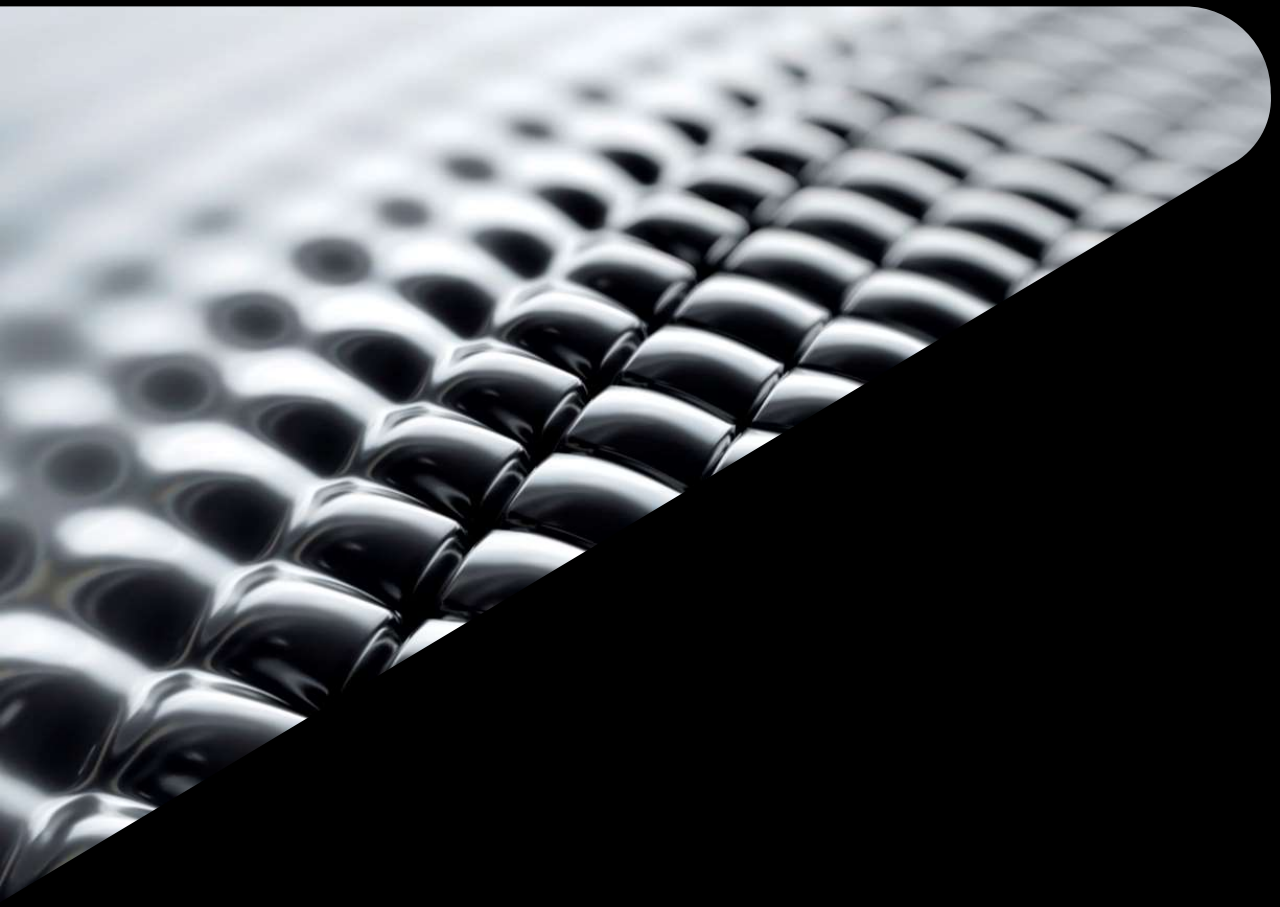
Model Properties API

Supported Files

- **Index** – any files supported Derivative Service /SVF2
- **Diff** - element id needs to be stable (uniquely identifiable)
  - **RVT**
  - **DWG**
  - **NWC** exported from:
    - Revit and
    - AutoCAD verticals
  - **IFC** exported from:
    - AutoCAD Architecture, MEP, Civil 3D 2018+
    - ARCHCAD
    - Revit
    - MigiCAD for Revit
    - Tekla Structures







Sample Applications

Model Properties API

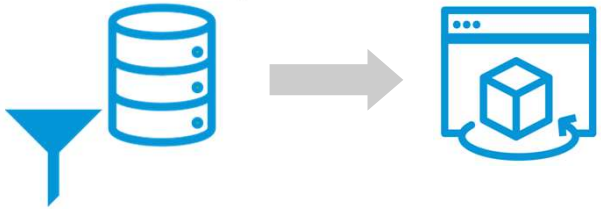
Filter Elements & Partial Model Load

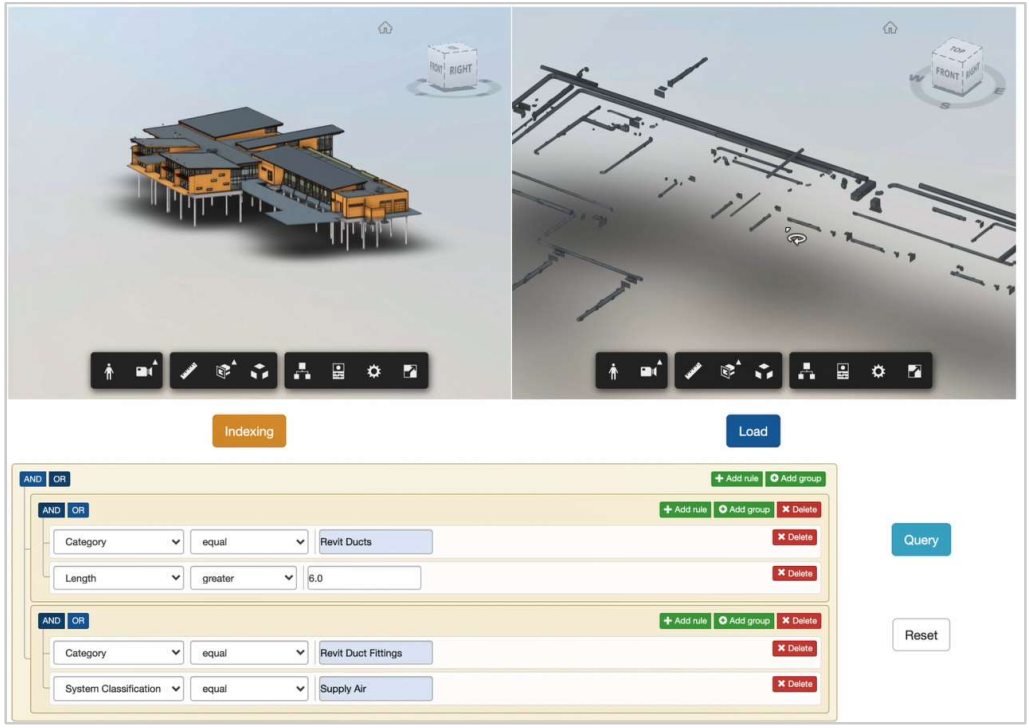
Index

What it does: Filters elements by properties, which can be geometric properties such as length and height. A filter condition is defined as a binary expression form and can be combined by AND/OR. The results are visualized in the Forge viewer.

Code: GitHub sample

Author: Xiaodong Liang, Autodesk



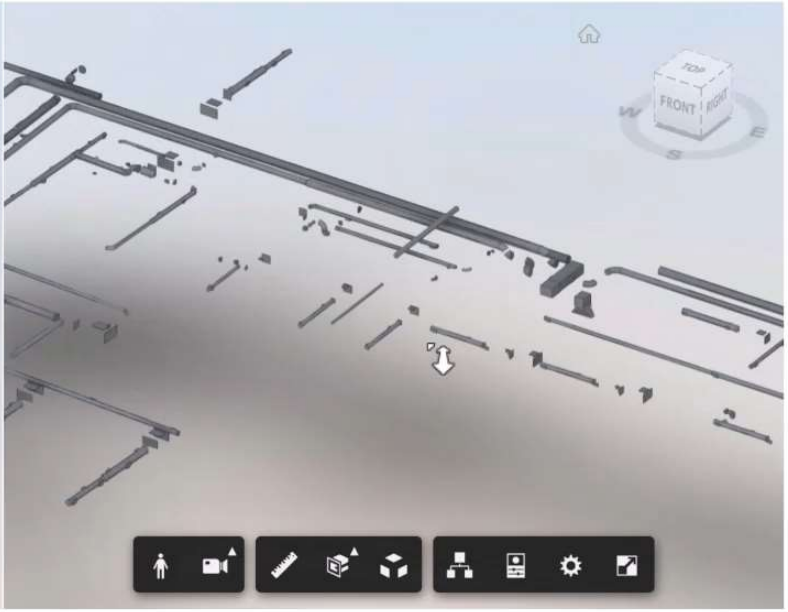


AUTODESK FORGE

Elements Filtering and Partial Loading

Xiaodong Liang

- Forge Sandbox MH Indexing
- Pacific Center Hospital v2
- Xiaodong Tes & Patch
- Xiaodong-test-MC
  - Plans
  - Project Files
- xiaodong-new -project
  - Plans
  - Project Files
    - C4R
    - Indexing
      - Audubon-Merged.nwd
      - Audubon_Architecture-2022-Root.rvt
        - v1: 8/17/2021, 2:40:11 PM by Xiaodong Liang
      - Audubon_Mechanical-2022.rvt
      - Audubon_Structure-2022.rvt
      - BIM 360 Sample Assembly.rvt
    - Name Standards
    - Project Files.1
    - Project Files.4
    - non-C4R
    - Audubon_Architecture.rvt
    - Audubon_Mechanical.rvt
    - Audubon_Structure.rvt
    - Audubon_Structure_assembly.rvt
    - STEEL_TEST.nwd
    - rac_advanced_sample_project - Floor Plan - 01 - Entry
    - rac_advanced_sample_project.rvt
      - v1: 8/8/2021, 11:14:22 PM by Xiaodong Liang
    - rac_basic_sample_project.rvt
    - rme_basic_sample_project.nwd
- xiaodong-old-project
- xiaodong-test-asset
- Developer Advocacy Support
- AU ACC TakeOff Sample Project
  - For the Field
  - Project Files
    - Shared
    - Zhong
      - Consumed



Indexing

Load

AND OR | + Add rule | Add group

AND OR | + Add rule | Add group | ✖ Delete

Category	equal	Revit Ducts	✖ Delete
Length	greater	6.0	✖ Delete

AND OR | + Add rule | Add group | ✖ Delete

Category	equal	Revit Duct Fittings	✖ Delete
System Classification	equal	Supply Air	✖ Delete

Query

Reset

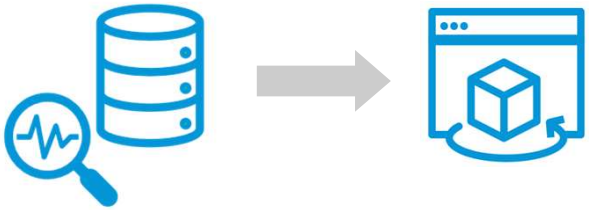
Compare Two Versions

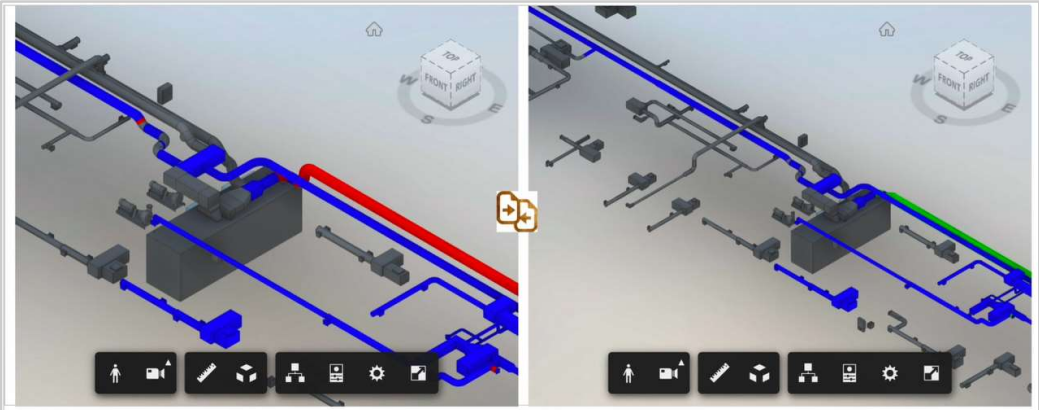
Diff

What it does: Compares two versions of a model and visualizes the differences found in the Forge Viewer. Differences can be in geometries and properties. Elements may be added, modified and removed.

Code: GitHub sample

Author: Xiaodong Liang, Autodesk





All Added

Added Items (25)

name	category	level	RC	Free
Round Elbow [1187817]	Revit Duct Fittings	Arch-FIRST FLOOR	Duct Fittings	16"ø·
Round Elbow [1187825]	Revit Duct Fittings	Arch-FIRST FLOOR	Duct Fittings	16"ø·
Round Elbow [1187827]	Revit Duct Fittings	Arch-FIRST FLOOR	Duct Fittings	16"ø·

All Removed

Removed Items (30)

name	category	level	RC	Free
Round Elbow [853203]	Revit Duct Fittings	Arch-FIRST FLOOR	Duct Fittings	4"ø·
Round Elbow [914477]	Revit Duct Fittings	Arch-FIRST FLOOR	Duct Fittings	4"ø·
Round Elbow [914480]	Revit Duct Fittings	Arch-FIRST FLOOR	Duct Fittings	4"ø·

All Changed

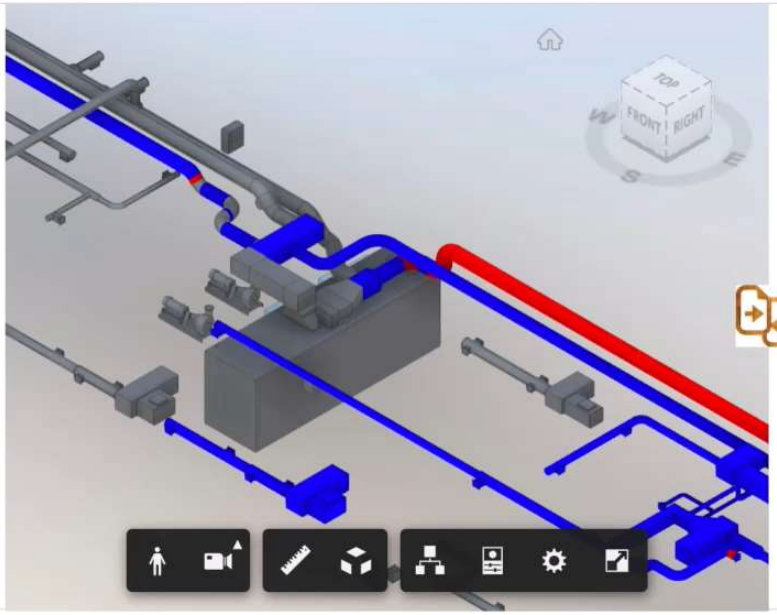
Changed Items (284)

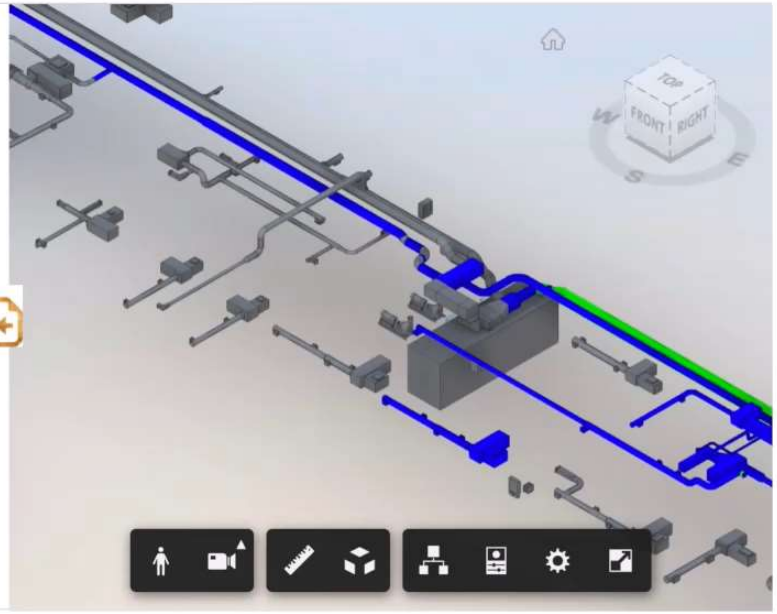
name	geometry changed?	property ch
Round Elbow [839567]	yes	yes
Round Elbow [839569]	yes	yes
Round Elbow [839571]	yes	yes

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

Added Items (25)

name	category	level	RC	Free
Round Elbow [1187825]	Revit Duct Fittings	Arch-FIRST FLOOR	Duct Fittings	16"ø-
Round Elbow [1187827]	Revit Duct Fittings	Arch-FIRST FLOOR	Duct Fittings	16"ø-
Rectangular Takeoff [1188475]	Revit Duct Fittings	Arch-FIRST FLOOR	Duct Fittings	6"x6"-

All Removed

Removed Items (30)

name	category	level	RC	Free
Round Elbow [853203]	Revit Duct Fittings	Arch-FIRST FLOOR	Duct Fittings	4"ø-
Round Elbow [914477]	Revit Duct Fittings	Arch-FIRST FLOOR	Duct Fittings	4"ø-
Round Elbow [914480]	Revit Duct Fittings	Arch-FIRST FLOOR	Duct Fittings	4"ø-

All Changed

Changed Items (284)

name	geometry changed?	property ch
Round Elbow [839567]	yes	yes
Round Elbow [839569]	yes	yes
Round Elbow [839571]	yes	yes

Change Analysis in Takeoff and Cost

Application of Diff in Estimate

What it does: Identifies changes in takeoff items, compares current and previous versions of a model and visualizes the differences in the Forge Viewer. Update the budgets data in Cost module.

Code: GitHub sample

Author: Zhong Wu, Autodesk

Chrome File Edit View History Bookmarks Profiles Tab Window Help 99+ 怀着冷却了的心窝漂远方 Thu May 19 6:53 PM

ACC Takeoff to Cost Sample | Autodesk Takeoff

localhost:3000 Update

AUTODESK FORGE

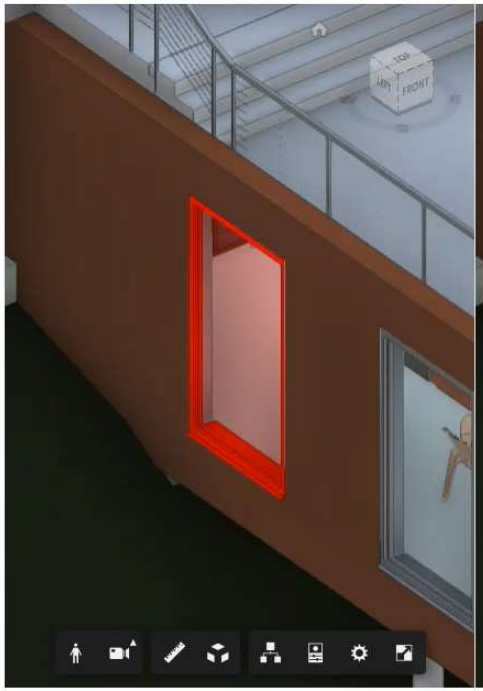
Enable my ACC Account

Sign Out Zhong Wu

ACC Project List

- Developer Advocacy Support
  - ACC Project Sample Zhong - 1
  - ACC Sample Project Zhong By API
  - ACC Sample Project Zhong By API 2
  - ACC Sample Project Zhong By API 3
  - ACC Sample Project Zhong By API 4
  - ACC Sample Project Zhong By API 5
  - ACC Sample Project Zhong UI -1
  - AU ACC TakeOff Sample Project
    - Package: 3rd Concrete
    - Package: Living Room Sample
      - A001
      - A101
      - A102
      - A103
      - A104
      - A105
      - rac_basic_sample_project.rvt - v6
        - Cavity wall_sliders
        - SIP 202mm Wall - conc clad
        - Standard
        - Standard
        - Standard
        - Standard
        - Standard
        - Standard
        - Standard
        - Standard
        - Wall - Timber Clad
        - Wall - Timber Clad
      - Project1.rvt - v2
    - Package: Sample Package
    - Package: Windows Level 1
    - Package: Windows Level 2
  - Check my project icon
  - Component Library Project 024d6467-98d4-448d-bb85-f8f6602b117d
  - JM ACC Test
  - MH ACC Forge Sandbox 002
  - Sample 2
  - Sample Template
  - Xiaodong-Test-Project
  - test sample

Version: 6



Version: 7

Added Items (1)

Name	Svf2Id
Single Window [1103656]	6117

Removed Items (3)

Name	Svf2Id
Single Window [735853]	597
M_Window Tag [941201]	5329
Single Window [1103176]	6047

Changed Items (9)

Name	svf2Id
Basic Wall [849032]	276
Railing [933840]	436
Basic Wall [428745]	442
Railing [567887]	491

Price Book (US$)

Show 10 entries Search:

Type	Unit Price($)	Unit
Door	836	nr
Floor	136	m2
Glass	230	nr
Wall	467	m2
Window	553	nr

Showing 1 to 5 of 5 entries Previous 1 Next

Takeoff Package Budget

Show 10 entries Search:

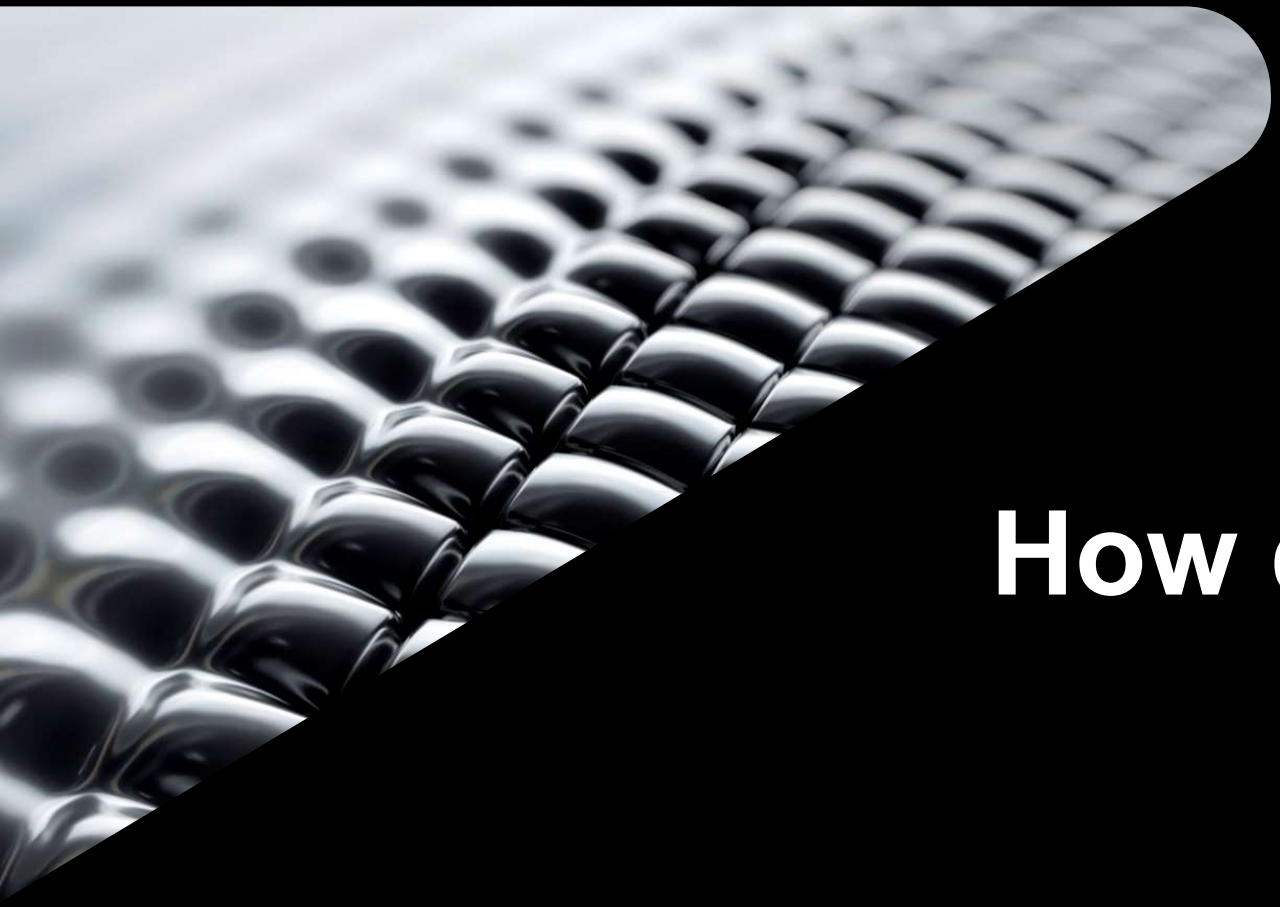
Element	Quantity	Unit	Unit Price($)	Amount($)
Door	5.00	nr	836	$4,180.00
Floor	88.07	m2	136	$11,977.18
Glass	0.00	nr	230	$0.00
Wall	343.14	m2	467	$160,244.25
Window	8.00	nr	553	$4,424.00

Showing 1 to 5 of 5 entries Previous 1 Next

Send to Cost

Cost Estimation of takeoff items for the latest version

Name	Current	Latest	Quantity	Cost($)	Svf2Id
Basic Wall [849032]	45.78	49.83	4.05(m2)	1891.35	276
Basic Wall [428745]	0.00	46.33	46.33(m2)	21636.11	442
Single Window [735853]	1	0	-1(nr)	-553.00	597
Single Window [1103176]	1	0	-1(nr)	-553.00	6047
				22593.46	



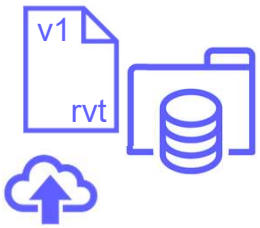
How does it work?

Model Properties API

How does it work?

Basic Workflow

Model Derivative | Model Properties



DM/Docs

Files uploaded to Docs via UI, Desktop Connector and API

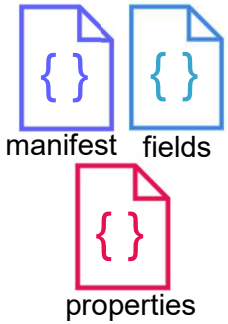
Model Derivative

Files are translated to svf/svf2. In Docs, translation is triggered automatically.

Design Data

PropertyDb

Property database, manifest, resources.



Index

Basic index generates three downloadable JSON resources:

- manifest
- fields
- properties

Query

results: JSON files matching the query filter.

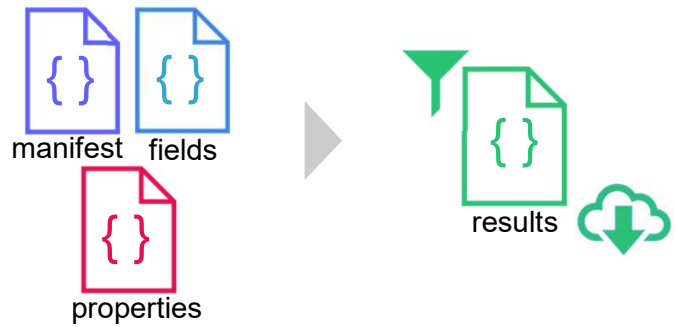
How does it work?

Resources Generated

Resource	Type	Description
manifest	JSON	Manifest for index or query detailing the seed files, svf2 propertyDb used to generate index rows
fields	NDJSON	The set of unique fields (property types) extracted for an index or query
properties	NDJSON	The raw objects property values for the index.
results	NDJSON	The object property results from a query executed.

NDJSON = new-line delimited JSON

Model Properties



Index

Basic index generates three downloadable JSON resources:

- manifest
- fields
- properties

Query

results: JSON files matching the query filter.

How does it work?

Index Endpoints

	Endpoints	
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 endpoints

How does it work?

Index Endpoints for Creation

	Endpoints	
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**

- Create basic index – “lazy”
  - First time – start the indexing job and cache the results
  - Once executed – use the cache
  - Cached 30 days since the last used
- Poll for progress
  - state: PROCESSING, FINISHED, FAILED
- Response JSON is identical
- stats: objects (# of object)
- Create 3 downloadable json.gz resources
  - Manifest, fields, properties

Ex. Create basic index – 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"
           }
       ]
     }'
```

request

```
[
    {
        "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
    }
]
```

response

Ex. Polling for Progress - GET indexes/:indexId

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

request

response


```
{
    "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"
}
```

How does it work?

Index Endpoints for Download

	Endpoints	
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**

- (Optional) download
  - 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 & Version

SVF2 Prop DB Resource URNs

Viewables

Index object count and byte size

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"}
```


Field Key → SQL column name	Category	Type	Name	UOM (Unit of Measurement)

Basic Properties (json.gz)


```
{
  "svf2Id": 68,
  "lineageId": "a19f7db",
  "externalId": "b5c4b31f-321a-418d-a61a-0c8e326aa154-0003f740",
  "lmvId": 2388,
  "databaseId": "3747dccf",
  "props": {
      "p00723fa6": "Main Model",
      "p13b6b3a0": "HSS7X7X.250",
      "p153cb174": "HSS-Hollow Structural Section-Column [259904]",
      "p188478f2": 0.485383241976329e0,
      "p20d8441e": "Structural Columns",
      "p30db51f9": "HSS-Hollow Structural Section-Column",
      "p5eddc473": "Revit Structural Columns",
      "p63ed81bb": "Superstructure",
      "p6637df3c": "Metal - Steel - ASTM A500 - Grade B - Rectangular and Square",
      "pbadfe721": "BEARING",
                    ...

      "pddd761c6": "FOUNDATION PLAN",
      "pe61a57c3": 0e0,
      "pee815a7f": "None",
      "pef87fde6": 0e0,
      "pf4ca60ab": 5833333333333334e-16,
  },
  "propsHash": "bcde34b3",
  "propsIgnored": {
      "p6a81eafd": 2386,
      "p93e93af5": 2387
  },
  "geomHash": "TCC2Cc9tvO4EVazM73O8BQ",
  "bboxMin": {
      "x": -1413565004170512e-13,
      "y": -5410244931321833e-14,
      "z": 10000000002097008e-14
  },
  "bboxMax": {
      "x": -14063352214982766e-14,
      "y": -53379471045994805e-15,
      "z": 11101965298365471e-14
  },
  "views": [
      "e7fda9d5",
      "12fcb372"
  ]
}
```


Object IDs, keys for index manifest JSON. (Type and ChangeType IF diff index)

Index field property values

Property hash + properties ignored when calculating the hash

SVF2 geometry hash

Bounding box min/max for viewable object

Viewables containing the object (manifest keys to viewable)

How does it work?

Index Endpoints for Query

	Endpoints	
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**

- Build and run query.
  - Index queries are described using custom JSON schema, (which is converted to a filter expression. AWS S3 Select)
  - Columns can be restricted. Can use alias (have different header)
- Poll for progress.
  - state: PROCESSING, FINISHED, FAILED

Sample Query: Get Revit Classification with Column Transform

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}
```

Query

```
{
    "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
    }
}
```

Row has Revit classification

Views array has count more than 0

Columns define alias

Equivalent in 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
```

How does it work?

Index Endpoint to Download Query Results

	Endpoints	
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**

- Download the query results
  - use the queryResultsUrl in query call or query id to download the index rows which match the submitted query expression.
  - Result: line delimited JSON
  - a sub-set of the property index rows
  - Format is the exactly the same as properties we saw earlier.

How does it work?

Diff Endpoints

	Endpoints	
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 - The steps are the same as Index
- Specify two version urn's to compare:

```
{
  "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"
        ]
    }
}
```


Type if diff index

Previous (prev) object embedded in current row. Lineage manifest key & viewer id.

Array of property keys which have values different to current

Previous bounding boxes, hashes and viewable keys in manifest

Basic Index Row vs. Diff Index Row

Description	Current version	Previous version
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*]

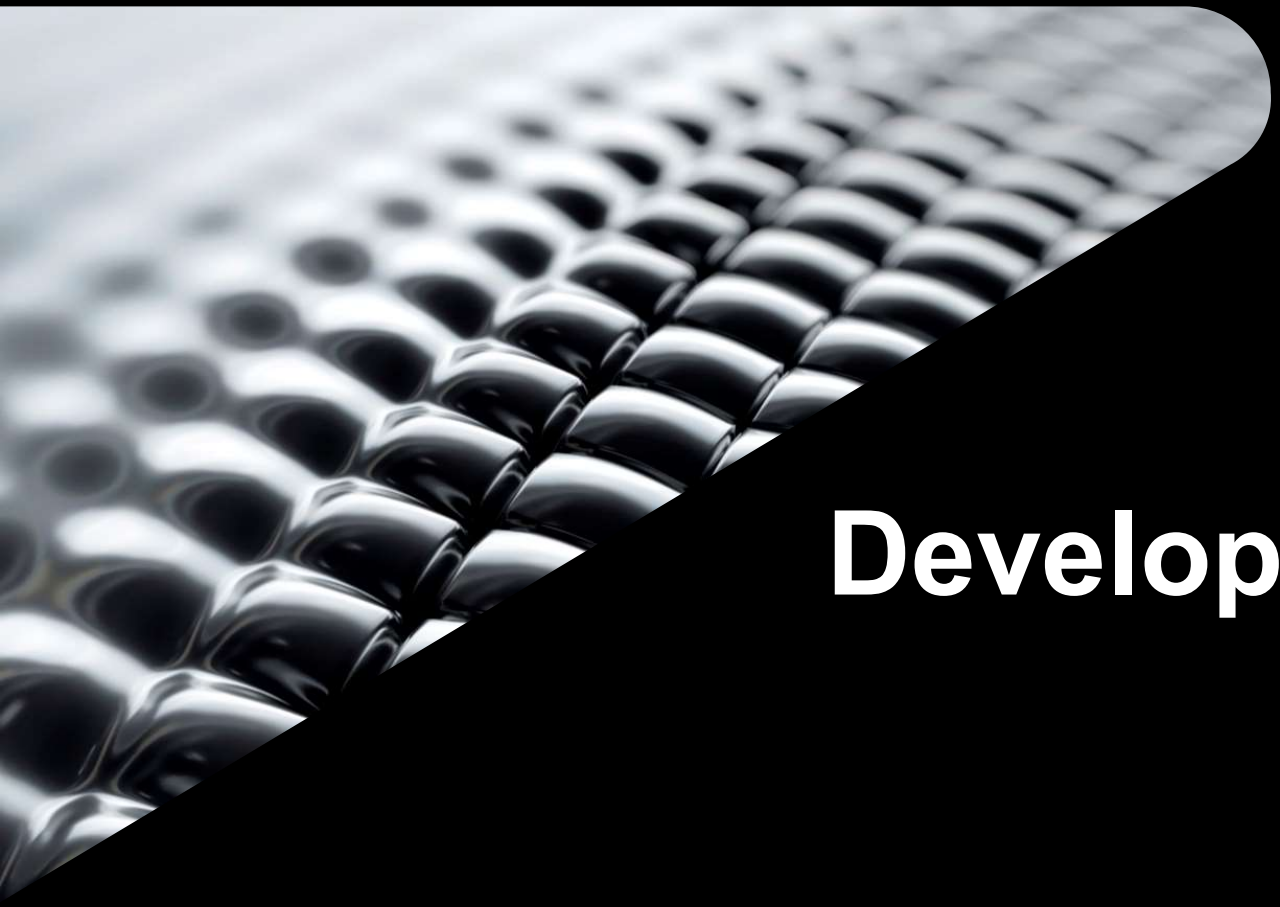
JSON Abstract Syntax Tree → S3 Select(AWS)

```
$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
```



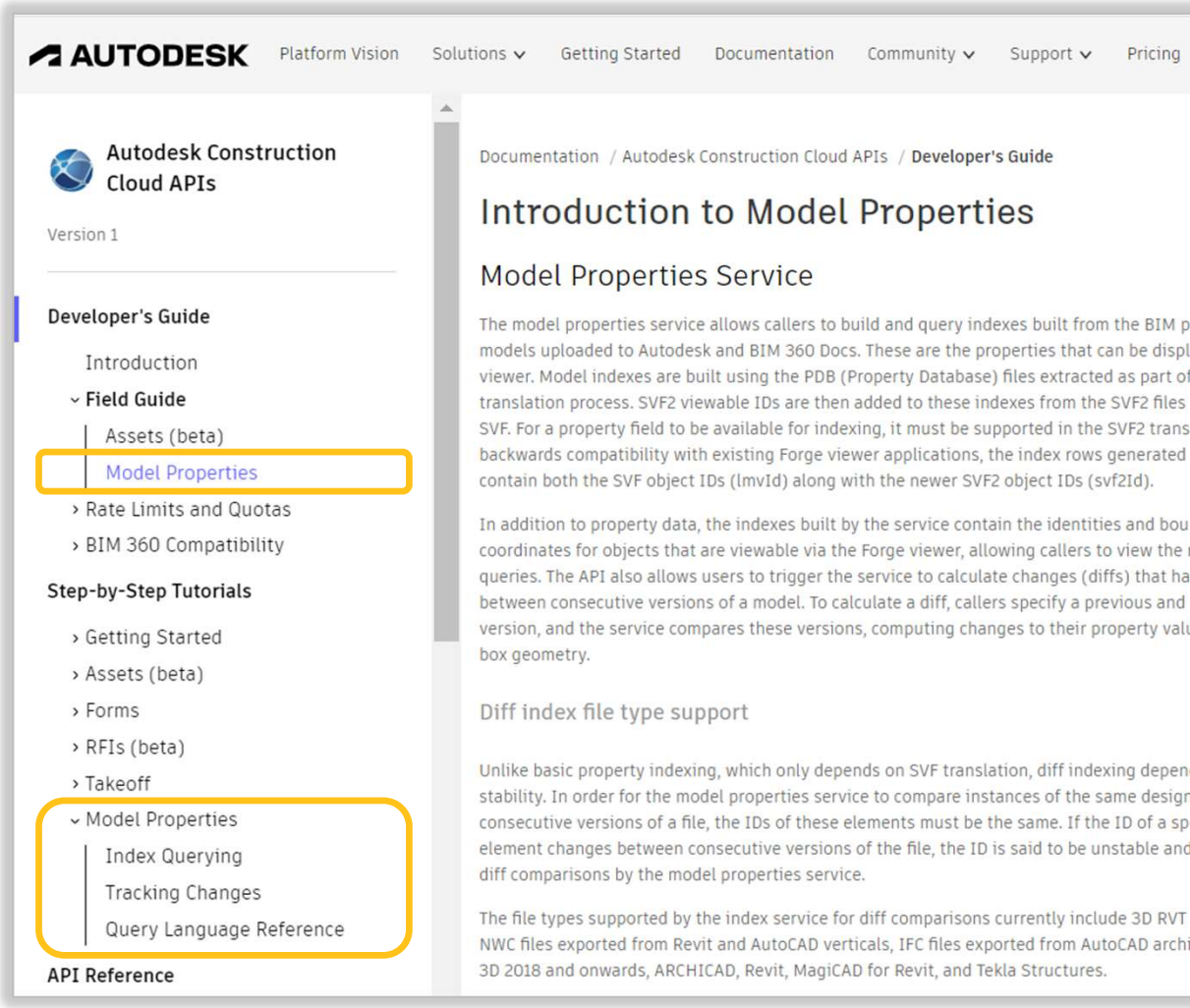
Developer Resources

Model Properties API

Developer Resources

Documentation

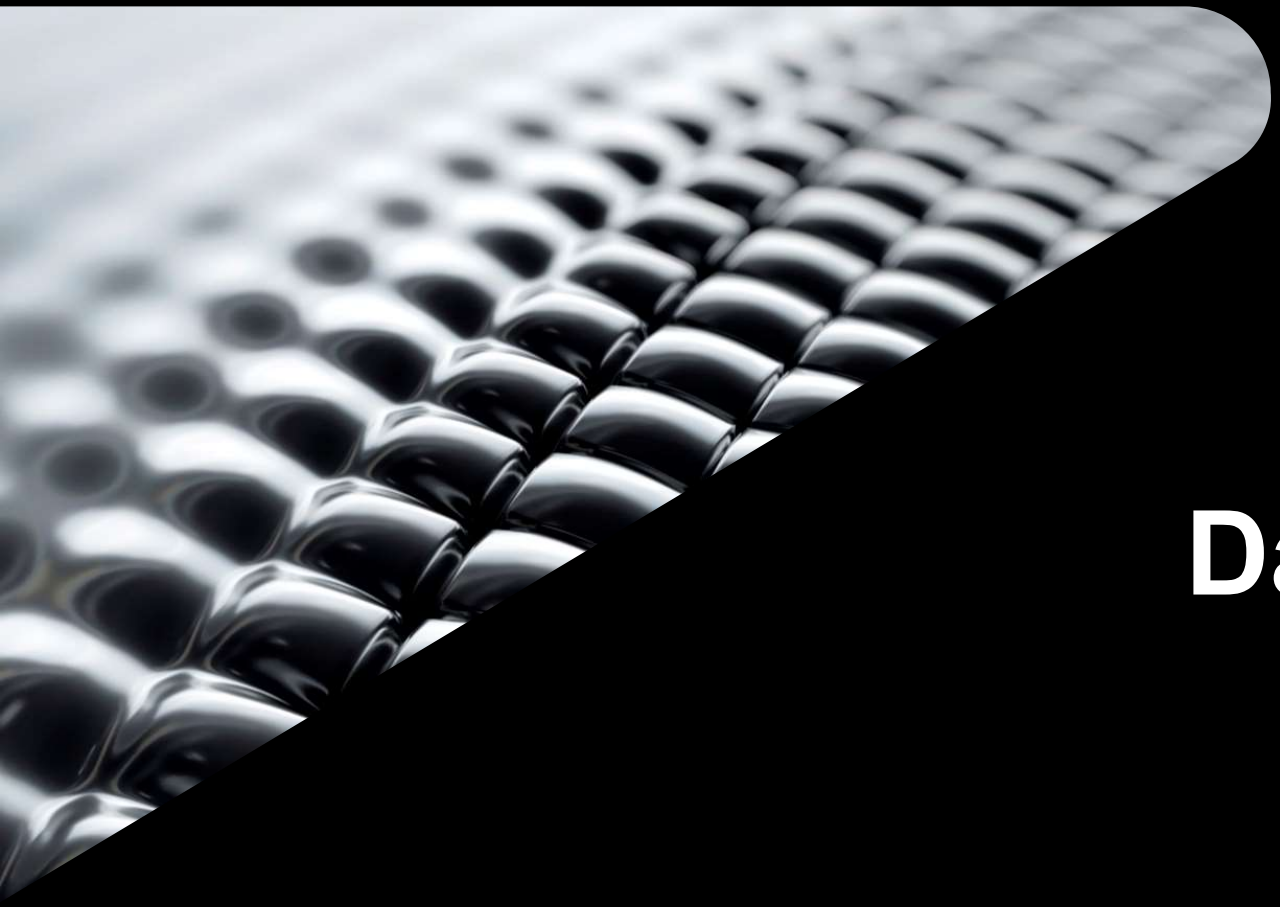
- Field Guide
  - Introduction to Model Properties
- Step-by-Step Tutorials
  - Index Querying
  - Tracking Changes
  - Query Language Reference
- Reference Guide
  - Index
  - Diff



Developer Resources

- Code Samples on GitHub
  - Postman Collection (correspond to three Step-by-Step tutorials)
  - Model Properties API Walkthrough in PowerShell Core (scripting to explore query language)
  - Element Filtering and Partial Model Load (Integration with Viewer)
  - Compare Two Versions (Integration with Viewer)

- Blog Post
  - "BIM 360/ACC Model Properties API"
    https://forge.autodesk.com/blog/bim-360acc-model-properties-api
    includes links to the resources
  - Search more with "Model Properties", e.g.,
    https://forge.autodesk.com/blog/model-properties-api-vs-model-derivative-api

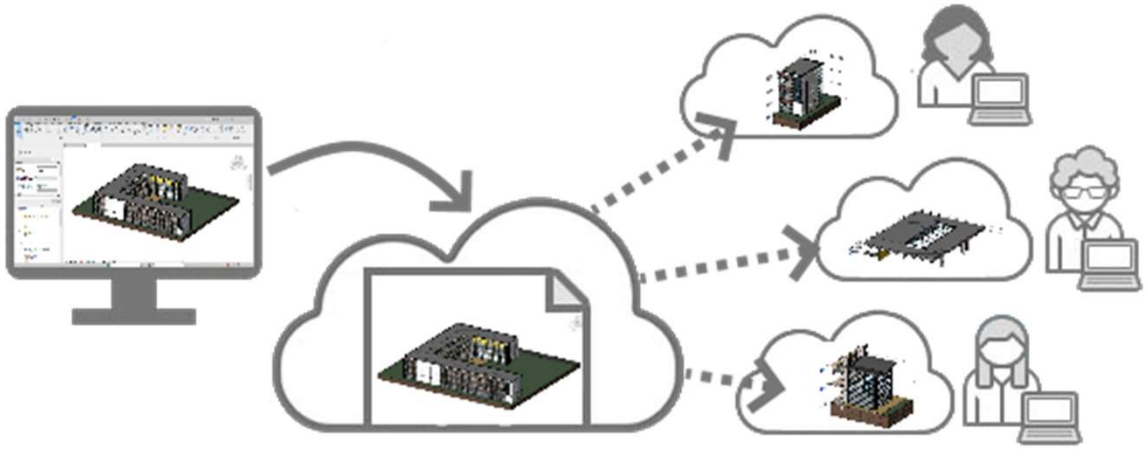


Data Exchange

Data Exchange

What is it?

- Used to collaborate with team members
- Share only a portion of the Revit model
- Models published to Autodesk Docs can have Data Exchanges made from 3D views
- Use cases in the product
  - Revit to Inventor (via Docs)
  - Revit to Power Automate
  - Data Exchange Connector for Revit (public beta)
  - Revit to/from Rhino (public beta)
- Read only API in beta



Revit to Inventor via Docs

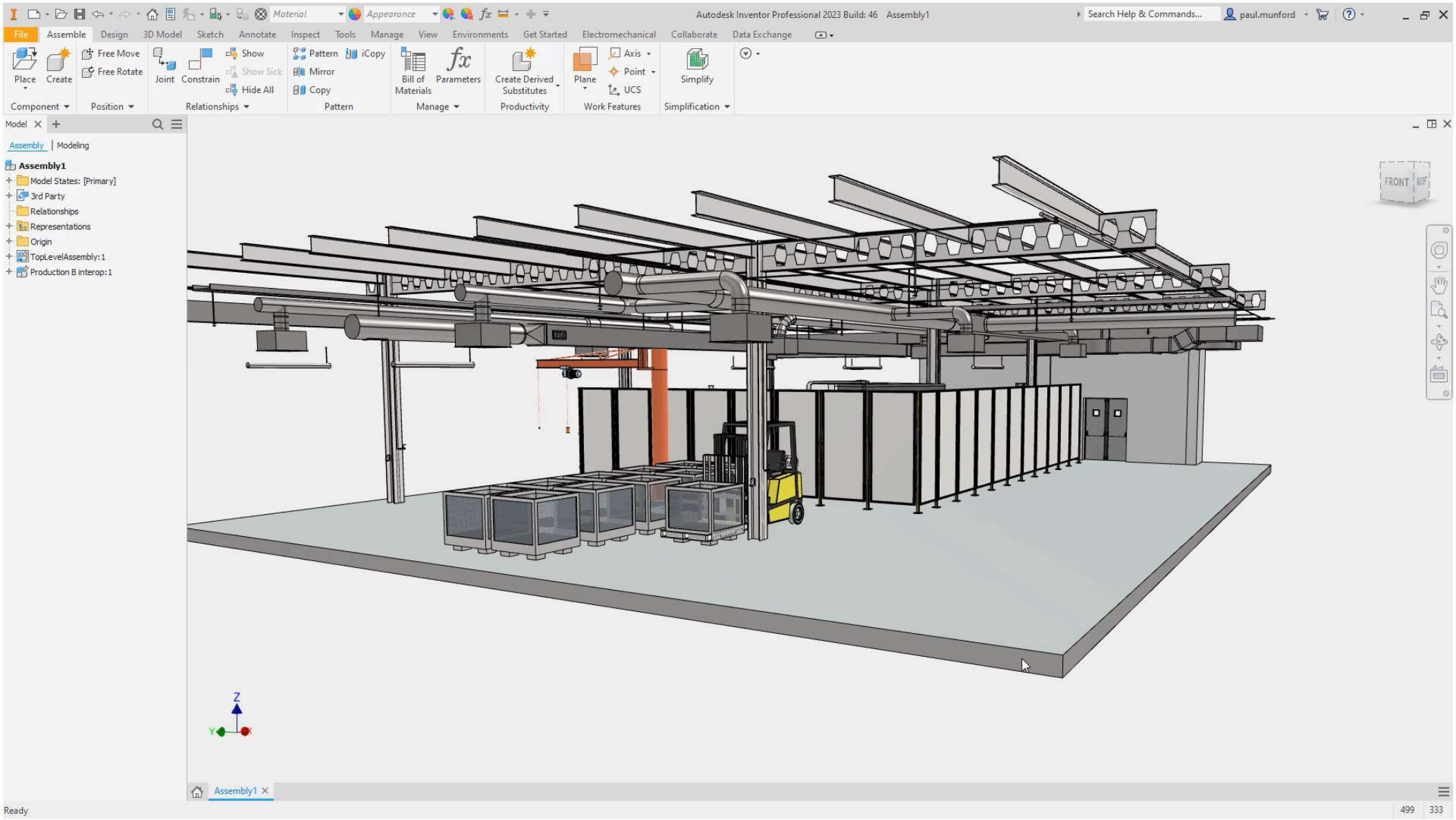
Data Exchange Ex.1

Workflow

1. Revit: Create a 3D view(s) with elements you want to share. Save to cloud/Docs
2. Docs: view model >> Create Exchange
3. Inventor: Load Data Exchange

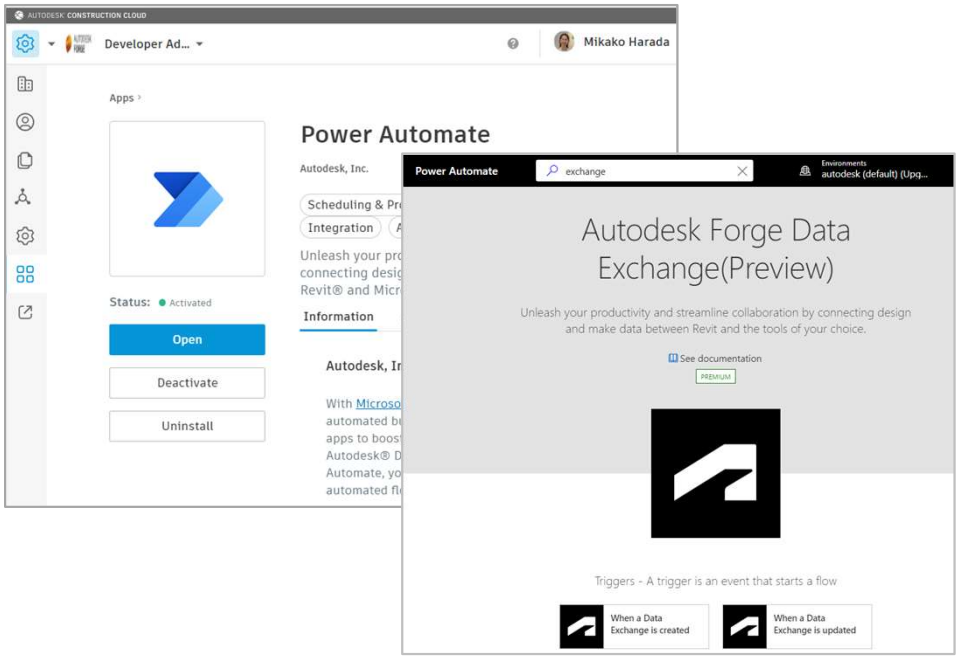
Features

- Share only the portion of the model
- Once created, changes are updated automatically
- 2023 versions; US hubs
- More info: Product help


Autodesk Inventor Professional 2023 Build: 46 Assembly1
File
Assemble
Design
3D Model
Sketch
Annotate
Inspect
Tools
Manage
View
Environments
Get Started
Electromechanical
Collaborate
Data Exchange
Place
Create
Free Move
Free Rotate
Joint
Constrain
Show
Show Sick
Hide All
Pattern
iCopy
Mirror
Copy
Bill of Materials
Parameters
Create Derived Substitutes
Plane
Axis
Point
UCS
Simplify
Component
Position
Relationships
Pattern
Manage
Productivity
Work Features
Simplification
Model
Assembly
Modeling
Assembly1
Model States: [Primary]
3rd Party
Relationships
Representations
Origin
TopLevelAssembly:1
Production B interop:1
FRONT
Z
Y
X
Assembly1
Ready
499
333

Revit to Power Automate

Data Exchange Ex.2

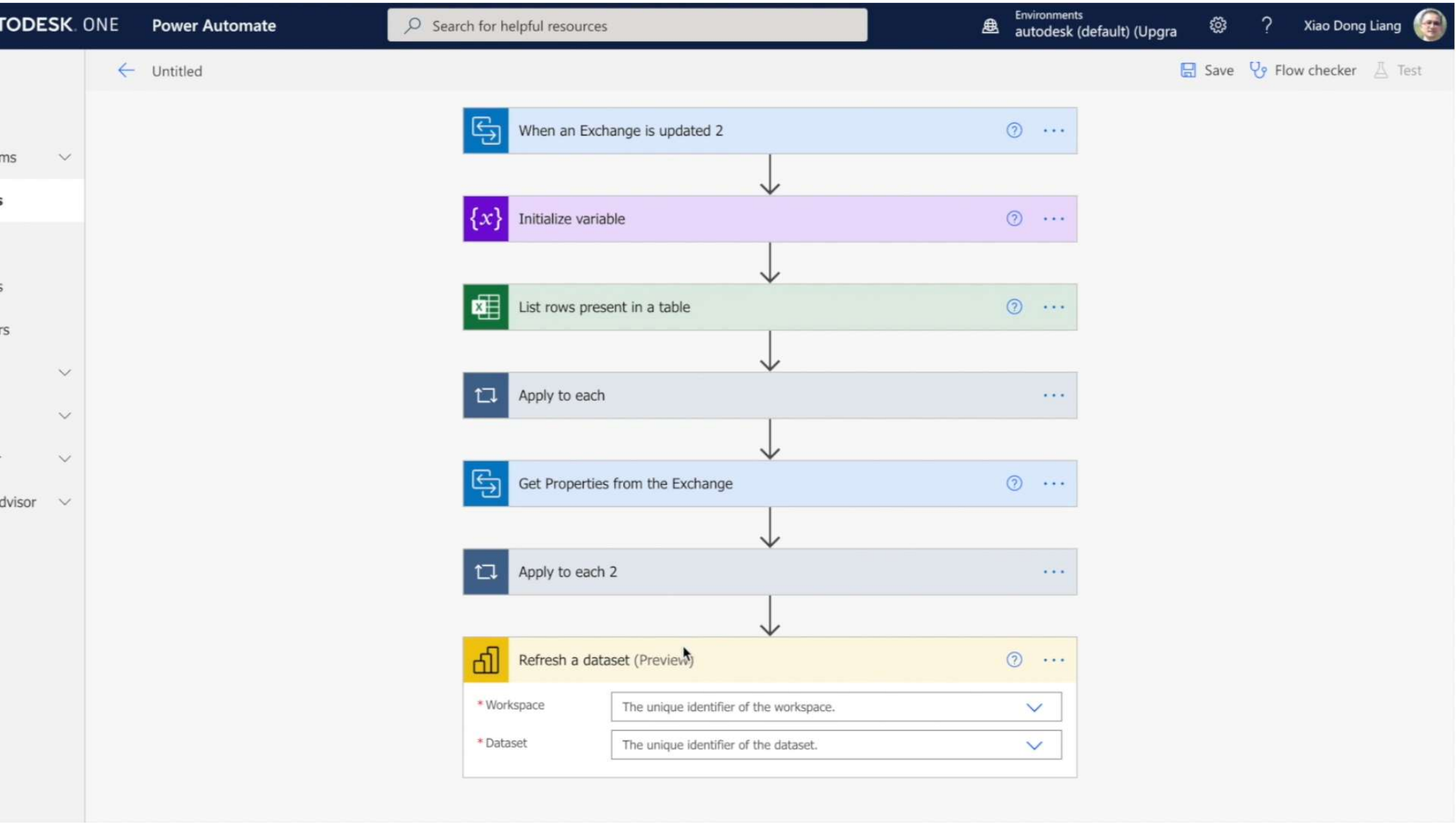


Workflow

1. Revit: Create a 3D view(s) to share. Save to cloud
2. Docs: view model >> Create Exchange
3. MS Power Automate Connector

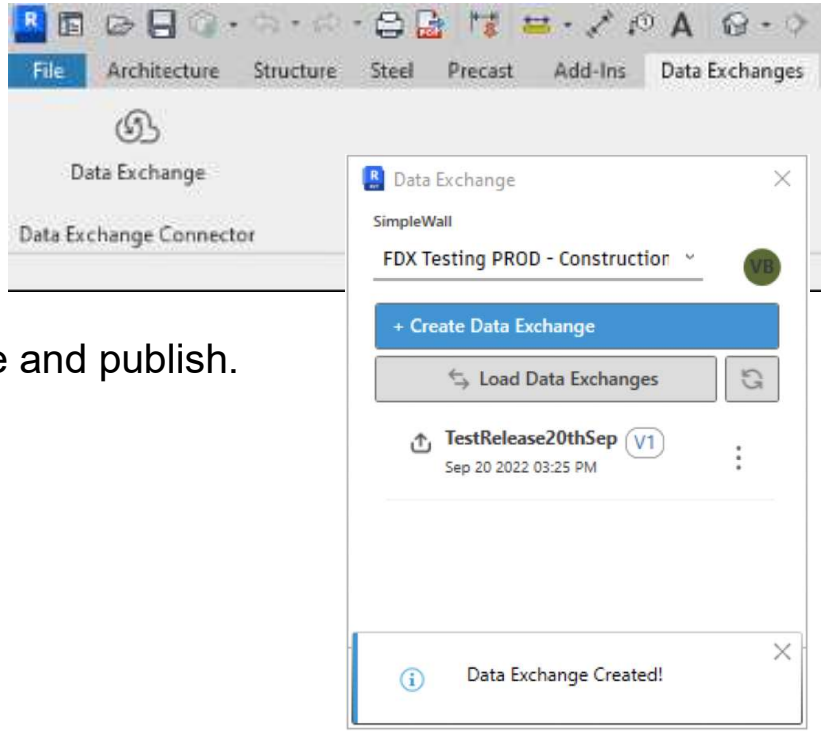
Features

- Low code, no code environment to automate workflow
- e.g., Excel, PowerBI
- https://blogs.autodesk.com/revit/2022/04/29/revit-2023-automate-workflows-using-revit-data-with-microsoft-power-automate/
- https://learn.microsoft.com/en-us/connectors/autodeskforgedataexc/


TODESK. ONE
Power Automate
Search for helpful resources
Environments
autodesk (default) (Upgra
Xiao Dong Liang
Untitled
Save
Flow checker
Test
When an Exchange is updated 2
Initialize variable
List rows present in a table
Apply to each
Get Properties from the Exchange
Apply to each 2
Refresh a dataset (Preview)
* Workspace
The unique identifier of the workspace.
* Dataset
The unique identifier of the dataset.

Exchange Connector for Revit

Data Exchange Ex.3 (public beta)



Workflow

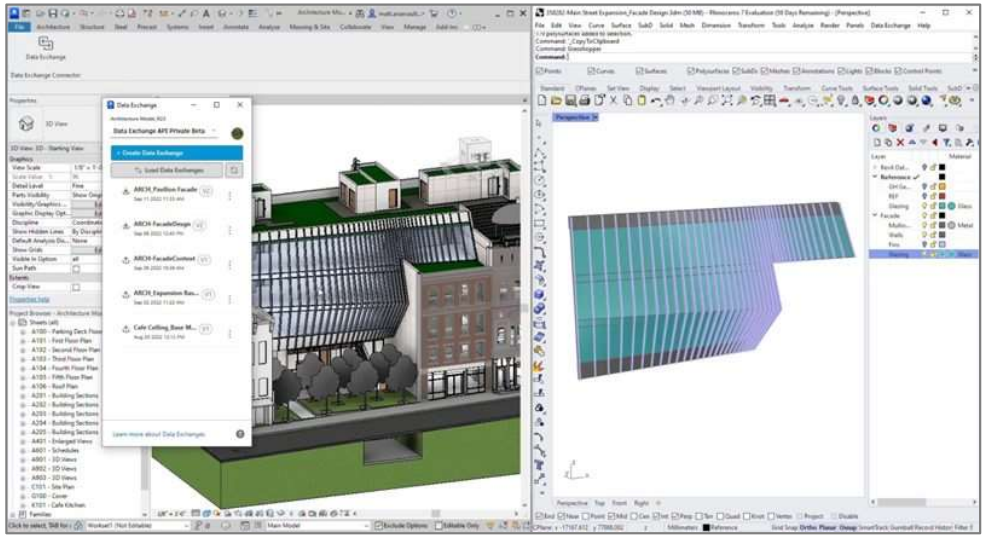
1. Revit: Create a 3D view(s) with elements you want to share and publish.
2. Revit: Create Exchange directly
3. Consumed by other apps

Features

- Create Exchange directly in Revit without switching to Docs
- Create, update, load
- https://blogs.autodesk.com/revit/2022/09/13/data-exchange-connector-for-autodesk-revit-now-in-public-beta/
- public beta (link in the above blog)

Revit to/from Rhino

Data Exchange Ex.4 (public beta)

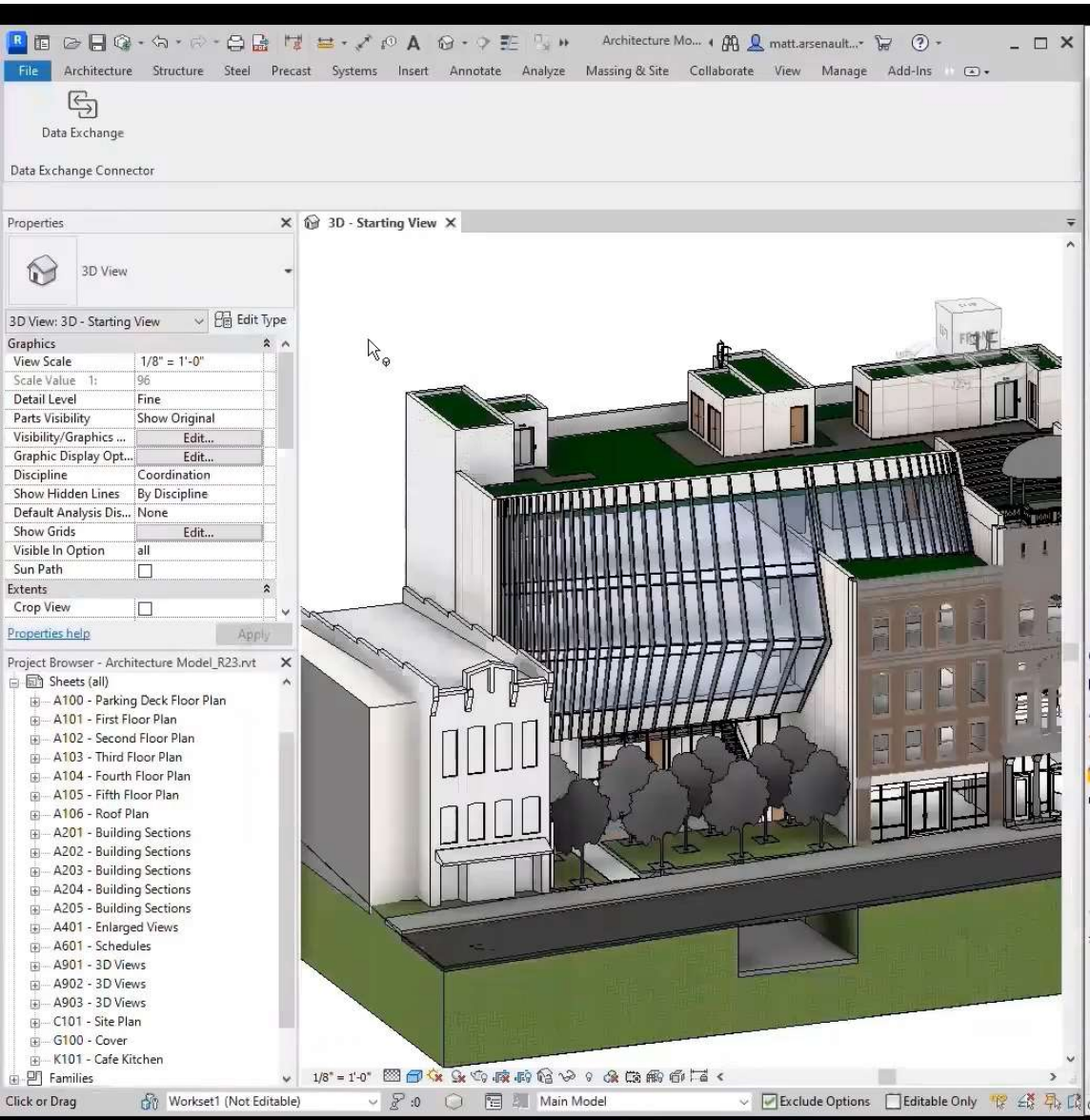


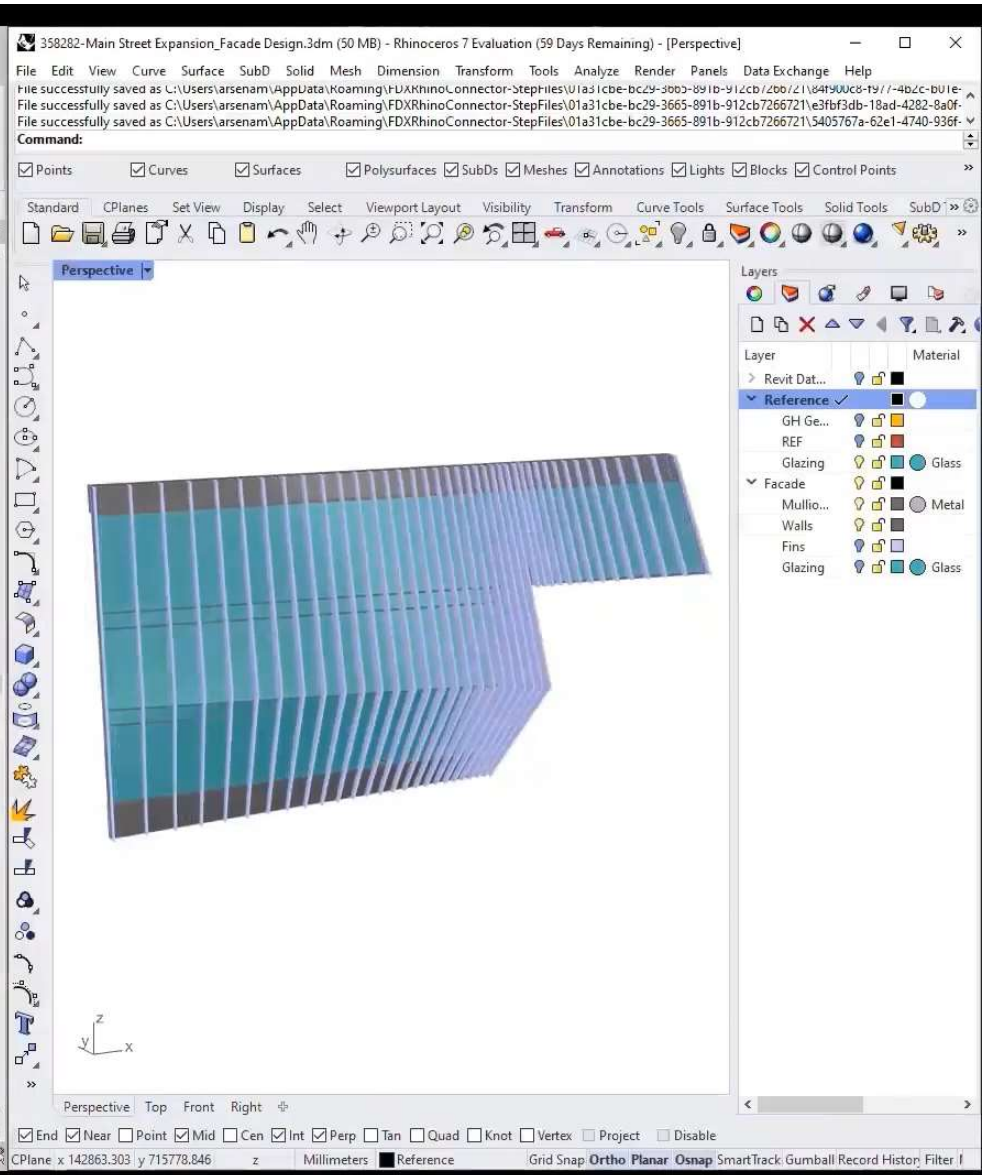
Workflow

1. Revit: Create a 3D view(s) to share and publish.
2. Revit: Create Exchange
3. Rhino: Consume using Rhino Connector
4. Do the same in the reverse direction

Features

- Two-way integrations
- Create, update, load
- https://blogs.autodesk.com/revit/2022/09/22/data-exchange-connector-for-mcneel-rhino-now-in-public-beta/
- public beta (link in the above blog)


Architecture Mo...
matt.arsenault...
File
Architecture
Structure
Steel
Precast
Systems
Insert
Annotate
Analyze
Massing & Site
Collaborate
View
Manage
Add-Ins
Data Exchange
Data Exchange Connector
Properties
3D View
3D View: 3D - Starting View
Edit Type
Graphics
View Scale 1/8" = 1'-0"
Scale Value 1: 96
Detail Level Fine
Parts Visibility Show Original
Visibility/Graphics ... Edit...
Graphic Display Opt... Edit...
Discipline Coordination
Show Hidden Lines By Discipline
Default Analysis Dis... None
Show Grids Edit...
Visible In Option all
Sun Path
Extents
Crop View
Properties help
Apply
Project Browser - Architecture Model_R23.rvt
Sheets (all)
A100 - Parking Deck Floor Plan
A101 - First Floor Plan
A102 - Second Floor Plan
A103 - Third Floor Plan
A104 - Fourth Floor Plan
A105 - Fifth Floor Plan
A106 - Roof Plan
A201 - Building Sections
A202 - Building Sections
A203 - Building Sections
A204 - Building Sections
A205 - Building Sections
A401 - Enlarged Views
A601 - Schedules
A901 - 3D Views
A902 - 3D Views
A903 - 3D Views
C101 - Site Plan
G100 - Cover
K101 - Cafe Kitchen
Families
3D - Starting View
1/8" = 1'-0"
Click or Drag
Workset1 (Not Editable)
Main Model
Exclude Options
Editable Only

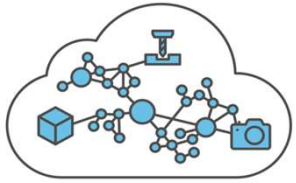

358282-Main Street Expansion_Facade Design.3dm (50 MB) - Rhinoceros 7 Evaluation (59 Days Remaining) - [Perspective]
File Edit View Curve Surface SubD Solid Mesh Dimension Transform Tools Analyze Render Panels Data Exchange Help
Command:
Points Curves Surfaces Polysurfaces SubDs Meshes Annotations Lights Blocks Control Points
Standard CPlanes Set View Display Select Viewport Layout Visibility Transform Curve Tools Surface Tools Solid Tools
Perspective
Layers
Layer Material
Revit Dat...
Reference
GH Ge...
REF
Glazing Glass
Facade
Mullio... Metal
Walls
Fins
Glazing Glass
Perspective Top Front Right
End Near Point Mid Cen Int Perp Tan Quad Knot Vertex Project Disable
CPlane x 142863.303 y 715778.846 z Millimeters Reference
Grid Snap Ortho Planar Osnap SmartTrack Gumball Record Histor Filter

Data Exchange API

Big Picture

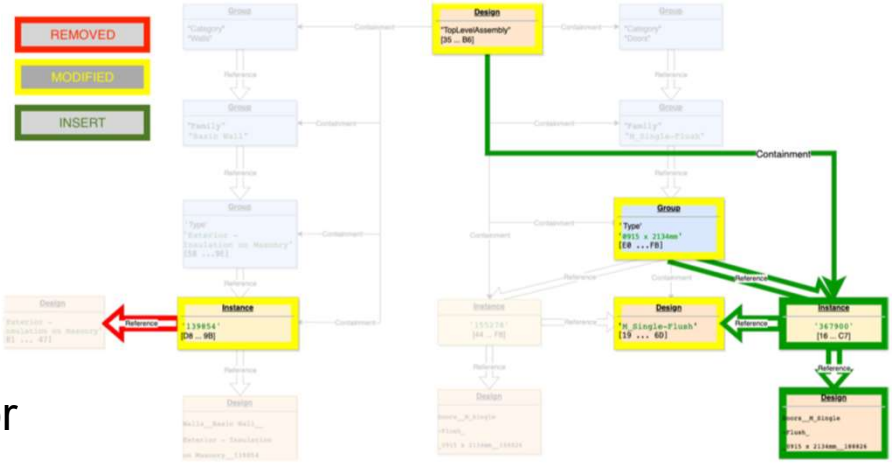
Data Exchange API - Big Picture

Data Structure



vs.

- Data is hosted directly in the cloud (not as a binary file)
- Data is structured as a graph
  - Assets (properties) : group, design, instance, binary, geometry
  - Relations (links between assets. reference/include)
  - Snapshot (state of data. **includes records of changes**)
  - Revisions (versions)
  - Collection (project)
  - Space (container for each exchange)
- Amount of data after many iterations

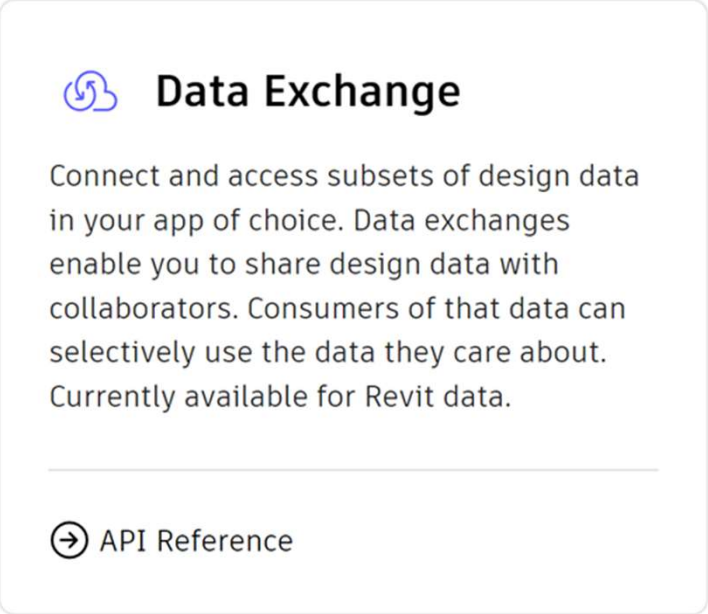


cf. Data Exchange API dedicated class by Denis Grigor

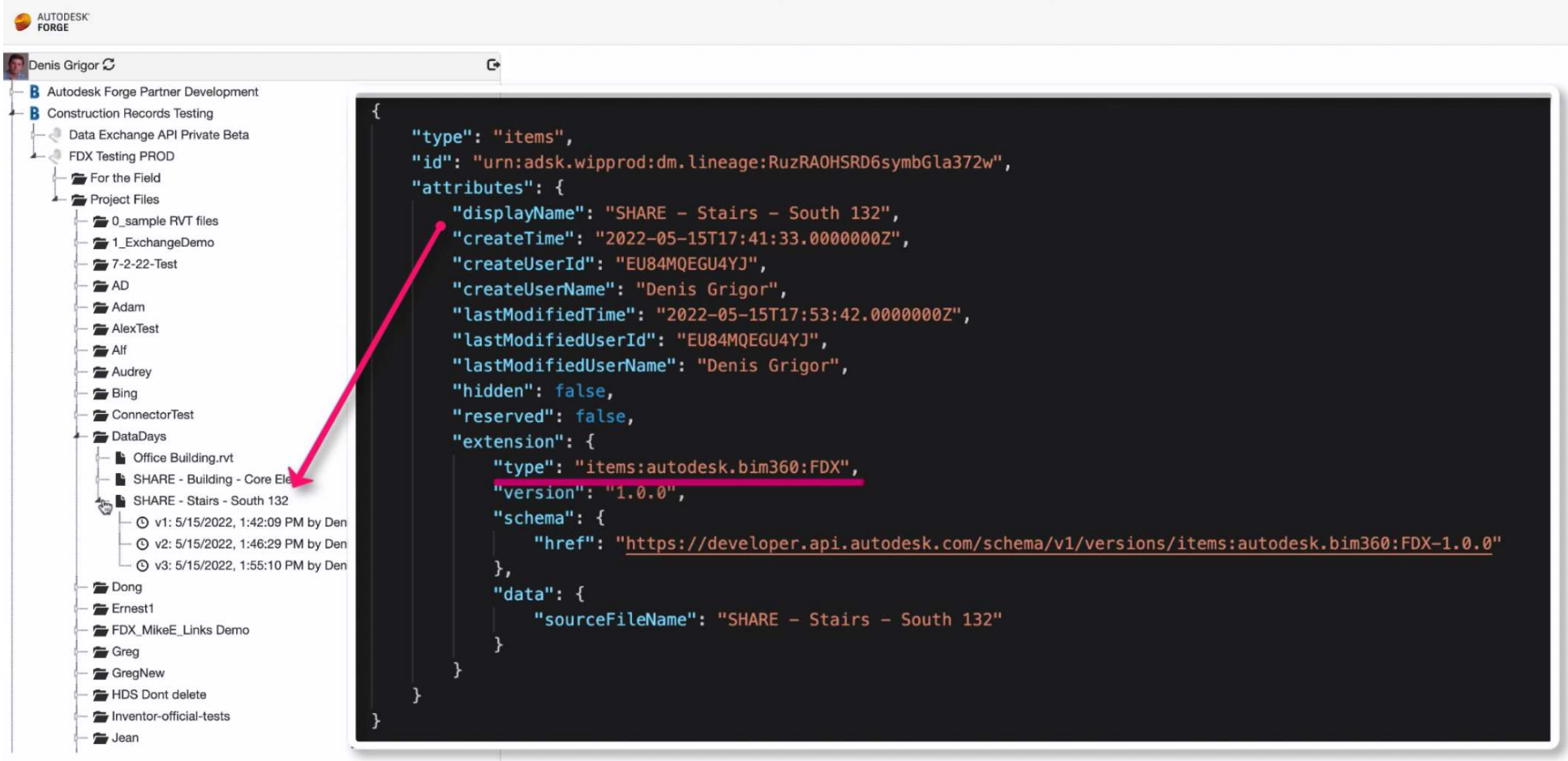
Data Exchange API

API

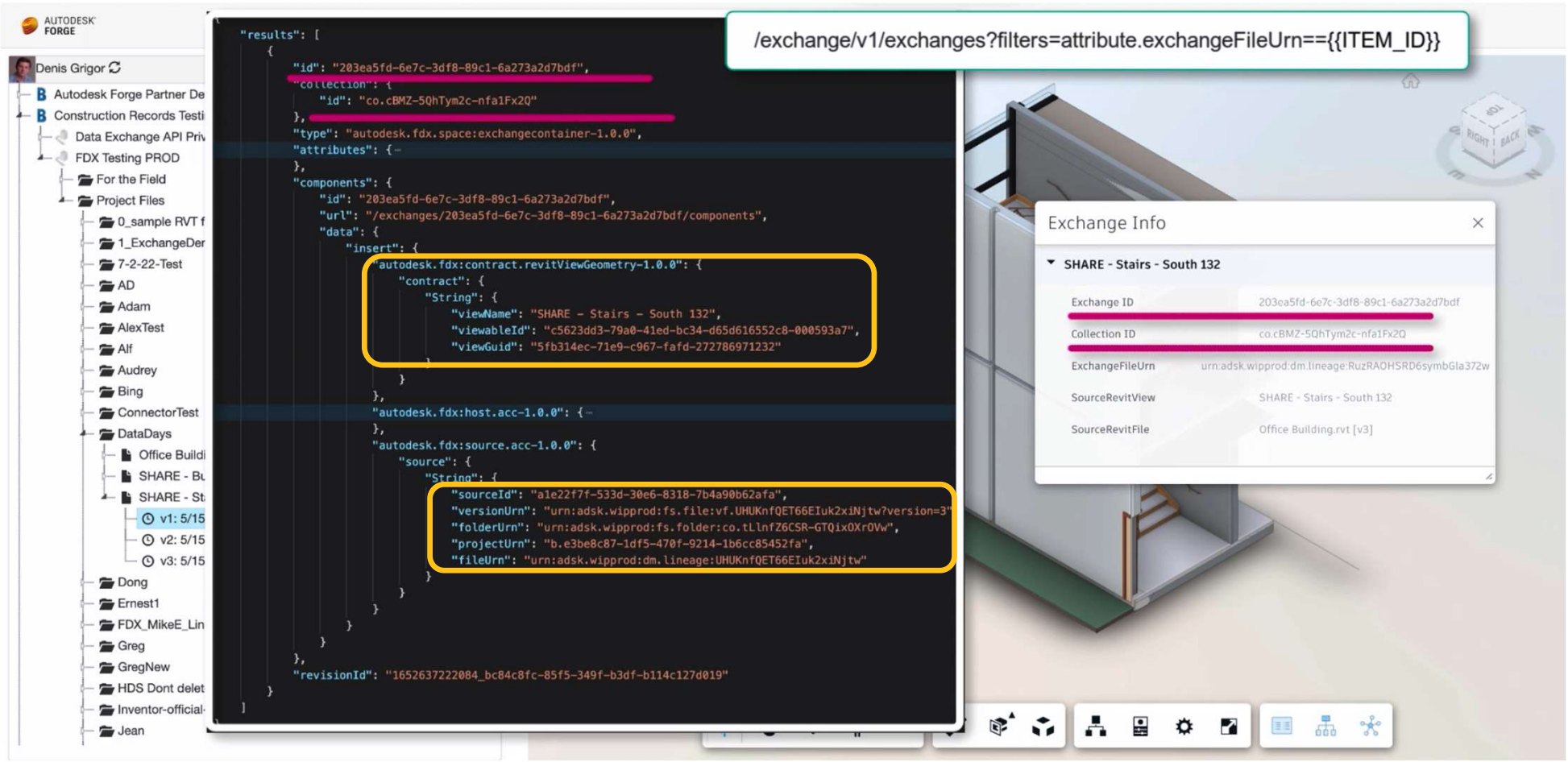
- Currently read-only public beta
  - get Exchange container
  - get Assets
  - get Relationships
  - get Snapshot revisions
- A step toward the future of cloud models
- Future
  - Difficult to filter/query >> GraphQL
  - Write API



Get Exchange Container



Get Exchange Container (cont.)



Learn More

Tuesday 8:00 AM, 222 & 4:30 PM, 217

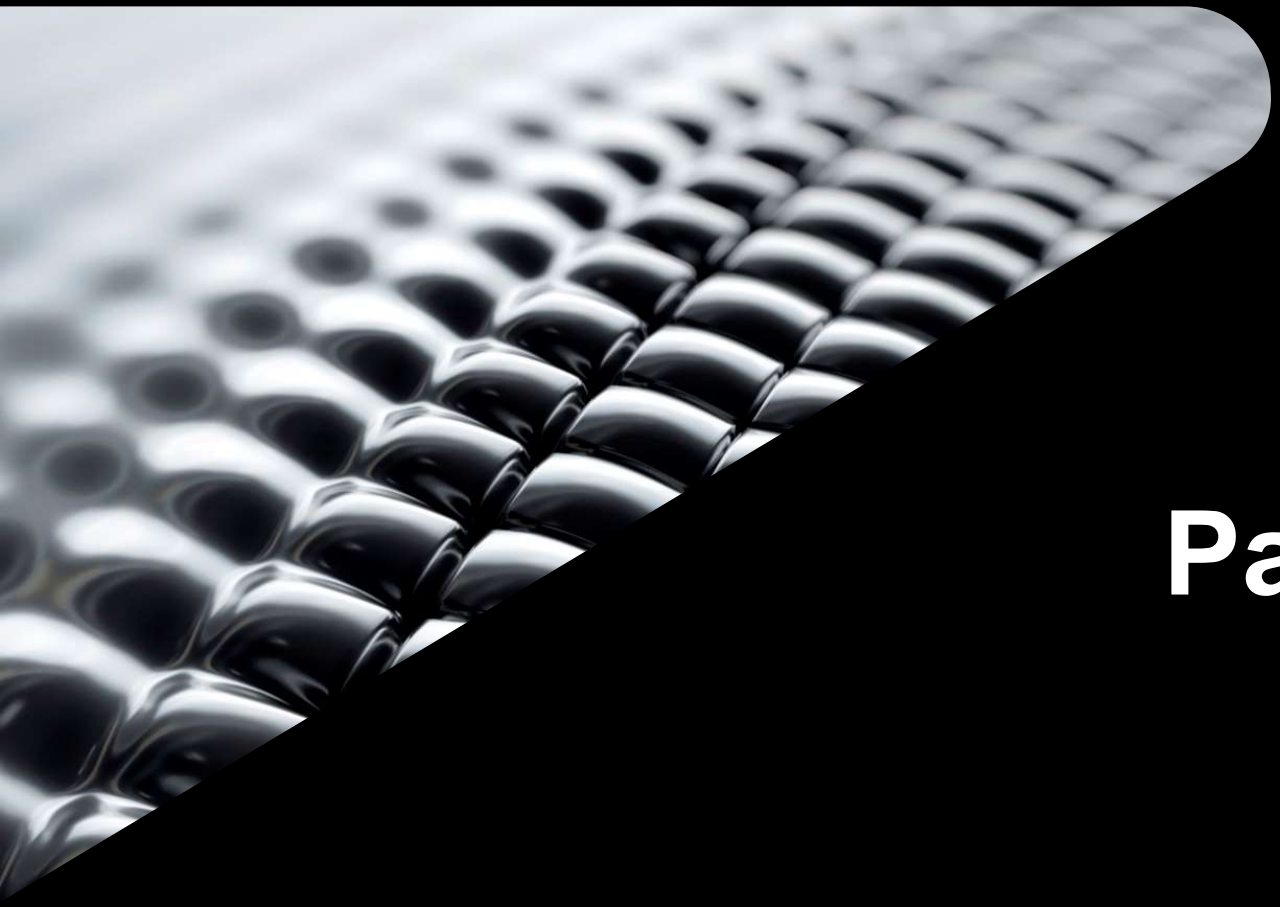
- FAB502817
  “Turn Out-Dated into Power-Automated: Revit Data Exchanges into Microsoft Power BI”,
  by Cesar Escalante and Philippe Videau

Wednesday 3:00 PM, 218

- SD502577
  “Autodesk Forge Data Road Map: Moving from Files to Cloud Data”,
  by Philippe Videau and Farzad Towhidi

Thursday 11:00 AM, 205

- SD501940
  “Up and Running with Data Exchange API: Where Does Model Derivative Fit?”
  by Denis Grigor and Jaime Rosales Duque



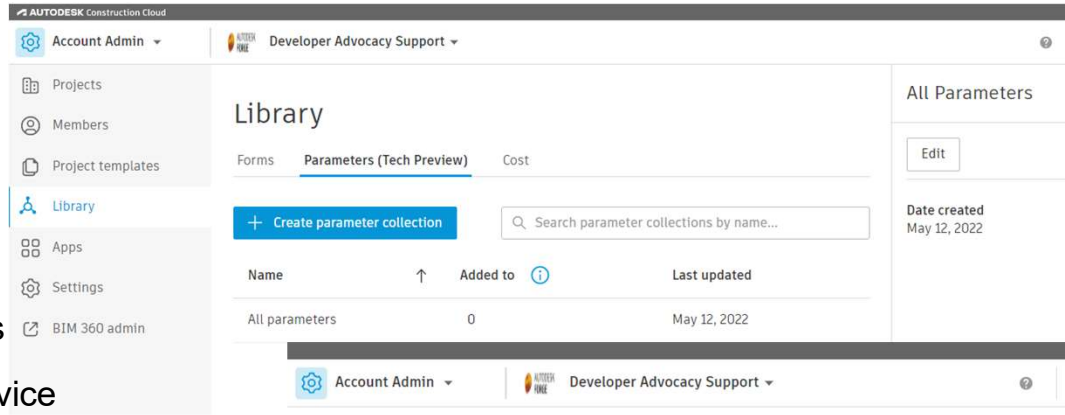
Parameters API

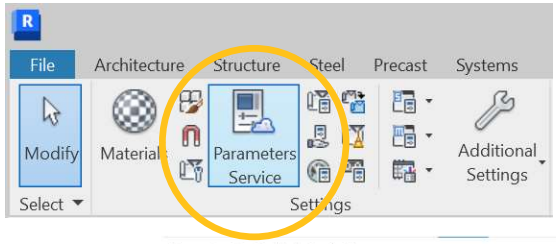
Parameters Service

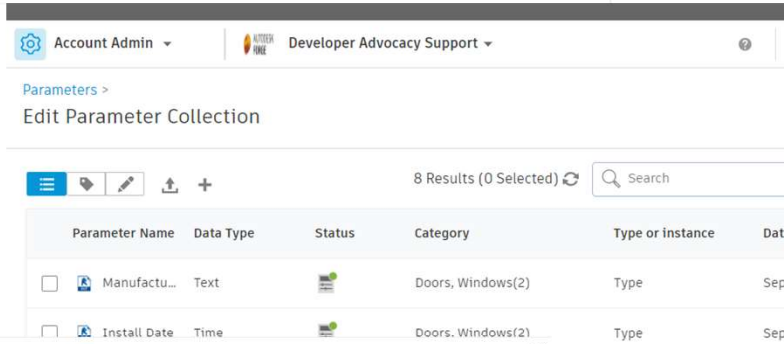
What is it?

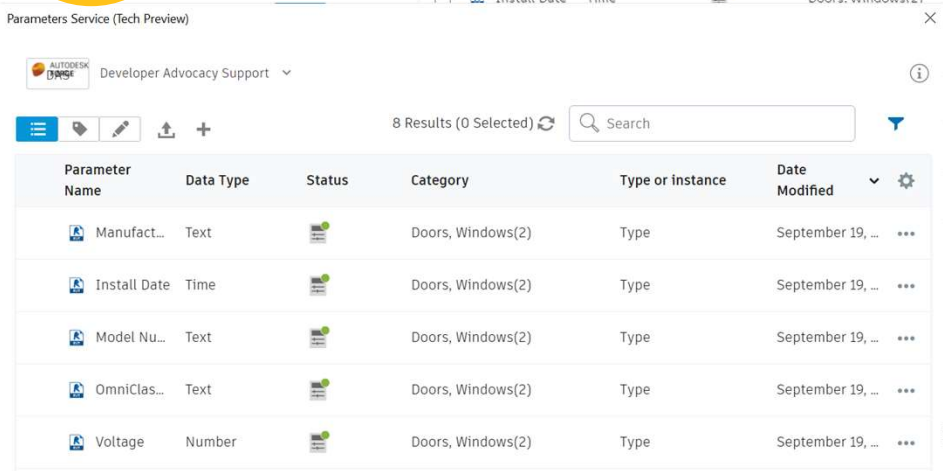
- Technology Preview Product Feature
  - ACC (US) >> Account Admin >> Library >> Parameters
  - Revit 2023 >> Manage >> Settings >> Parameters Service
- “Shared” parameters in the cloud

- Address challenges
  - Revit properties are user-defined
  - Shared among contents, projects, firms
  - Many versions of definitions in .txt files all over
  - Coordinate and collaboration
- Unified, hosted location to create, manage and share



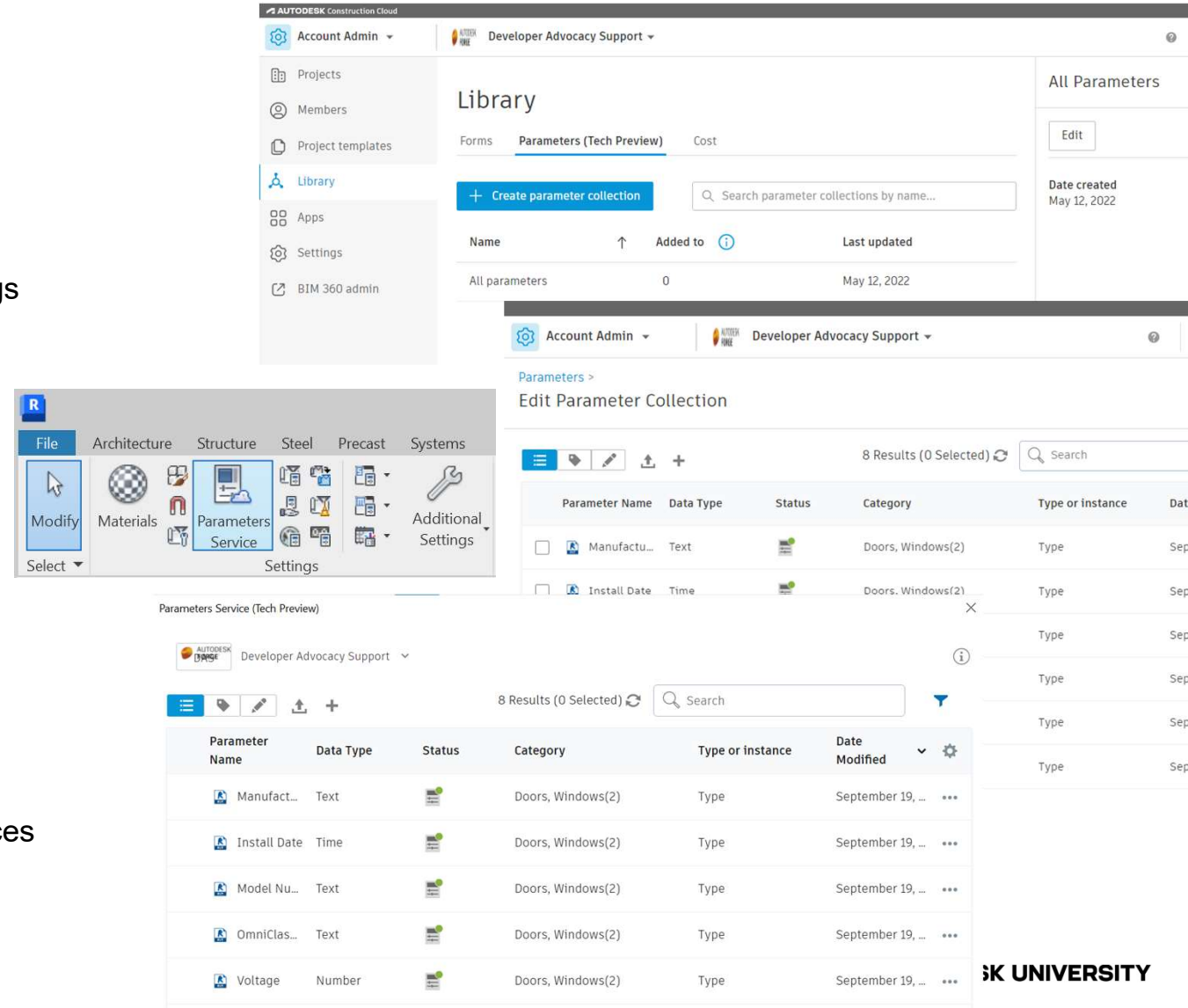


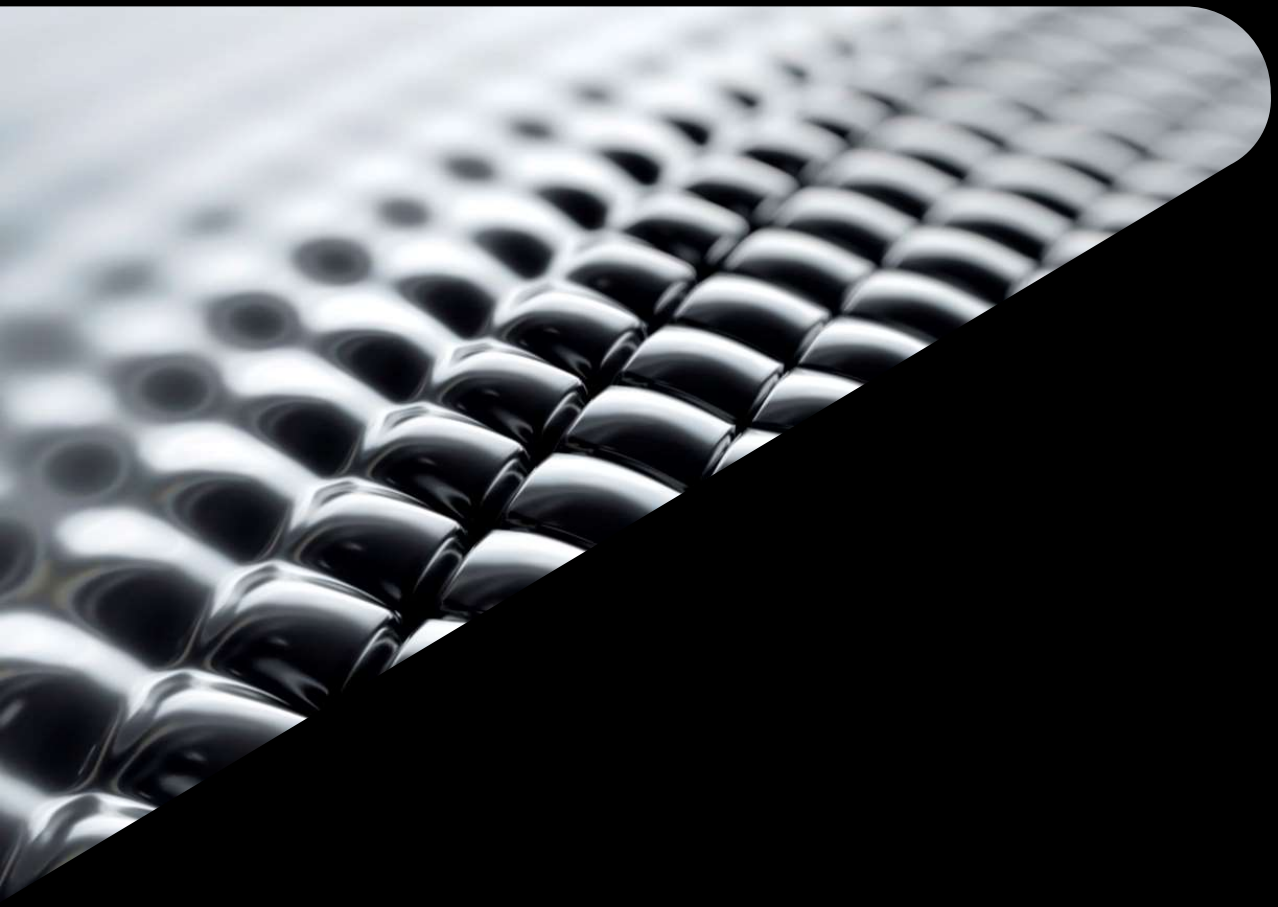




Parameters API

- Initial release
  - Read/write parameter definitions, tags
  - REST API
  - Later in the year (beta starting)
- More to follow
  - Filter and search
  - Revit API
- A lot of potential
  - Not limited to Revit
  - Integration with other software/services
  - Various standards collections





Sample Applications

Parameters API

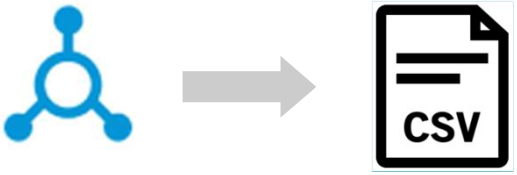
Export to csv

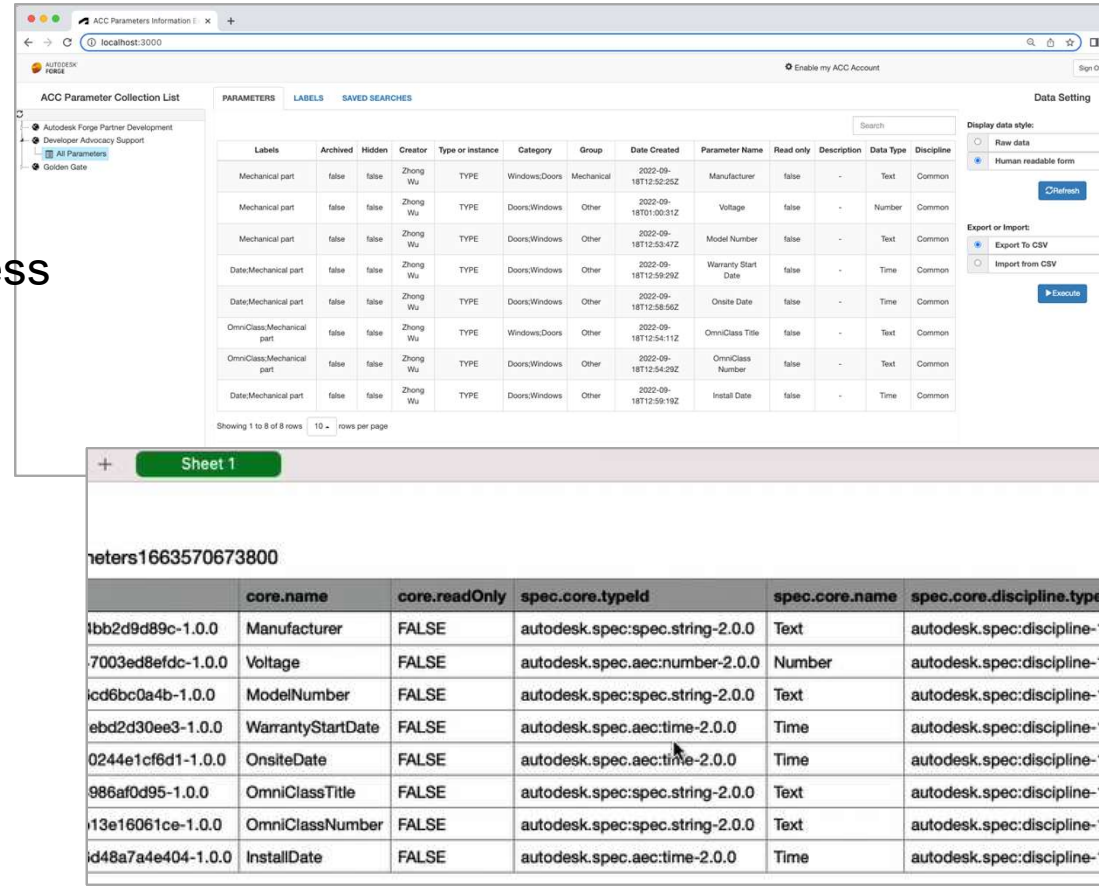
Parameters API

What it does: Export parameter definitions in ACC to a custom web page, then to csv. This sample is to demonstrate the basic read access with Parameters API. (Hope to have import working by the time of API release.)

Code: GitHub sample (TBD)

Author: Zhong Wu Liang, Autodesk





Chrome File Edit View History Bookmarks Profiles Tab Window Help Mon Sep 19 2:57 PM

Parameters - Autodesk Accou | ACC Parameters Information E

localhost:3000

AUTODESK FORGE

Enable my ACC Account

Sign Out Zhong Wu

ACC Parameter Collection List

- Autodesk Forge Partner Development
- Developer Advocacy Support
  - All Parameters
- Golden Gate

PARAMETERS LABELS SAVED SEARCHES

Search

Labels	Archived	Hidden	Creator	Type or instance	Category	Group	Date Created	Parameter Name	Read only	Description	Data Type	Discipline
Mechanical part	false	false	Zhong Wu	TYPE	Doors;Windows	Mechanical	2022-09-18T12:52:25Z	Manufacturer	false	-	Text	Common
Mechanical part	false	false	Zhong Wu	TYPE	Doors;Windows	Other	2022-09-18T01:00:31Z	Voltage	false	-	Number	Common
Mechanical part	false	false	Zhong Wu	TYPE	Doors;Windows	Other	2022-09-18T12:53:47Z	Model Number	false	-	Text	Common
Mechanical part;Date	false	false	Zhong Wu	TYPE	Doors;Windows	Other	2022-09-18T12:59:29Z	Warranty Start Date	false	-	Time	Common
Date;Mechanical part	false	false	Zhong Wu	TYPE	Doors;Windows	Other	2022-09-18T12:58:56Z	Onsite Date	false	-	Time	Common
Mechanical part;OmniClass	false	false	Zhong Wu	TYPE	Windows;Doors	Other	2022-09-18T12:54:11Z	OmniClass Title	false	-	Text	Common
OmniClass;Mechanical part	false	false	Zhong Wu	TYPE	Doors;Windows	Other	2022-09-18T12:54:29Z	OmniClass Number	false	-	Text	Common
Date;Mechanical part	false	false	Zhong Wu	TYPE	Doors;Windows	Other	2022-09-18T12:59:19Z	Install Date	false	-	Time	Common

Showing 1 to 8 of 8 rows 10 rows per page

Data Setting

Display data style:

- Raw data
- Human readable form

Refresh

Export or Import:

- Export To CSV
- Import from CSV

Execute

Parameter List

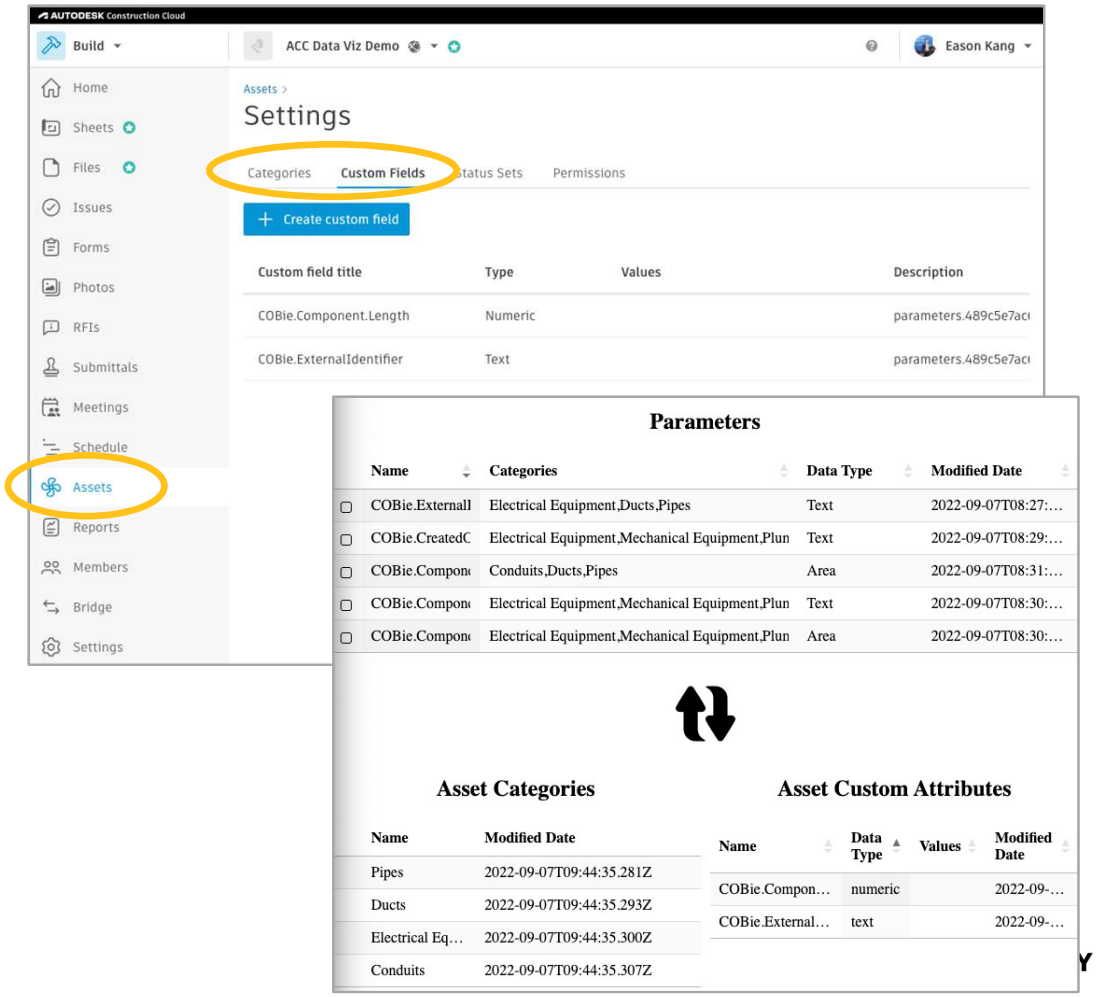
Integration: Import to Assets Custom Fields

Parameters API

What it does: Import selected parameter definitions into Assets module and set up categories and custom fields in settings.

Code: GitHub sample (TBD)

Author: Eason Kang, Autodesk



Microsoft Edge File Edit View History Favorites Tools Profiles Tab Window Help

Parameters - Autodesk Accou | Assets - Autodesk Build

https://acc.autodesk.com/build/assets/project...

AUTODESK Construction Cloud

Build | ACC Data Viz Demo | Eason Kang

- Home
- Sheets
- Files
- Issues
- Forms
- Photos
- RFIs
- Submittals
- Meetings
- Schedule
- Assets
- Reports
- Members
- Bridge
- Settings

Assets >

Settings

Categories | Custom Fields | Status Sets | Permissions

+ Create custom field

Custom field title	Type	Values
COBie.Component.Length	Numeric	
COBie.ExternalIdentifier	Text	

Wed Sep 7 5:45 PM

Autodesk Forge: Parameter M

localhost:3000

AUTODESK FORGE Parameter Manager

Logout (Eason Kang)

Parameters

Name	Categories	Data Type	Modified Date
COBie.Component./	Electrical Equipment,Mechanical Equipment,Plumbing Equi	Area	2022-09-07T08:30:46Z
COBie.Component.I	Electrical Equipment,Mechanical Equipment,Plumbing Equi	Text	2022-09-07T08:30:29Z
COBie.Component.I	Conduits,Ducts,Pipes	Area	2022-09-07T08:31:00Z
COBie.CreatedOn	Electrical Equipment,Mechanical Equipment,Plumbing Equi	Text	2022-09-07T08:29:02Z
COBie.ExternalIden	Electrical Equipment,Ducts,Pipes	Text	2022-09-07T08:27:03Z

Asset Categories

Name	Modified Date
Pipes	2022-09-07T09:44:35.281Z
Ducts	2022-09-07T09:44:35.293Z
Electrical Equip…	2022-09-07T09:44:35.300Z
Conduits	2022-09-07T09:44:35.307Z

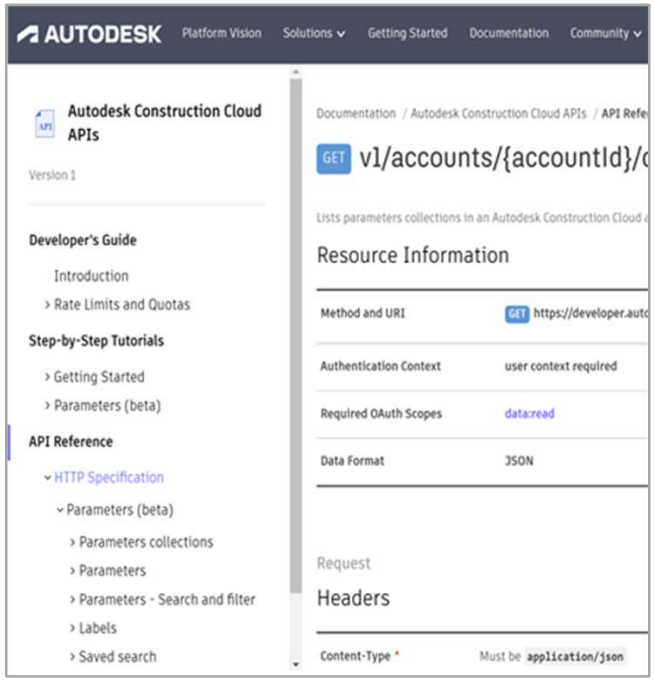
Asset Custom Attributes

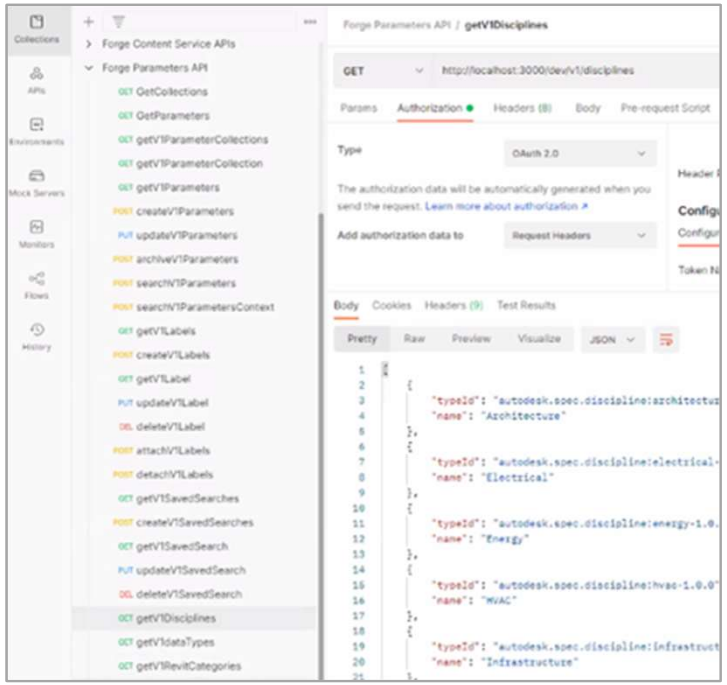
Name	Data Type	Values	Modified Date
COBie.Component.Length	numeric		2022-09-…
COBie.ExternalIdentifier	text		2022-09-…

Parameters API: Developer Resources

Work in progress …

- Documentation
- Code samples on GitHub
- Postman collection





Parameters API

Private Beta

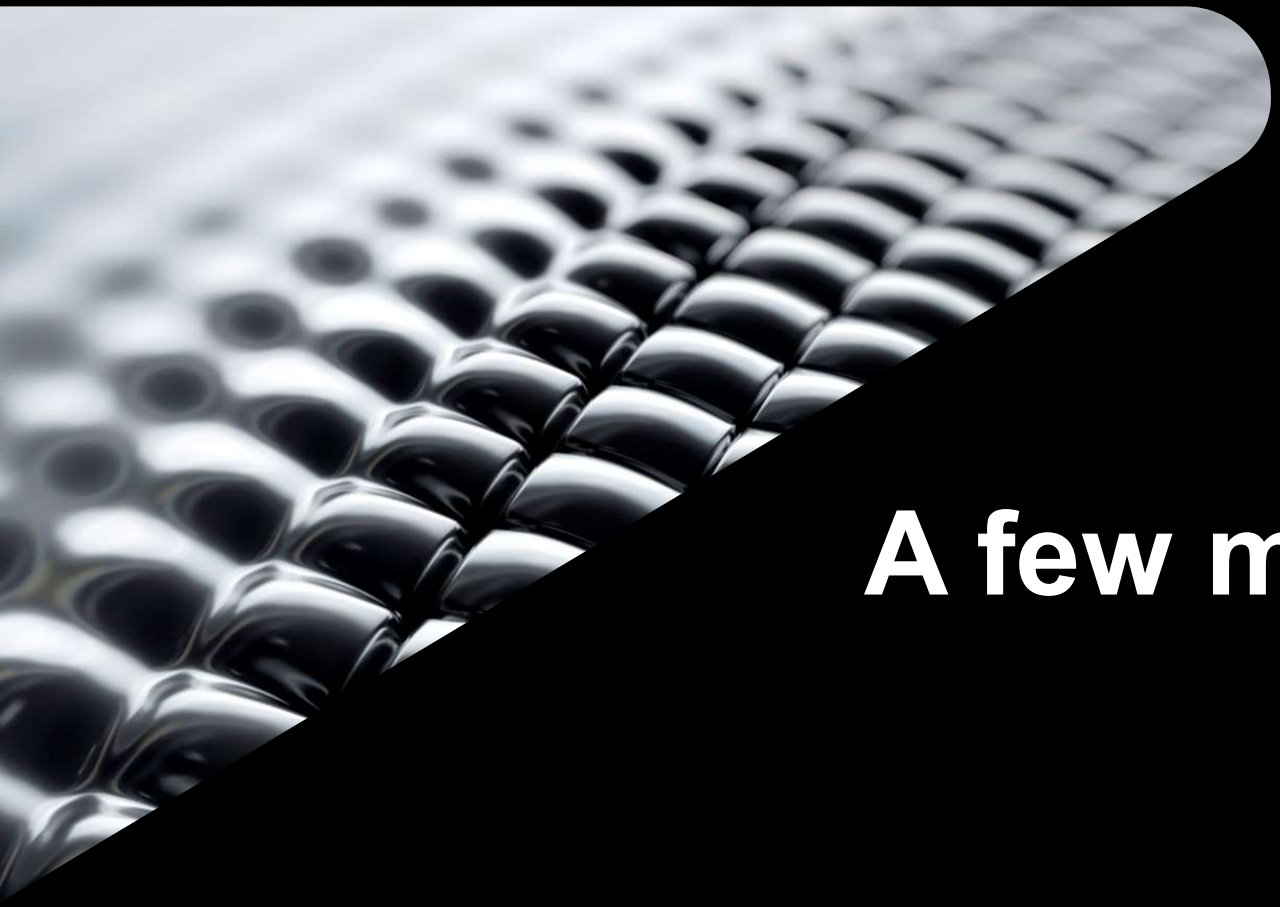
Call for participation

Sept. 26th ~ Nov. 4th, 2022

Interested?

Send an e-mail to: **parameters.api.beta@autodesk.com**

Title: **“Interested in Parameters API Beta Testing”**

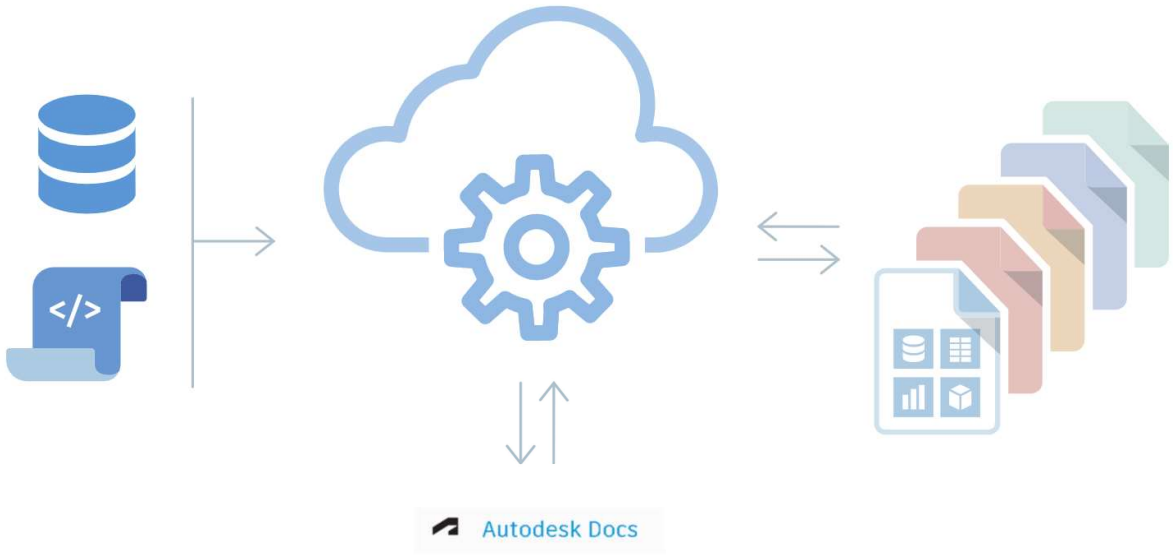


A few more things …

Messages from the factory

Design Automation API | Cloud Model Support

- Access is enabled **NOW**
- **Free** trial until **Oct 3rd**
- **Same** cost for jobs
- Revit **2022** and **2023** supported
- Functionality
  - File Open, Save, Read, Modification, Checkout, and SWC
  - Support for RCM cloud links



Heads up: Revit API ElementId will move to 64-bit

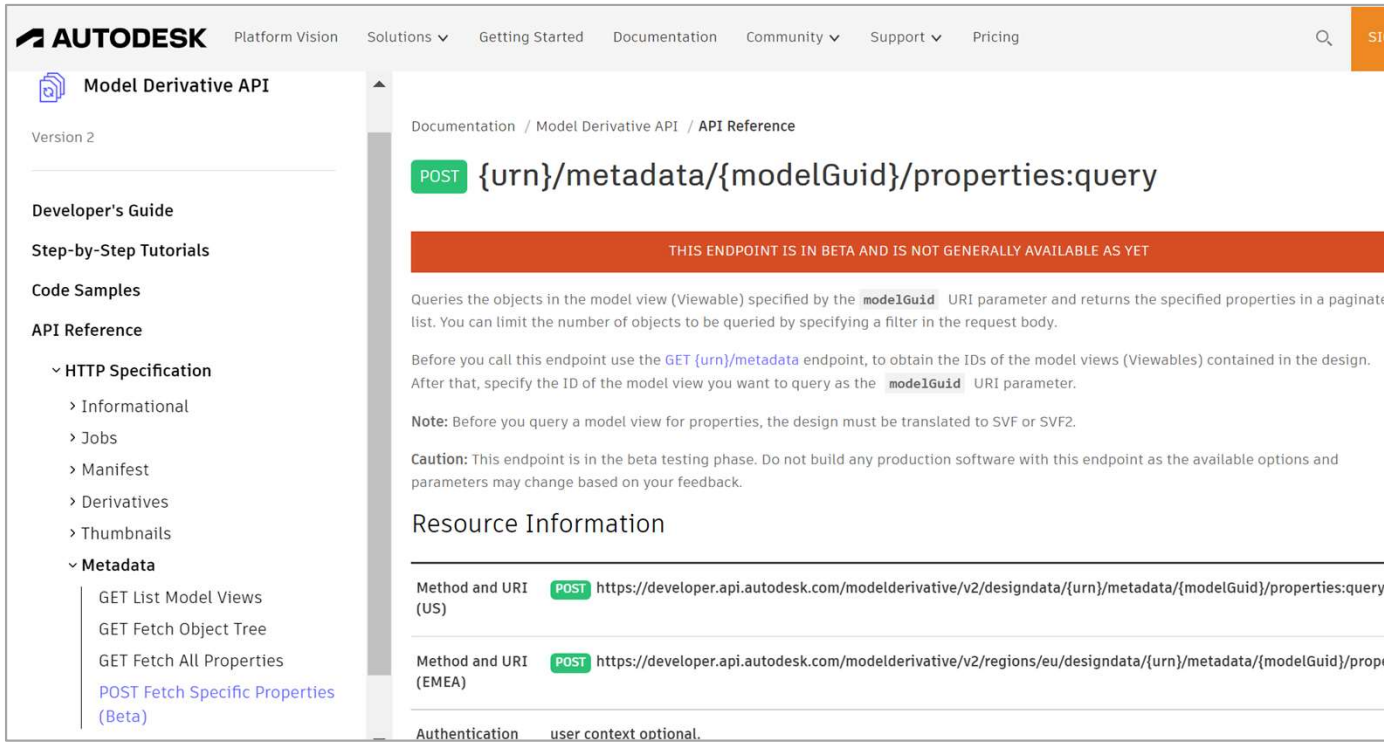
- Heads up to Revit API community: ElementId >> 64-bit
- Allow larger, more complex models
  - type long in C# or Int64 in .NET
  - Autodesk.Revit.DB.ExtensibleStorage.SchemaBuilder.AddSimpleField() and AddMapField() can take in 64-bit values for the fieldType, keyType, and valueType parameters.
  - Autodesk.Revit.DB.BuiltInCategory, Autodesk.Revit.DB.BuiltInParameter
  - More info. coming soom

Deprecated API	Replacement
Autodesk.Revit.DB.ElementId(Int32)	Autodesk.Revit.DB.ElementId(Int64)
Autodesk.Revit.DB.ElementId.IntegerValue	Autodesk.Revit.DB.ElementId.Value

Model Derivatives

Updates

- Versioning: previous/current/next
- Query specific properties



Blogs

- https://forge.autodesk.com/blog/model-derivative-extractor-version-now-available-revit
- https://forge.autodesk.com/blog/advanced-query-model-derivative-api

Methods to Access Design Data

Model Derivative	Model Properties	Design Automation	Data Exchange
Forge	ACC/BIM 360	Platform	ACC
File based	File based	File-based/RCM	Cloud hosted
Entire model	Entire model	Entire model	Partial model
Light weight query (beta)	Full query/filter	Control same as add-in	Limited
Read	Read	Read/write	Read
Today	Today	Today	Future

Summary

- Model Property API
- Data Exchange API (beta)
- Parameters Service API (in pipeline)

Acknowledgement

Don Whittle,
Distinguished Architect, Autodesk Construction Solutions

Alan Edwardes,
Software Architect, Autodesk Construction Solutions

Felix Beer,
Senior Principal Software Engineer, Autodesk Construction Solutions

Xiaodong Liang,
Principal Developer Advocate, Developer Advocacy

Zhong Wu,
Senior Developer Advocate, Developer Advocacy

Denis Grigor,
Senior Developer Advocate, Developer Advocacy

Introducing ***Autodesk Platform Services***
the next evolution of Forge

AUTODESK Forge → AUTODESK Platform Services

Learn more at **https://aps.autodesk.com**

Thank you!

@autodeskforge @mikako_harada

AUTODESK UNIVERSITY

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 and services offerings, and specifications and pricing at any time without notice, and is not responsible for typographical or graphical errors that may appear in this document.

© 2022 Autodesk. All rights reserved.

Autodesk Confidential & Proprietary Information - Please do not post, copy or distribute without authorization.