

Getting Started on Design Automation for Revit on Forge

Rahul Bhobe | Jason Kunkel

Autodesk | CADD Microsystems





About Rahul Bhobe

Principal Software Engineer

He holds a B. Tech degree in Naval Architecture from IIT Madras and a masters degree in Software Systems. He has been with Autodesk for 12 years working on several features of Revit. Currently he is working on developing Design Automation for Revit. In the past he has implemented Revit Conceptual Modeling features like Adaptive Components, Point Elements and Divided Surface. He has then worked on several Revit features like Family, Groups, Links and Worksharing.



About Jason Kunkel

CADD Microsystems

Jason has worked across the design and technology spectrum of the AEC industry for over 20 years first as an architectural designer for a major mid-Atlantic architecture firm specializing in large, public sector projects then migrating to the IT support world, where he worked to help architects and engineers leverage technology in new and exciting ways, and save time in the process. Currently at CADD Microsystems, Jason has been able to apply his knowledge and experience to help a wider range of customers achieve the same goals. He is one of the founders of RevitRVA, a Revit user group in the central Virginia area, and has a wide array of knowledge and experience with both software and hardware to help companies improve their processes and work more effectively.

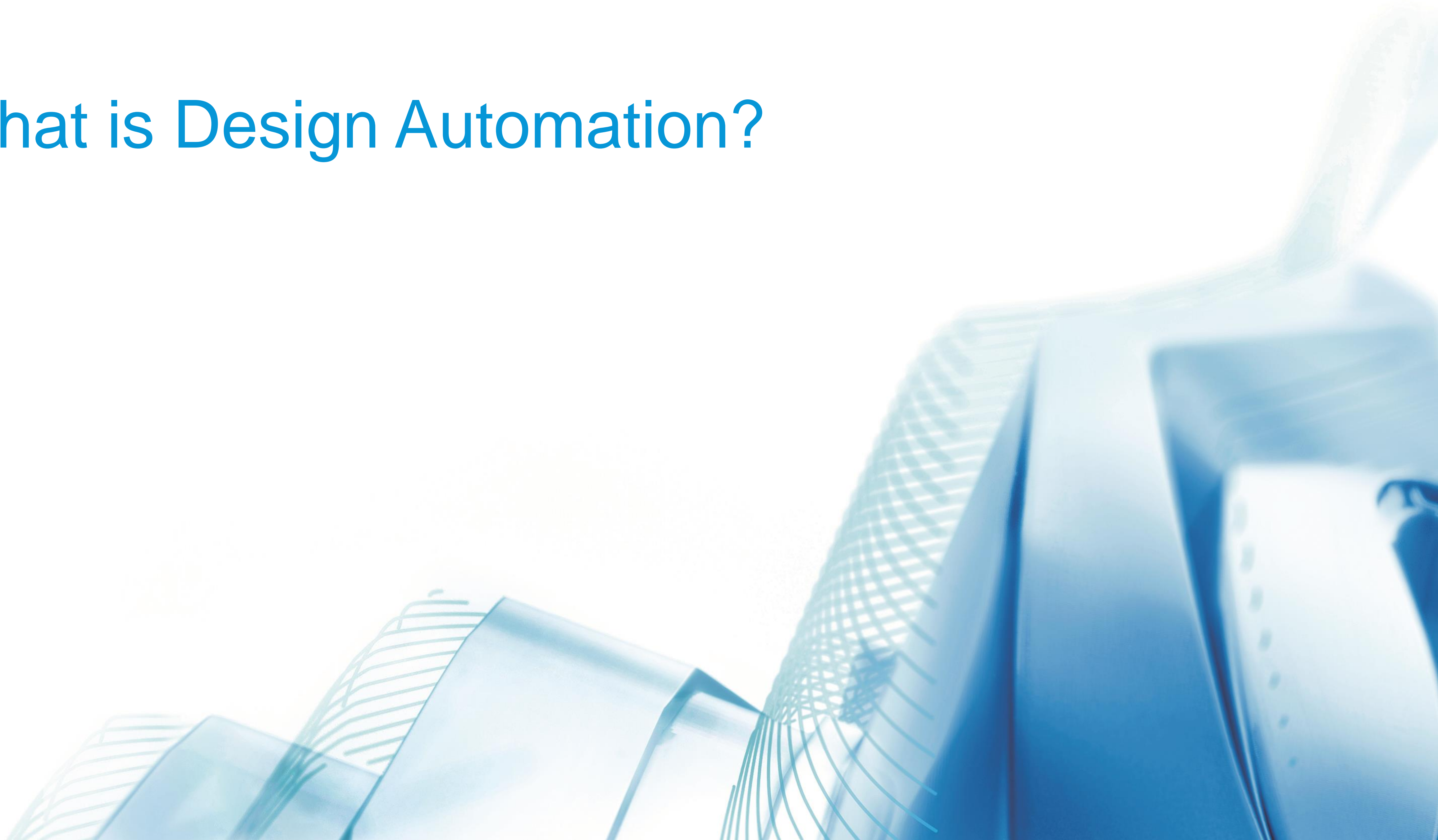
Ideal target audience

- Awesome Revit API developer that knows the power of Design Automation
- Awesome Revit user that knows the power of Revit API and Design Automation

Agenda

- Getting Started (how to learn Design Automation)
- A real customer's experience (Jason)
- Demos and common problems/solutions
- Tips and tricks we learnt in the last year

What is Design Automation?





Design Automation API V3

Run automations on your design files



AUTODESK® REVIT®



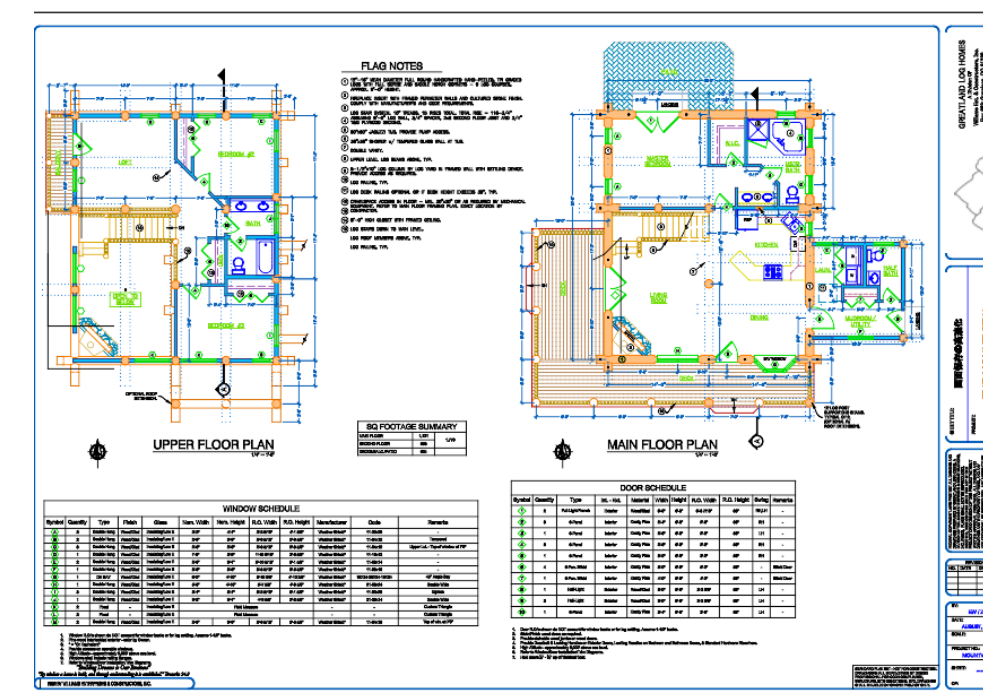
AUTODESK® INVENTOR®

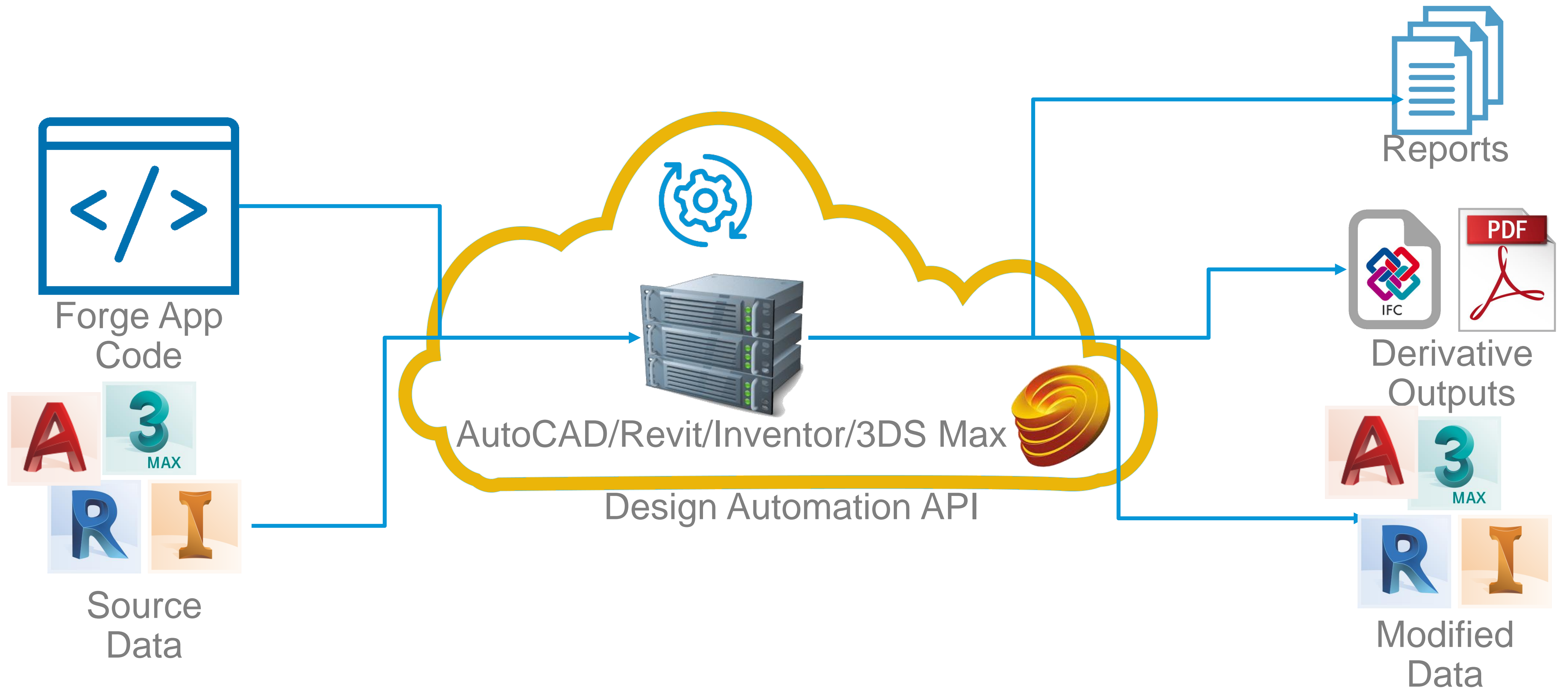


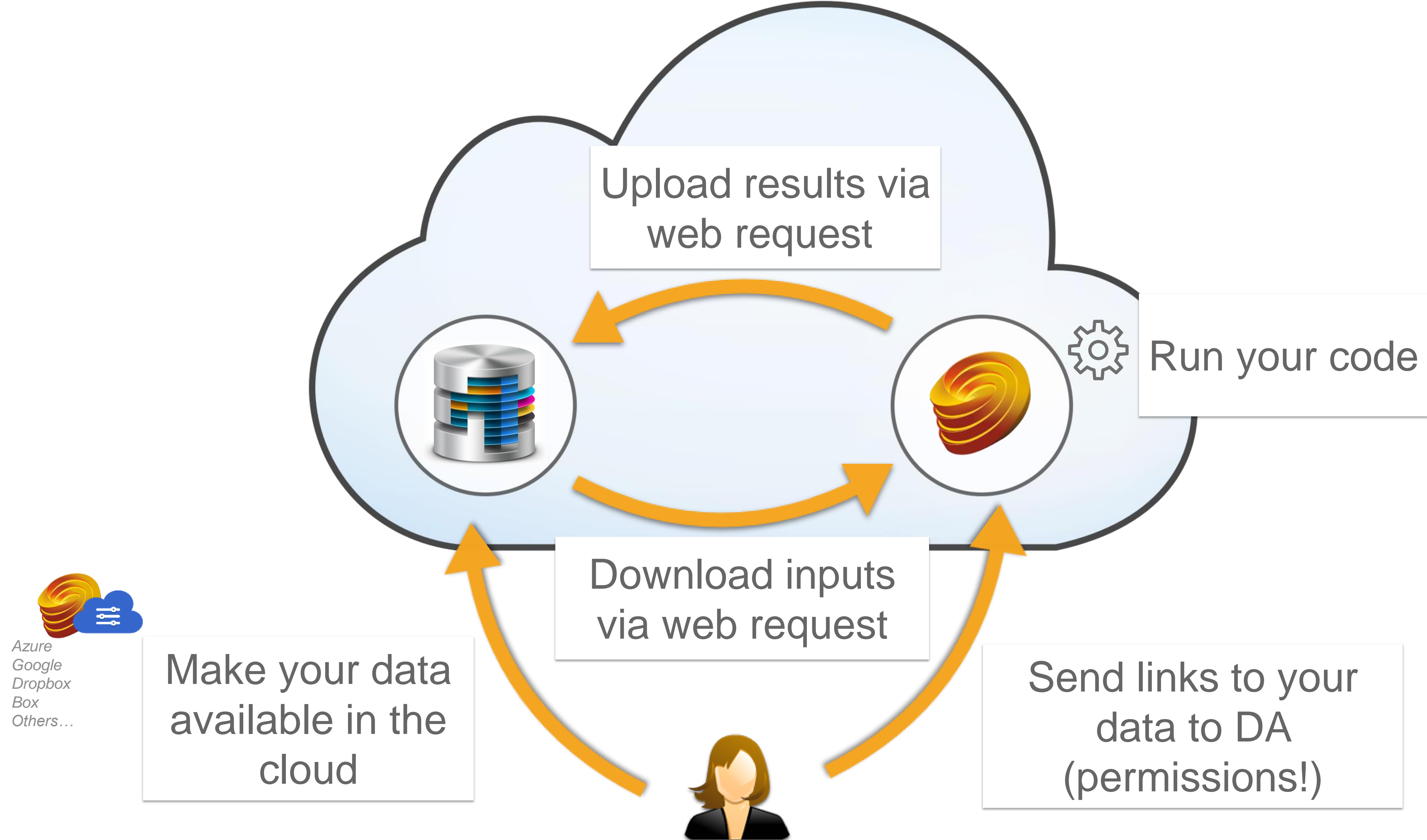
AUTODESK® 3DS MAX®



AUTODESK® AUTOCAD®







❗ Input/outputs are **NOT** stored, they are downloaded/processed and **DISCARDED**

The API

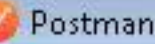
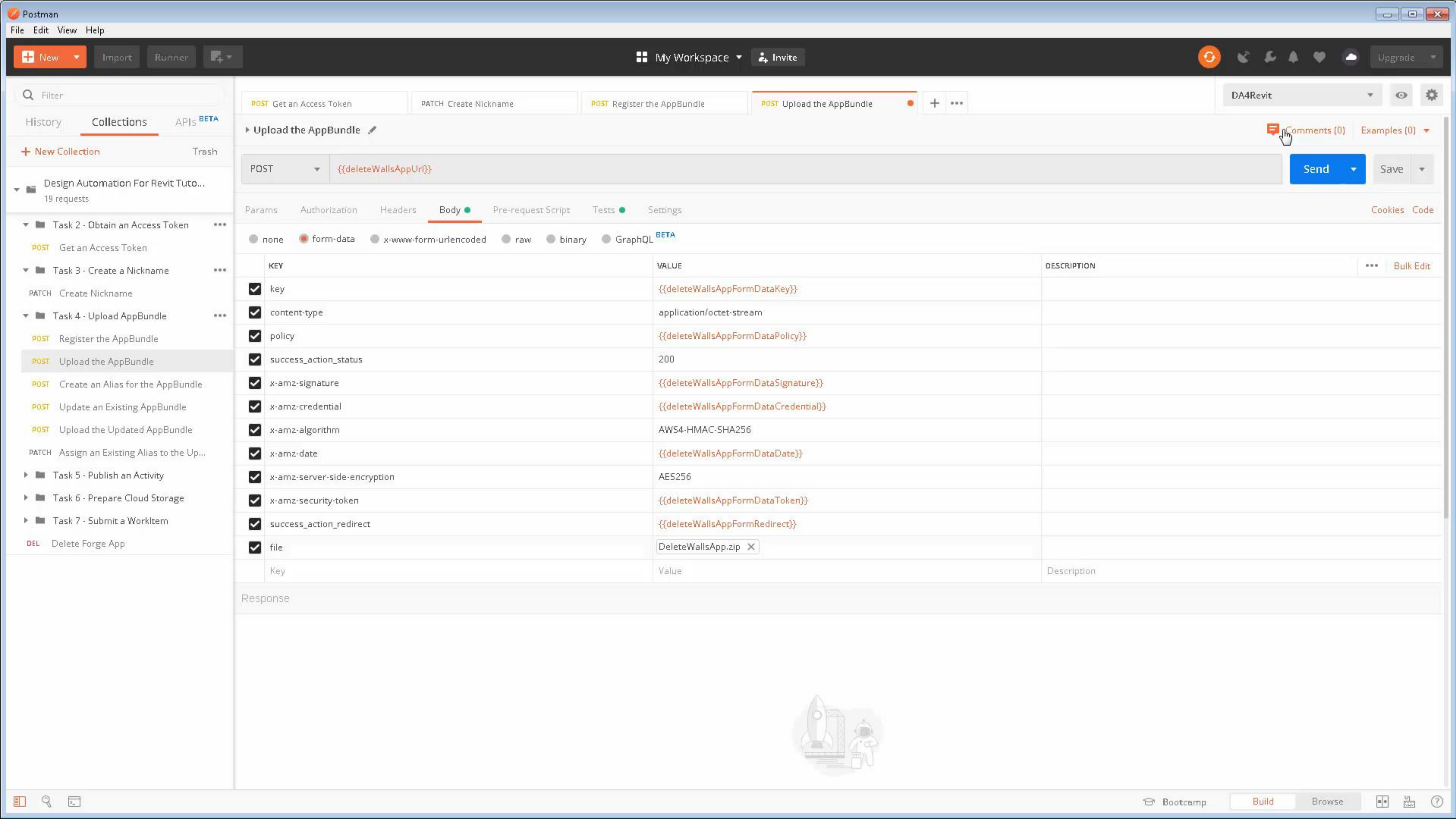
| V3 HTTP Endpoint | Programming concept | Product concept |
|------------------|---------------------|------------------------------------|
| /workitems | Function call | Product execution, job |
| /activities | Function definition | Input/Output types, parameters |
| /appbundles | Shared library | Addin |
| /engines | Instruction set | Product (revit, inventor..) to use |

Documentation



Postman Collections





Postman

File Edit View Help

New

Import

Runner

My Workspace Invite

Refresh

Share

Sync

Heart

Cloud

Upgrade

Filter

History Collections APIs BETA

+ New Collection Trash

Design Automation For Revit Tuto...
19 requests

Task 2 - Obtain an Access Token

Task 3 - Create a Nickname

Task 4 - Upload AppBundle

POST Get an Access Token

POST Create an Alias for the AppBundle

POST Update an Existing AppBundle

POST Upload the Updated AppBundle

PATCH Assign an Existing Alias to the Up...

Task 5 - Publish an Activity

Task 6 - Prepare Cloud Storage

Task 7 - Submit a WorkItem

DEL Delete Forge App

POST Get an Access Token

PATCH Create Nickname

POST Register the AppBundle

POST Upload the AppBundle

+

...

DA4Revit

Upload the AppBundle

Comments (0) Examples (0)

POST

{{deleteWallsAppUri}}

Send

Save

Params Authorization Headers Body Pre-request Script Tests Settings Cookies Code

none form-data x-www-form-urlencoded raw binary GraphQL BETA

| KEY | VALUE | DESCRIPTION | | Bulk Edit |
|--|--------------------------------------|-------------|--|-----------|
| <input checked="" type="checkbox"/> key | {{deleteWallsAppFormDataKey}} | | | |
| <input checked="" type="checkbox"/> content-type | application/octet-stream | | | |
| <input checked="" type="checkbox"/> policy | {{deleteWallsAppFormDataPolicy}} | | | |
| <input checked="" type="checkbox"/> success_action_status | 200 | | | |
| <input checked="" type="checkbox"/> x-amz-signature | {{deleteWallsAppFormDataSignature}} | | | |
| <input checked="" type="checkbox"/> x-amz-credential | {{deleteWallsAppFormDataCredential}} | | | |
| <input checked="" type="checkbox"/> x-amz-algorithm | AWS4-HMAC-SHA256 | | | |
| <input checked="" type="checkbox"/> x-amz-date | {{deleteWallsAppFormDataDate}} | | | |
| <input checked="" type="checkbox"/> x-amz-server-side-encryption | AES256 | | | |
| <input checked="" type="checkbox"/> x-amz-security-token | {{deleteWallsAppFormDataToken}} | | | |
| <input checked="" type="checkbox"/> success_action_redirect | {{deleteWallsAppFormRedirect}} | | | |
| <input checked="" type="checkbox"/> file | DeleteWallsApp.zip X | | | |
| Key | Value | Description | | |

Response



Postman Recap

Why is this a good start?

- Starts from scratch
- Takes less than 10 mins
- Simple appbundle (addin)
- Simple input/outputs
- You end with successful Design Automation job

Decoding Activity



Decoding Activity (command line)

```
{
  "id": "DeleteWallsActivity",
  "commandline": [ "$(engine.path)\\revitcoreconsole.exe /i $(args[rvtfile].path) /al $(appbundles[DeleteWallsBundle].path)" ],
  "parameters": {
    "rvtfile": {
      "zip": false,
      "ondemand": false,
      "verb": "get",
      "description": "input revit model",
      "required": true
    },
  },
  "result": {
    "zip": false,
    "ondemand": false,
    "verb": "put",
    "description": "results",
    "required": true,
    "localName": "result.rvt"
  },
},
"engine": "Autodesk.Revit+2018",
"appbundles": [ "MyNickname.DeleteWallsBundle+myalias" ],
"description": "Delete walls from Revit file."
}
```

/al \$(appbundles[DeleteWallsBundle].path)

\$(engine.path)\\revitcoreconsole.exe

/i \$(args[rvtfile].path)

["MyNickname.DeleteWallsBundle+myalias"]

Decoding Activity (parameters)

```
{
  "id": "DeleteWallsActivity",
  "commandline": [ "$(engine.path)\\revitcoreconsole.exe /i $(args[rvtFile].path) /al $(appbundles[DeleteWallsBundle].path)" ],
  "parameters": {
    "rvtfile": {
      "zip": false,
      "ondemand": false,
      "verb": "get",
      "description": "input revit model",
      "required": true
    },
    "result": {
      "zip": false,
      "ondemand": false,
      "verb": "put",
      "description": "results",
      "required": true,
      "localName": "result.rvt"
    }
  },
  "engine": "Autodesk.Revit+2018",
  "appbundles": [ "MyNickname.DeleteWallsBundle+myalias" ],
  "description": "Delete walls from Revit file."
}
```

rvtfile

"verb": "get"

"required": true

result

"verb": "put"

"localName": "result.rvt"

"required": true

A real customer's experience



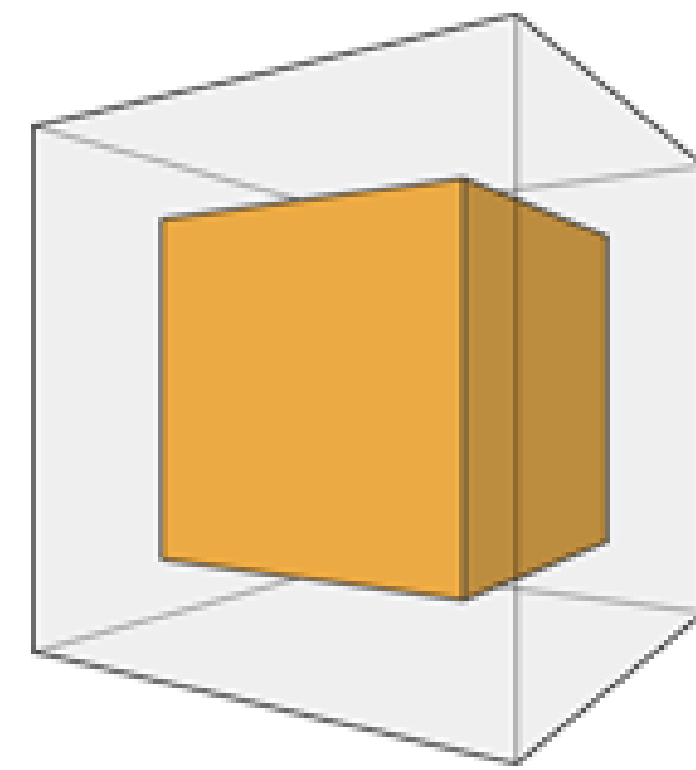
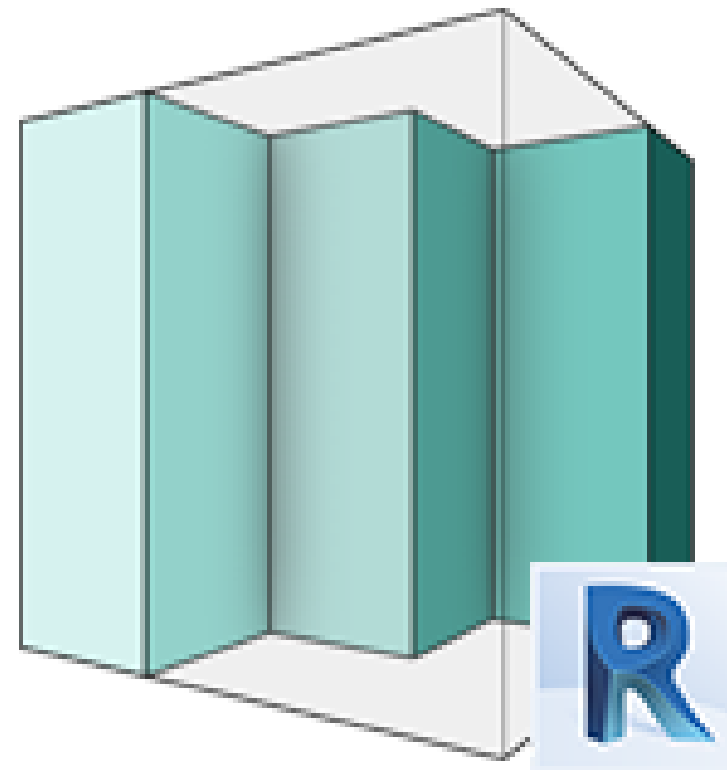
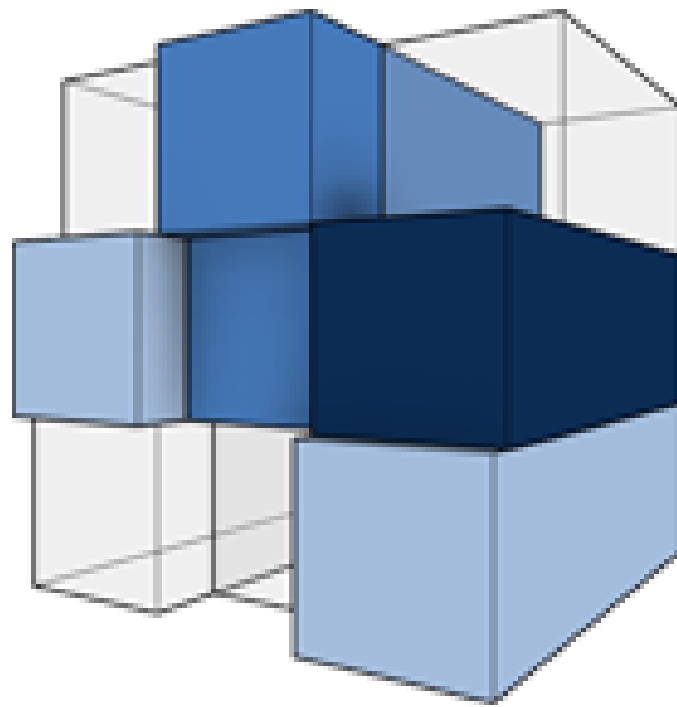
Get us all on the same page

- Discuss existing Revit add-in transition to DA4R
- I am a high-level “copy/paster”
- Member of development team for BIM Interoperability Tools



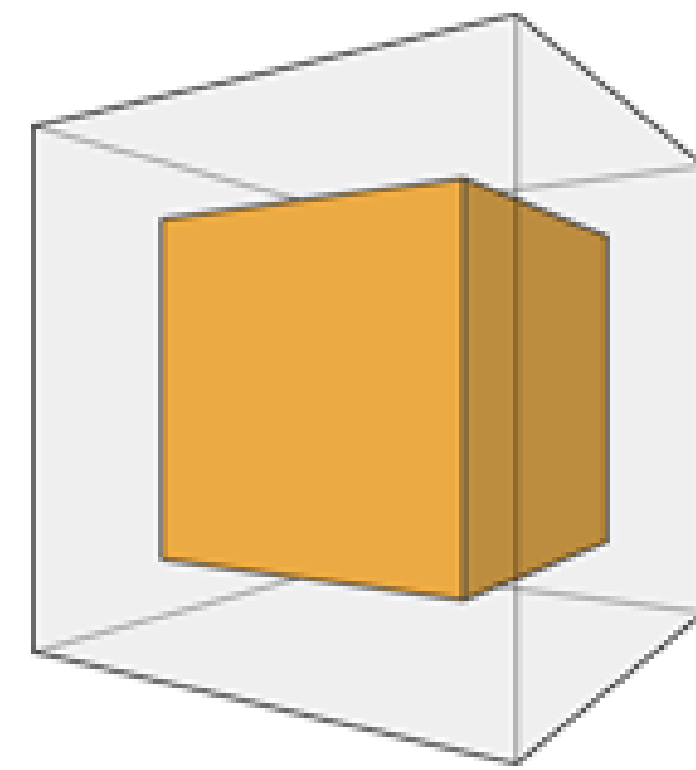
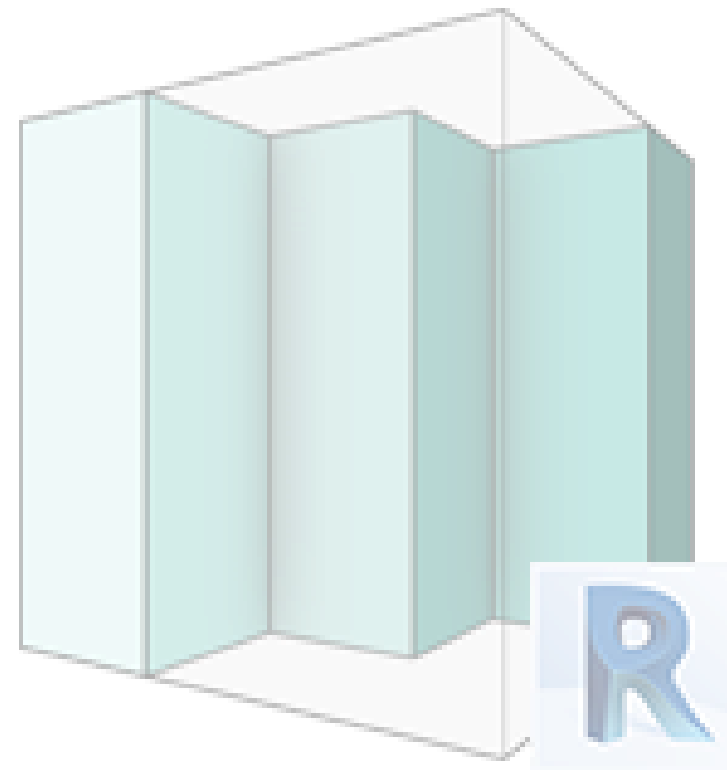
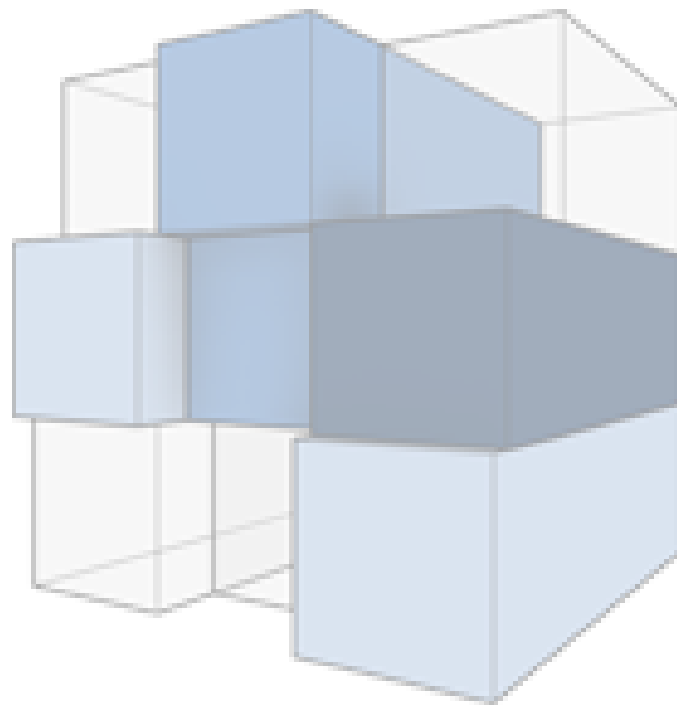
BIM Interoperability Tools

- Classification Manager
- COBie Extension
- Model Checker



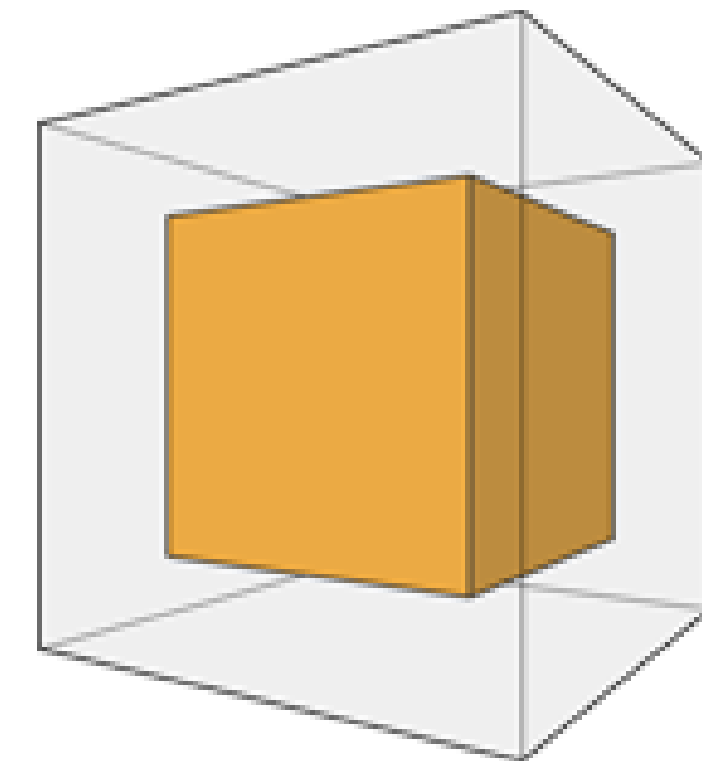
BIM Interoperability Tools

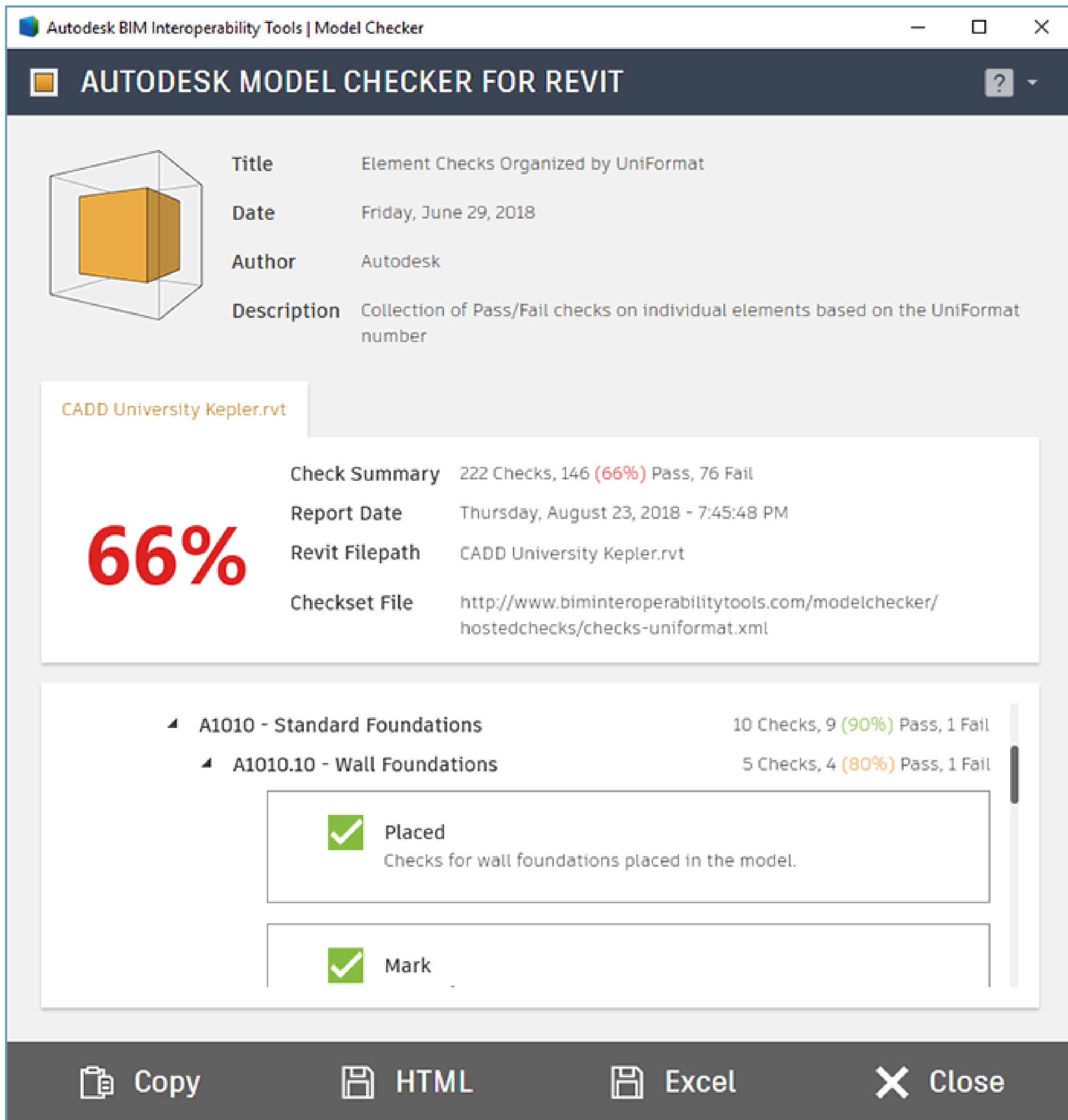
- Classification Manager
- COBie Extension
- Model Checker



Model Checker on Forge

- Why?
 - Huge opportunity to expand use of the Model Checker to non-Revit users
 - Had the least amount of impact on the Revit model
 - Clean code
 - Workflow was very 1-2-3 vs the other tools





Desktop version workflow

1. Open model
2. Start the add-in
 1. Identify already created XML “checkset” with check rules
 2. Turn “on” and “off” individual checks
 3. Save configuration to the model
3. Start another part of the add-in
 1. Identify models to run checks against
 2. Run the check(s)
 3. Review the report which is saved to the model
 4. Export the report as HTML or Excel

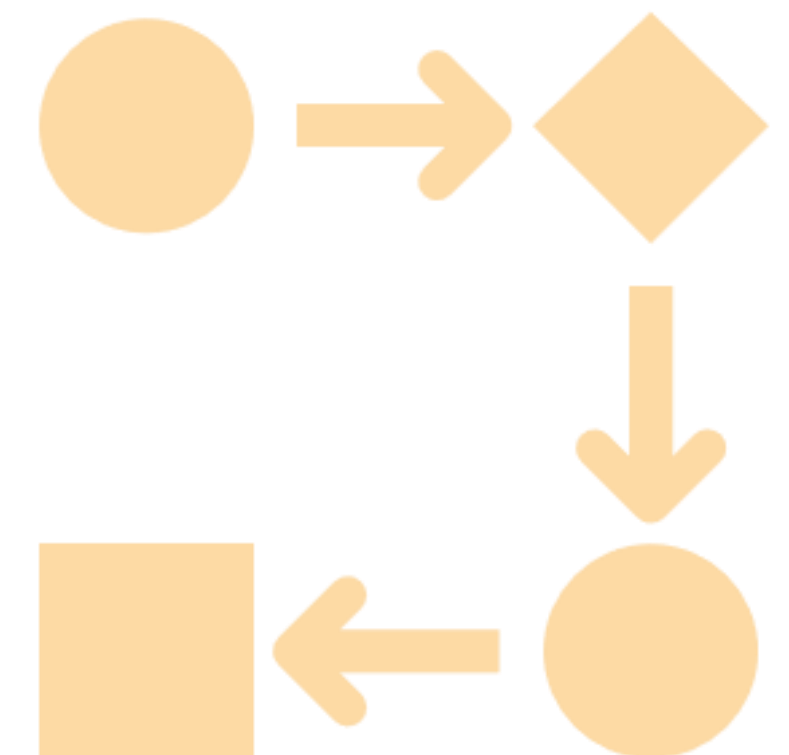
Process

- Education
 - Lots of reading
 - Lots of small tests
 - Forge Accelerator in Boston (Super Bowl week)
 - Forge DevCon



Process

- Re-design
 - Breaking down the original into parts
 - Realign them with new understanding and rules
 - Remove the UI
 - Offload processing outside of Forge
 - Identify inputs and outputs through a new interface
 - Optimize for web
 - Simplify for initial beta



Desktop version workflow

1. Open model
 2. Start the add-in
- Move to later**

- Web form**
1. Identify already created XML “checkset” with check rules

Simplification and lack of functionality

- ~~2. Turn “on” and “off” individual checks~~
 - ~~3. Save configuration to the model~~
- Simplification**

3. Start another part of the add-in
- Move to later**

- Web form**
1. Identify models to run checks against
 2. Run the check(s)
- Forge**

- Web**
3. Review the report which is saved to the model
 4. Export the report as HTML or Excel

Design Automation on Forge workflow

WEB INTERFACE

1. Login to BIM 360 (new functionality)
2. Identify BIM 360 hosted model
3. Identify XML checkset – via upload or “Public Library”

FORGE

4. Location of model and XML are passed off
5. Model is loaded and the check is run
6. JSON results are generated and returned

WEB INTERFACE

7. JSON read through webpage or exported

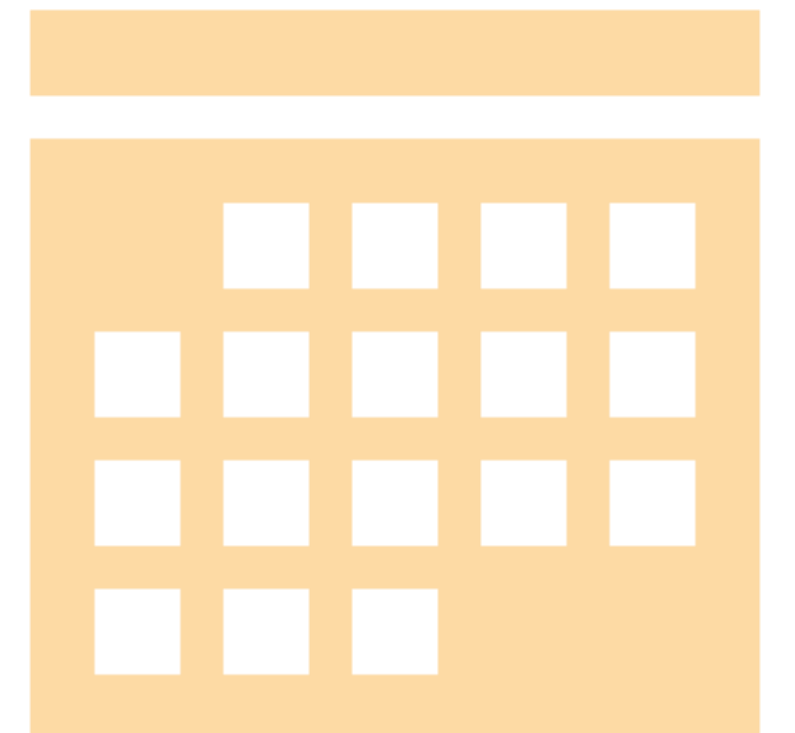
Execute

- Like any other application
- Develop, test, break it, develop some more



Where it stands now

- Have not yet rolled out
- All the parts work
- Internal testing
- Next steps are to have a closed beta





STEP 1 | CHOOSE YOUR MODEL !



BIM360

Recent Models

Test Company > Test Project 0 ▼

| Name | Description | Last Update | Update By | Version |
|---|----------------------|---------------------|------------------|---------|
| Sample Model 0 - Test Project 0 - Level 0 | Sample Description 0 | 11/11/2019 10:22 PM | Sample Updater 0 | V0 |
| Sample Model 1 - Test Project 0 - Level 0 | Sample Description 1 | 11/11/2019 10:22 PM | Sample Updater 1 | V1 |
| Sample Model 2 - Test Project 0 - Level 0 | Sample Description 2 | 11/11/2019 10:22 PM | Sample Updater 2 | V2 |
| Sample Model 3 - Test Project 0 - Level 0 | Sample Description 3 | 11/11/2019 10:22 PM | Sample Updater 3 | V3 |
| Sample Model 4 - Test Project 0 - Level 0 | Sample Description 4 | 11/11/2019 10:22 PM | Sample Updater 4 | V4 |

STEP 2 | CHOOSE YOUR CHECKSET !

Publicly Available

Local



sample title 0



sample title 1



sample title 2



sample title 3



sample title 4

STEP 2 | CHOOSE YOUR REPORT

✓

| Report Date | Check Set | Summary | |
|---------------------|--|--|---|
| 11/11/2019 10:27 PM | sample checklist 4 for model Sample Model 0 - Test Project 0 - Level 0 | 17 Checks, 7 (61%) Pass, 3 Fail, 5 Count/List, 2 Not Run | ▲ |
| 11/11/2019 10:27 PM | sample checklist 5 for model Sample Model 0 - Test Project 0 - Level 0 | 17 Checks, 7 (61%) Pass, 3 Fail, 5 Count/List, 2 Not Run | |
| 11/11/2019 10:27 PM | sample checklist 6 for model Sample Model 0 - Test Project 0 - Level 0 | 17 Checks, 7 (61%) Pass, 3 Fail, 5 Count/List, 2 Not Run | |
| 11/11/2019 10:27 PM | sample checklist 7 for model Sample Model 0 - Test Project 0 - Level 0 | 17 Checks, 7 (61%) Pass, 3 Fail, 5 Count/List, 2 Not Run | |
| 11/11/2019 10:27 PM | sample checklist 8 for model Sample Model 0 - Test Project 0 - Level 0 | 17 Checks, 7 (61%) Pass, 3 Fail, 5 Count/List, 2 Not Run | |
| 11/11/2019 10:27 PM | sample checklist 9 for model Sample Model 0 - Test Project 0 - Level 0 | 17 Checks, 7 (61%) Pass, 3 Fail, 5 Count/List, 2 Not Run | ▼ |

STEP 3 | VIEW AND EXPORT YOUR RESULTS

!

Export to Excel

Export to HTML

61%

Check Summary

Report Date

Revit Filepath

Checkset File

17 Checks, 7 (61%) Pass, 3 Fail, 5 Count/List, 2 Not Run

Mon Nov 11 2019 22:27:32 GMT-0500 (Eastern Standard Time)

C:\some\model\path\model 4

C:\some\checkset\path\checkset 4

▶

Heading 0 - sample checklist 4 for model Sample Model 0 - Test Project 0 - Level 017 Checks, 7 (75%) Pass, 3 Fail, 5 Count/List, 2 Not Run

▶

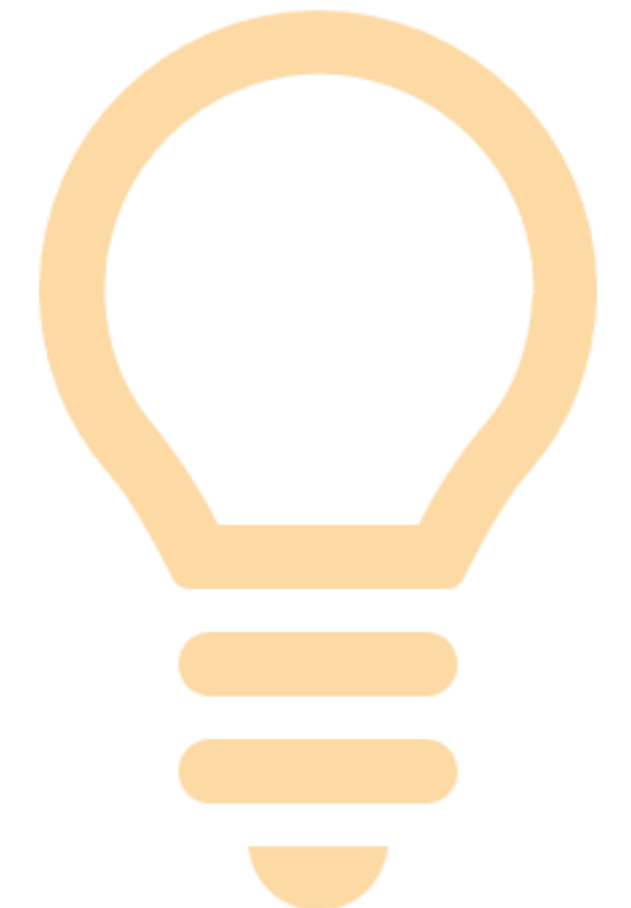
Heading 1 - sample checklist 4 for model Sample Model 0 - Test Project 0 - Level 017 Checks, 7 (75%) Pass, 3 Fail, 5 Count/List, 2 Not Run

▶

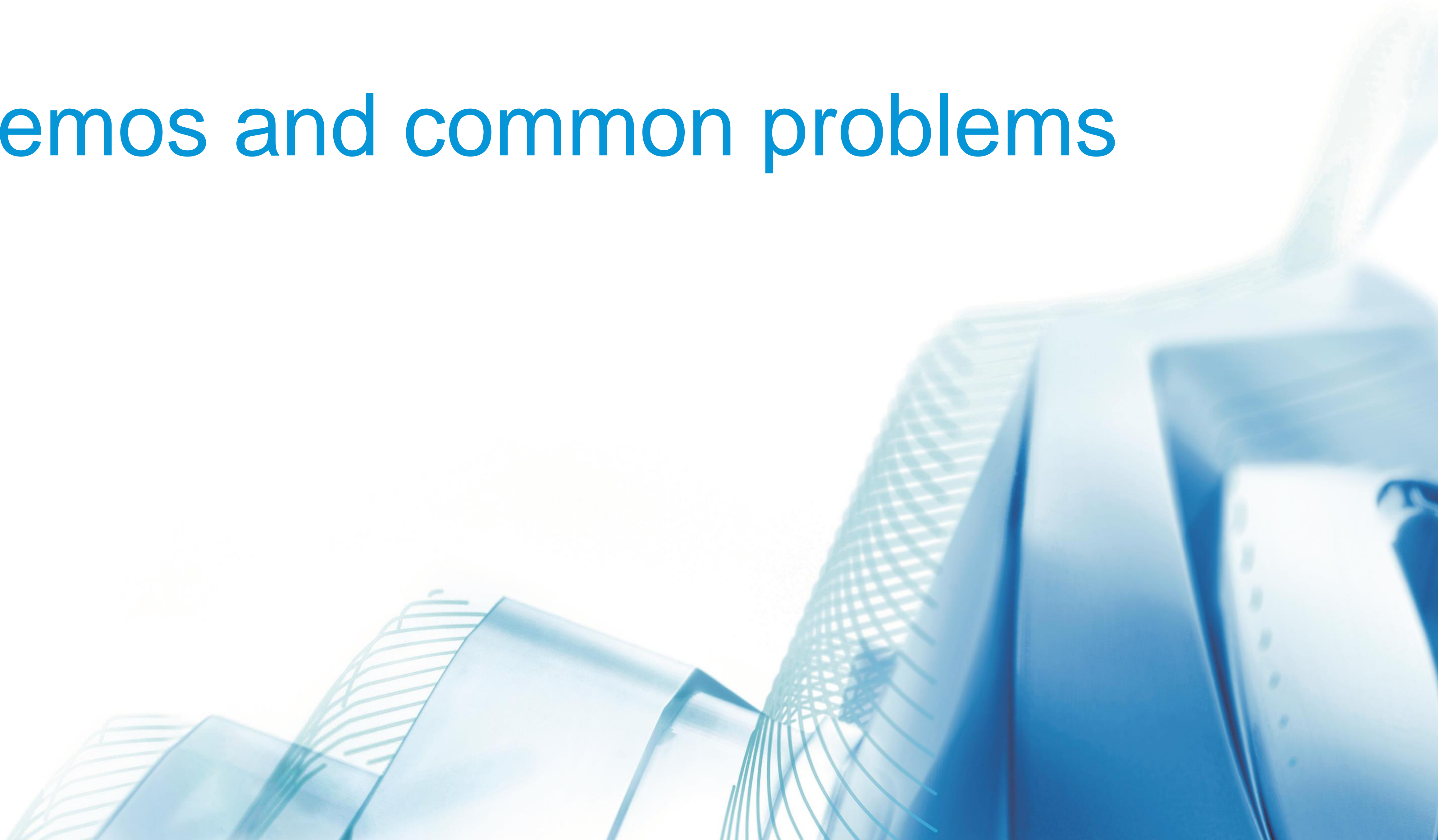
Heading 2 - sample checklist 4 for model Sample Model 0 - Test Project 0 - Level 017 Checks, 7 (75%) Pass, 3 Fail, 5 Count/List, 2 Not Run

Tips

- As always, anticipate change
- Make sure you are looking at the right help file
- Pull in another resource for all the new stuff



Demos and common problems



Content Management & Creation

Who

Atkins

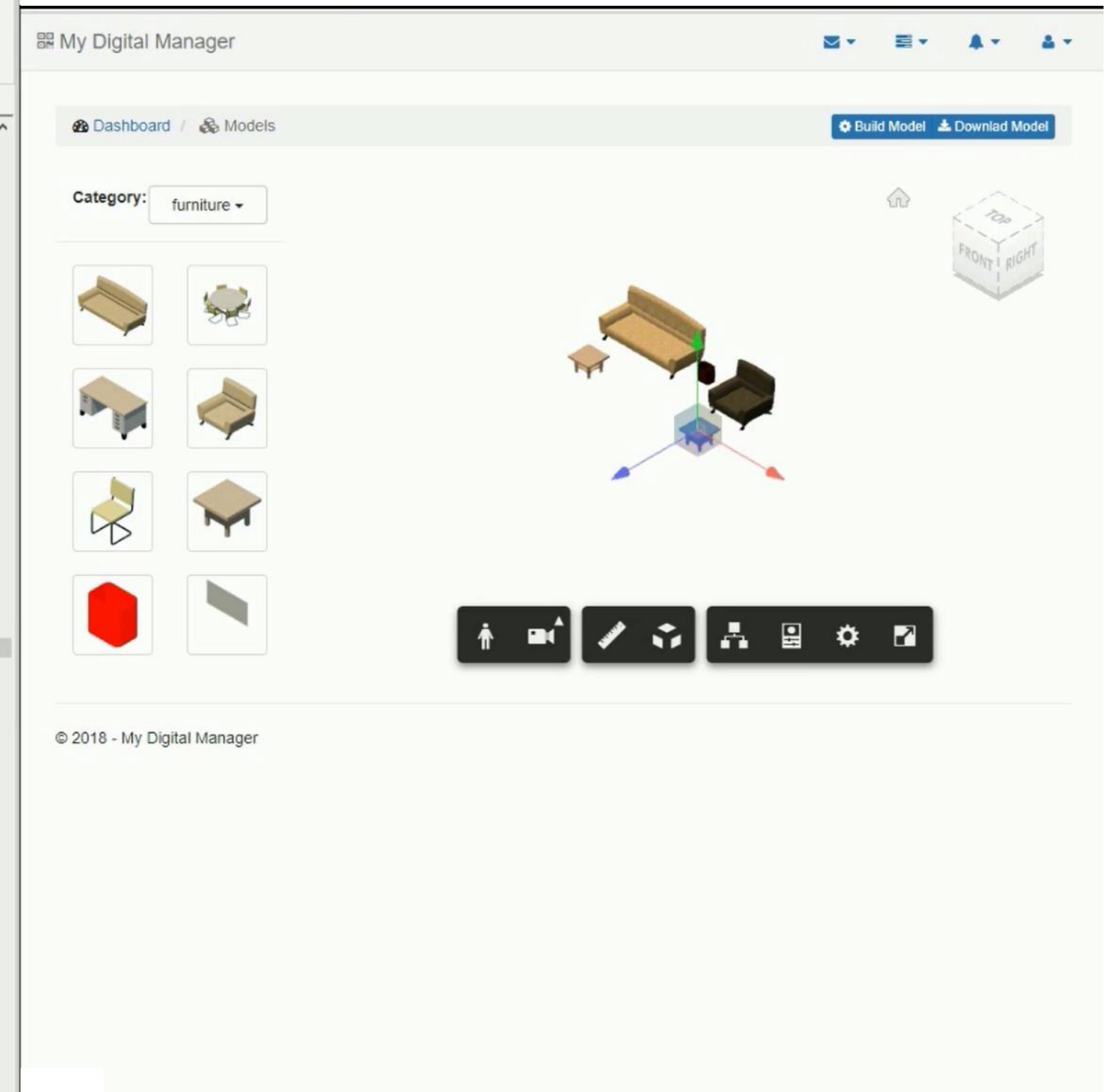
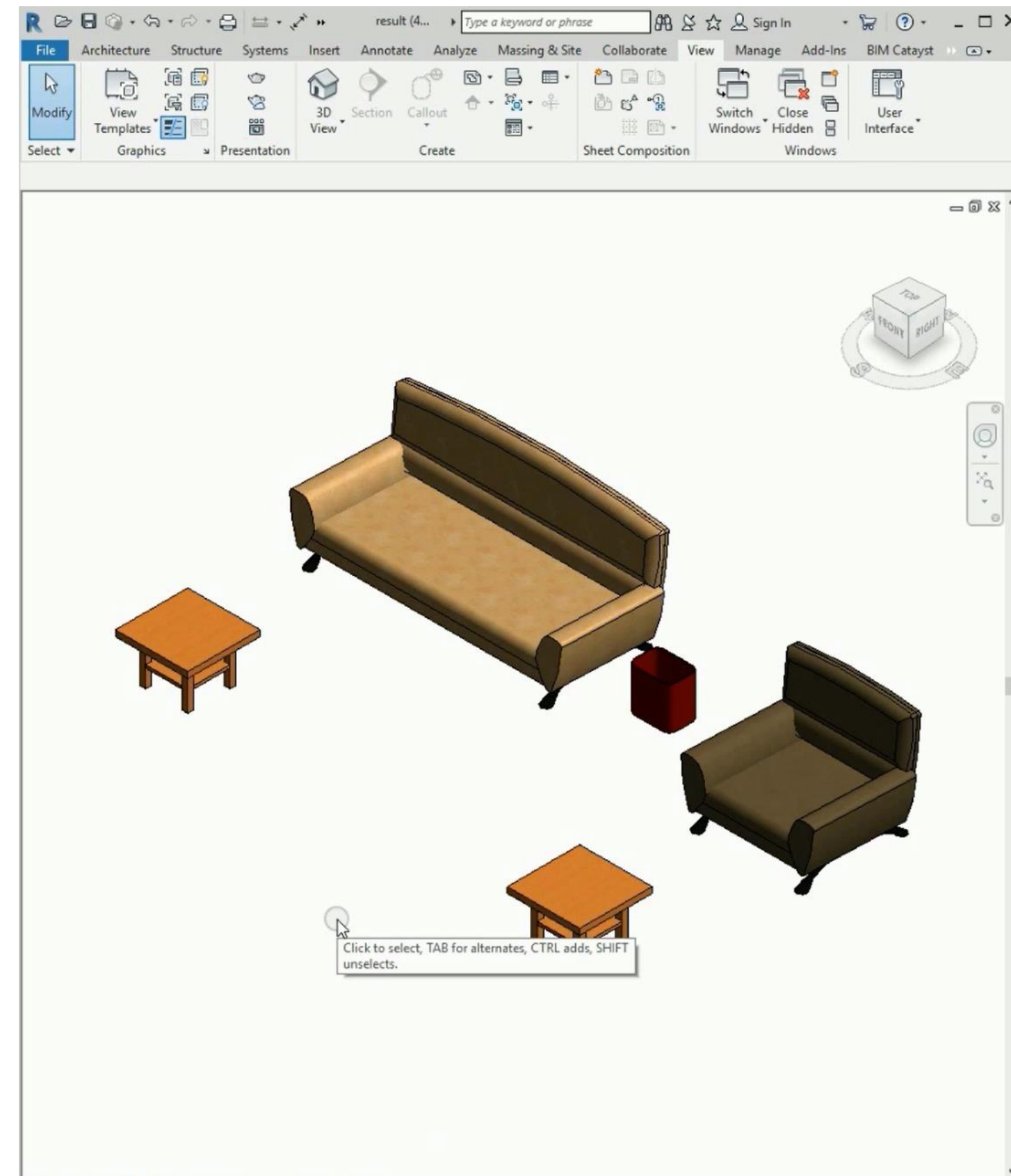
Mustafa Salaheldin Ali

Solution

Revit families are stored and organized by category in a digital manager. Using the Forge Viewer the families can be selected, inserted and manipulated directly in the browser.

Once the families have been positioned a combined template model can be generated using Design Automation for Revit. This model matches the geometry organized in the Forge viewer, and contains individual families for all content.

ATKINS



APIs Used:



Design
Automation



Viewer



Model
Derivative

result (4...) Type a keyword or phrase Sign In

File Architecture Structure Systems Insert Annotate Analyze Massing & Site Collaborate View Manage Add-Ins BIM Catalyst

Modify View Templates 3D View Section Callout Sheet Composition Switch Windows Close Hidden User Interface

Select Graphics Presentation Create Windows



My Digital Manager

mustafasalah-001-site3.htempurl.com/Home/Models

My Digital Manager

Build Model Download Model

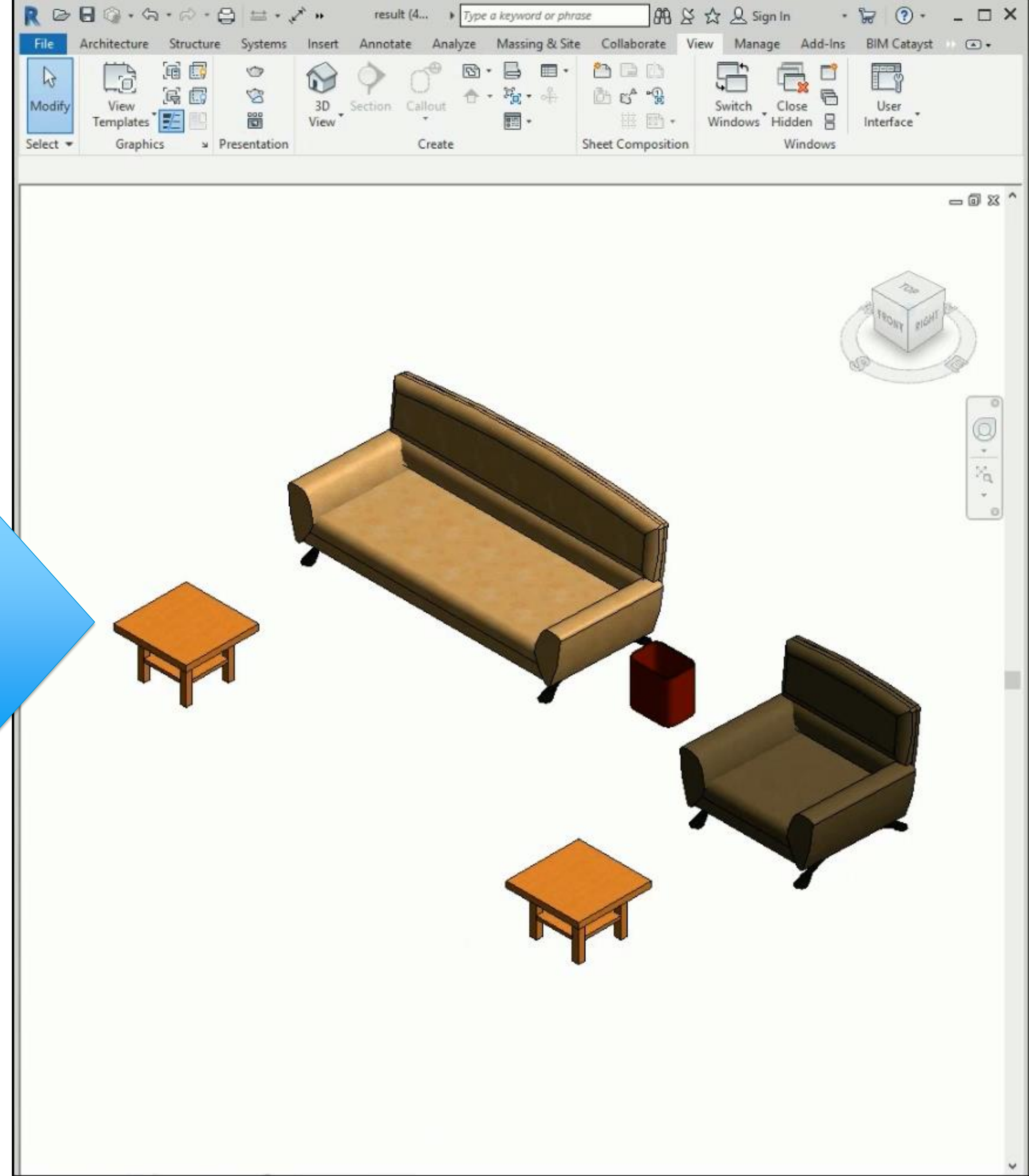
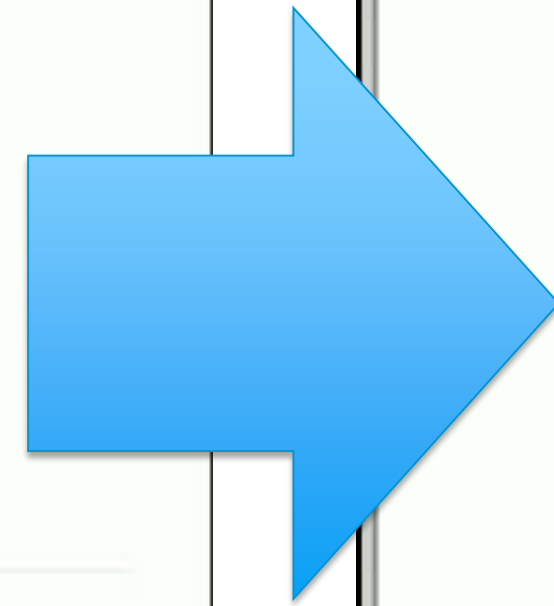
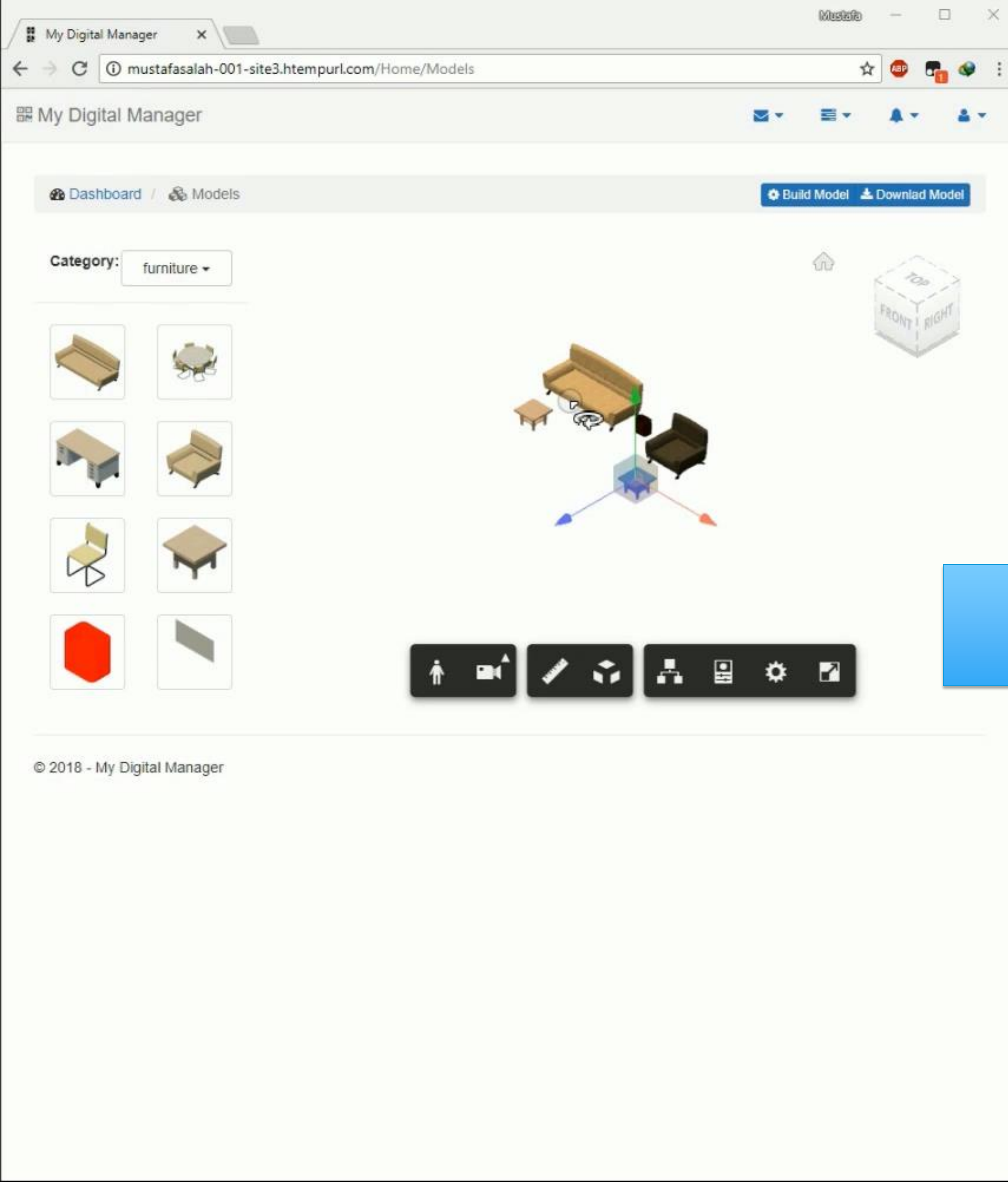
Dashboard / Models

Category: furniture

3D view of a living room interior with a sofa, armchair, and coffee table.

Navigation icons: Home, Camera, Rotate, Translate, Scale, Delete, Undo, Redo, Refresh, Help.

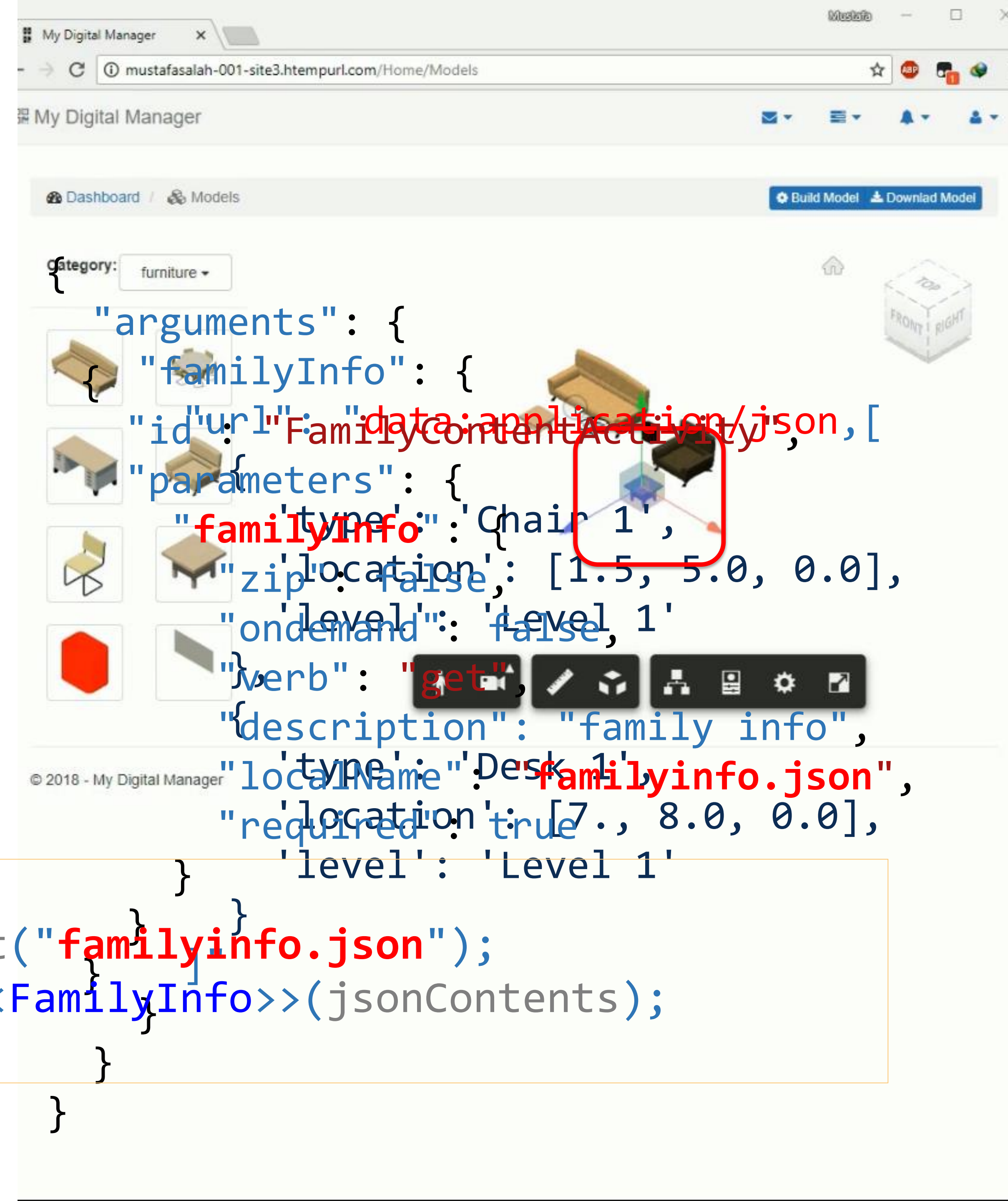
© 2018 - My Digital Manager

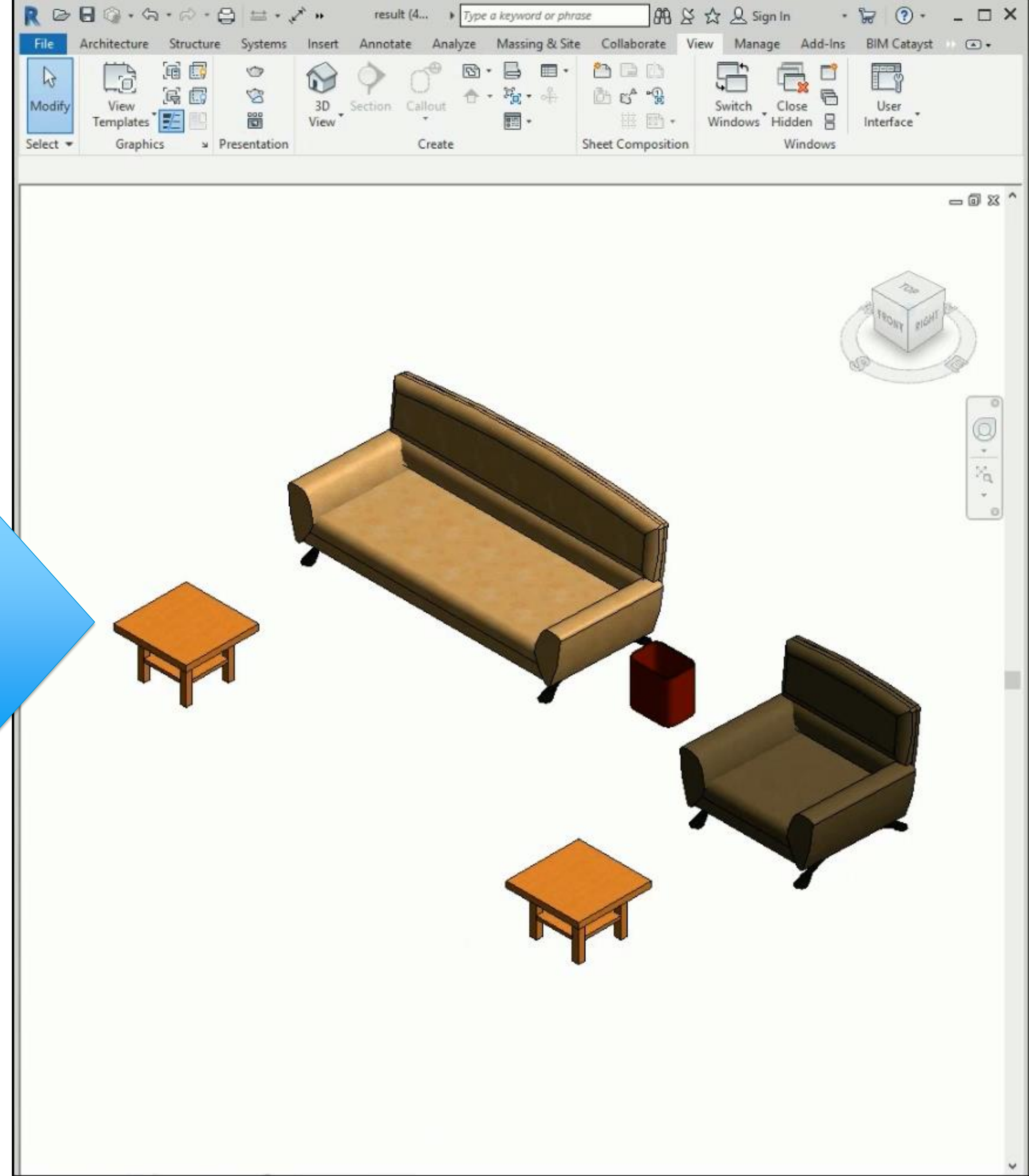
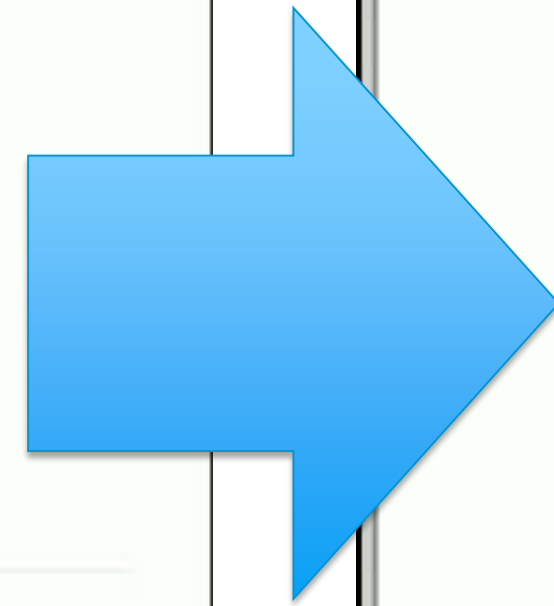
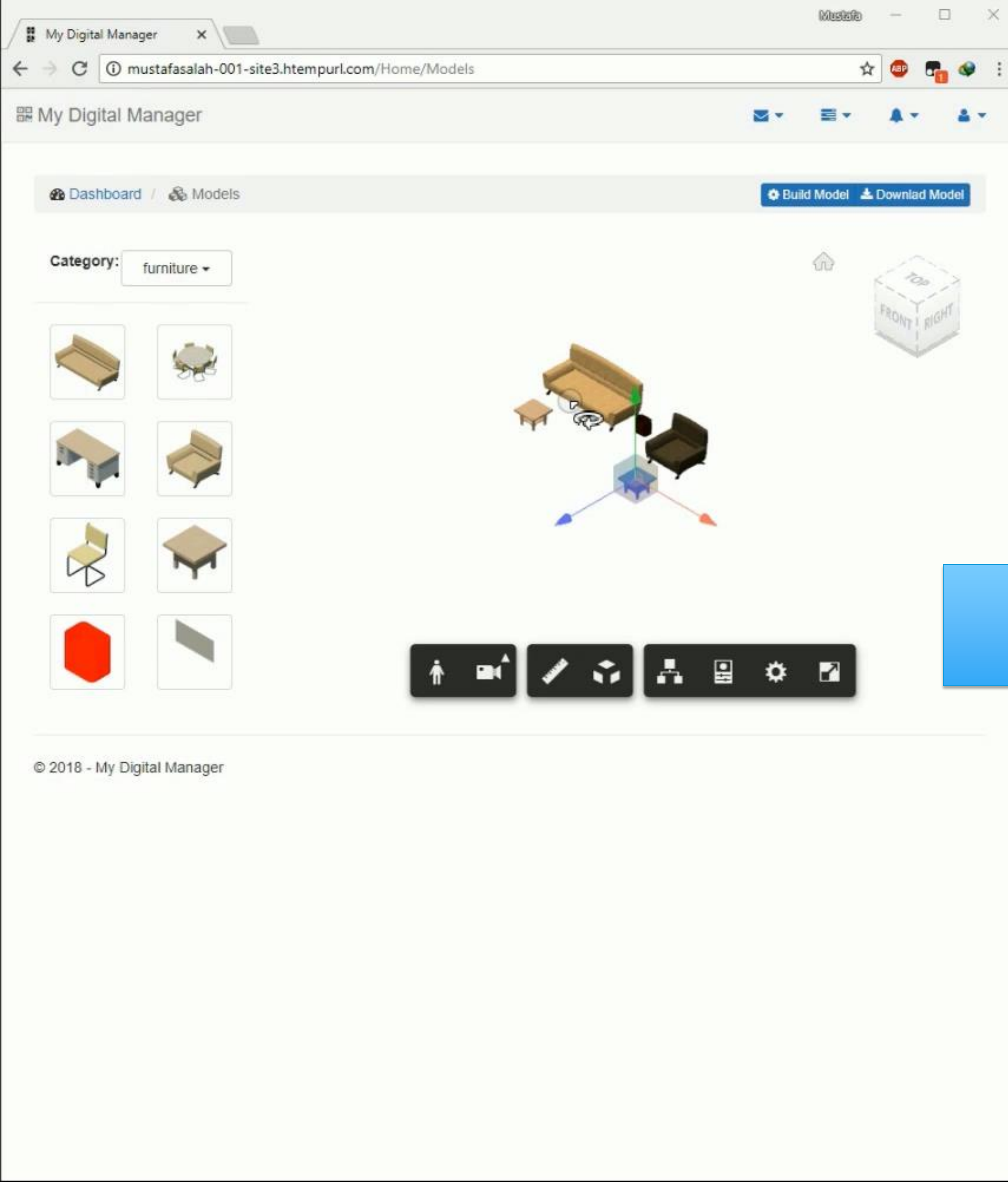


Behind the scenes

- Specify an *activity* parameter that saves json file with local name
- Send json payload in the *workitem*
- Read the json file in the *appbundle* code

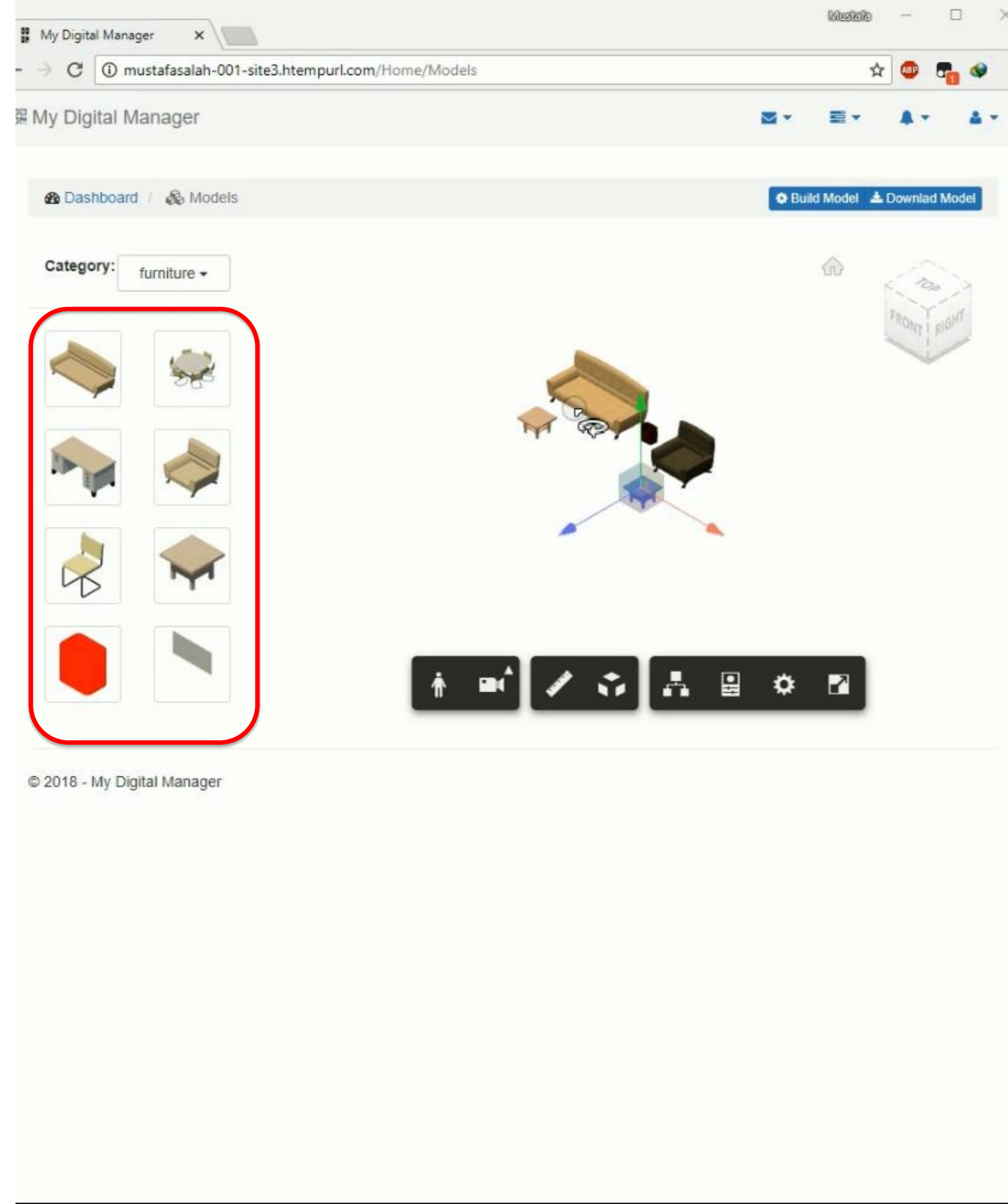
```
var jsonContents = File.ReadAllText("familyinfo.json");  
JsonConvert.DeserializeObject<List<FamilyInfo>>(jsonContents);
```





Handle variable number of inputs

- Zip all the contents into a single file and provide as input argument to the **workitem**
- Zip all the contents into a single file and package it with **appbundle**
- Create a template file with all family types loaded

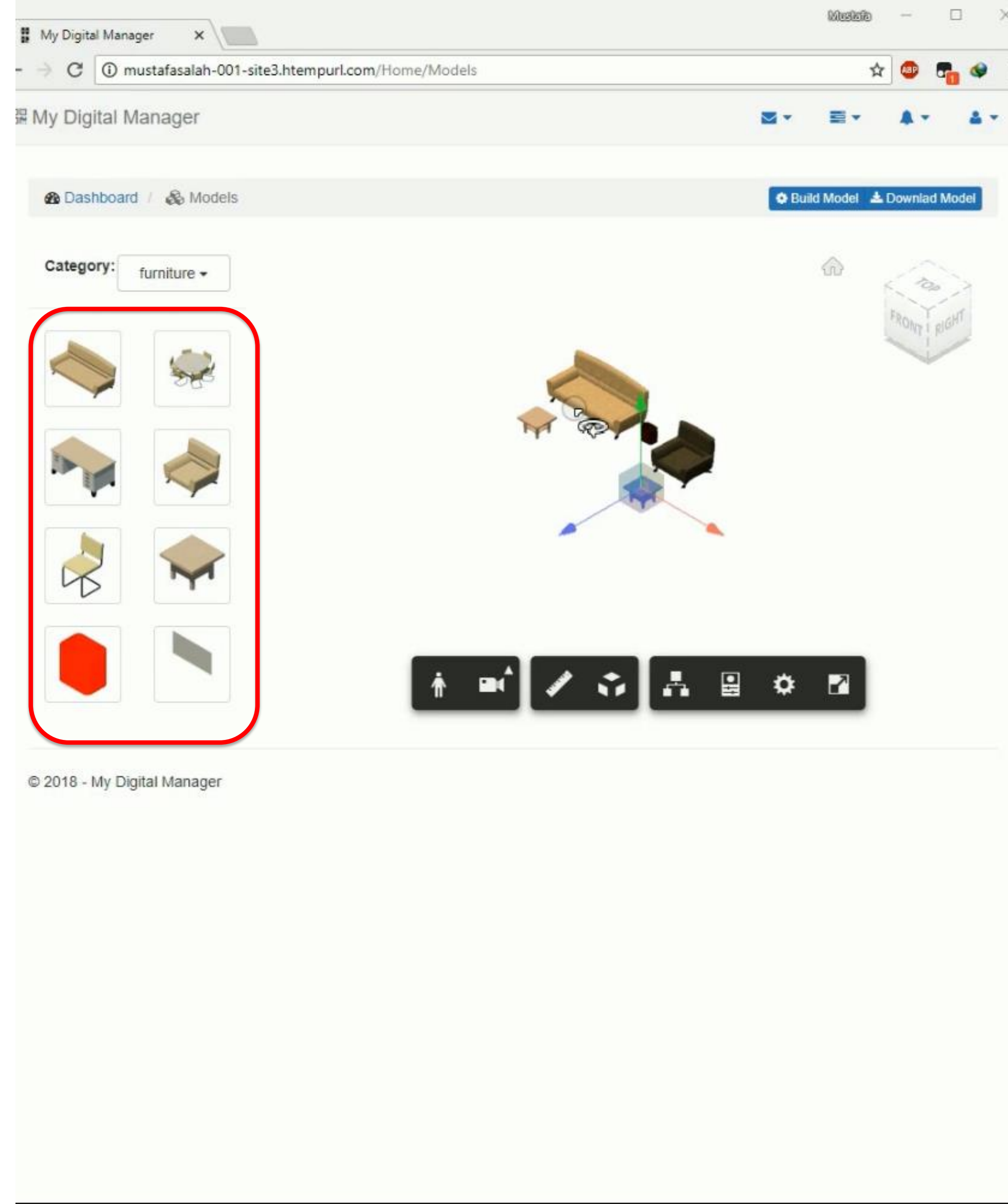


Family types and Forge viewables

1. Place a single instance in a Revit project
2. Generate the viewables
3. Use the thumbnails and geometry in the viewer (browser)

Or:

- Convert rfa to dwf



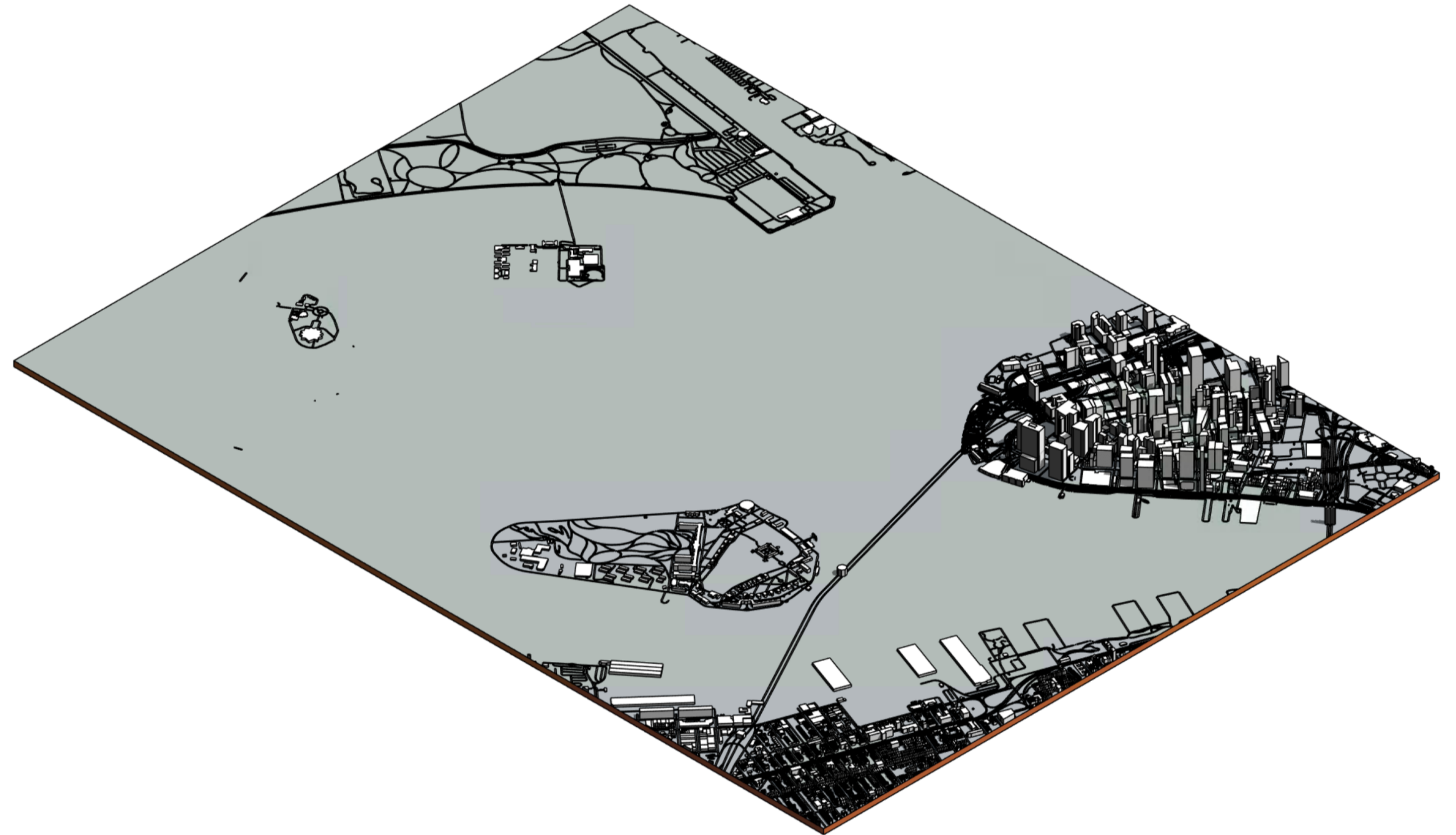
City Extractor Demo

Who

Blackbird Industries
Thiago Almeida

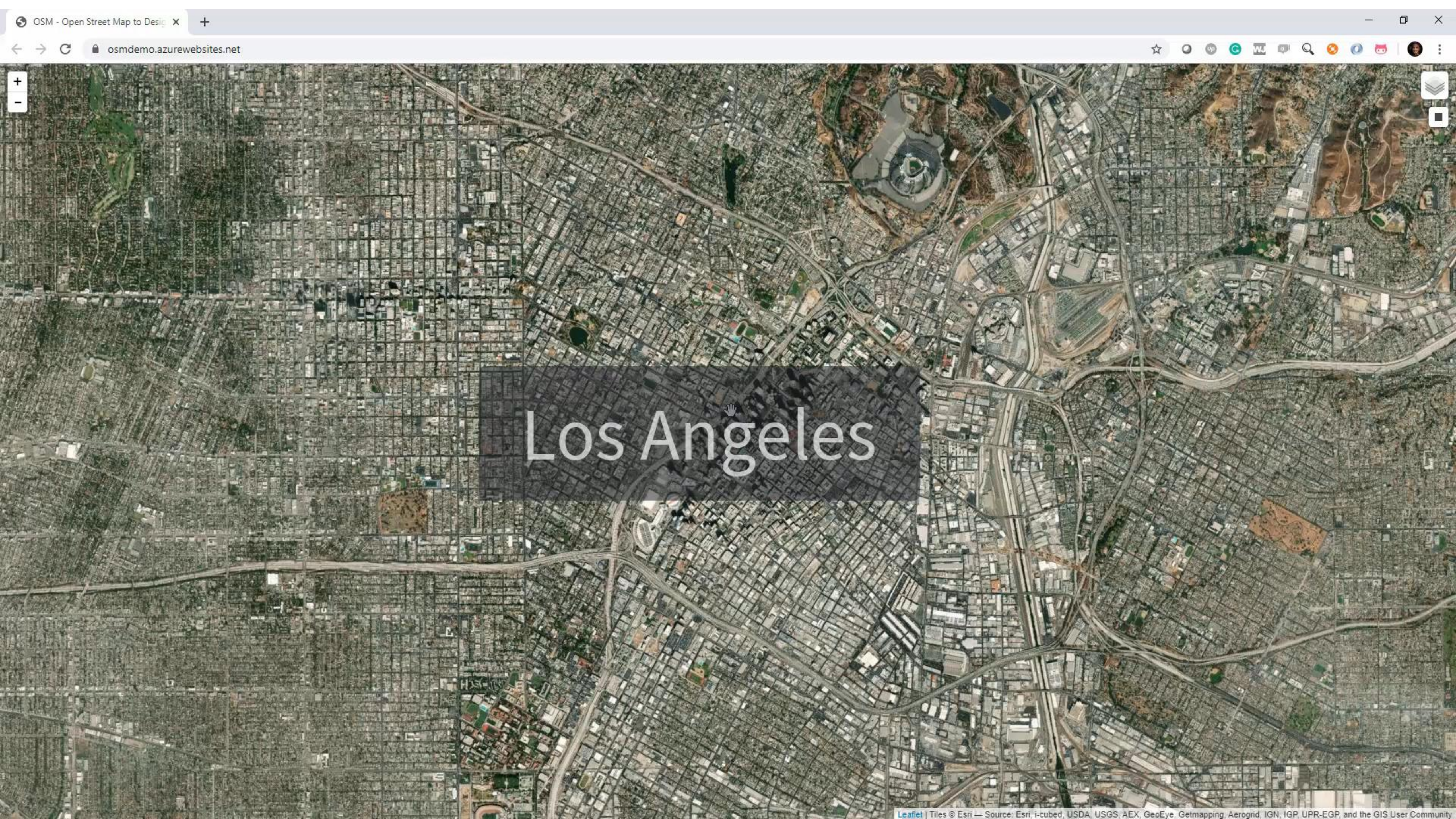
Solution

Use Open Street API to extract buildings, streets, highway information and generate buildings and streets in Revit using Direct Shapes.



APIs Used:





Los Angeles

Behind the scenes

- Specify an *activity* parameter that saves xml file with local name
- Specify OSM url along with coordinates bbox while posting the *workitem*

```
{  
  "id": "OsmDemoActivity",  
  "parameters": {  
    "osmData": {  
      "zip": false,  
      "ondemand": false,  
      "verb": "get",  
      "description": "OSM Data",  
      "localName": "osmData.xml",  
      "required": true
```

- Read the xml file in the *appbundle* code

```
{  
  "arguments": {  
    "osmData": {  
      "url": "https://api.openstreetmap.org/api/0.6/map?bbox=-71.061,42.356,-71.055,42.362"  
    }  
  }  
  var osmData = File.ReadAllText("osmData.xml");  
  GenerateBuildings(osmData);  
}
```


Tips and tricks we learnt in the last year



Workshared files

- Not supported:
 - Live Revit Cloud Worksharing models (yet)
 - Local files
- Currently supported:
 - Published Revit Cloud Worksharing models (read only)
 - Central files
 - Central files from eTransmit

Workshared files

- Do not specify input argument in command line for the **activity**
- Specify a local name to your input parameter in **activity**
- In the **appbundle** use **OpenOptions**

```
var path = ModelPathUtils.ConvertUserVisiblePathToModelPath(input.rvt);
var opts = new OpenOptions
{
    DetachFromCentralOption = DetachFromCentralOption.DetachAndPreserveWorksets
};
var document = application.OpenDocumentFile(path, opts);
```

```
{
  "commandline": [
    "$(engine.path)\\revitcoreconsole.exe
/i $(args[inputFile].path)
    /al $(appbundles[MyBundle].path)"
  ]
  "id": "MyActivity",
  "parameters": {
    "inputFile": {
      "zip": false,
      "ondemand": false,
      "verb": "get",
      "description": "input revit file",
      "localName": "input.rvt",
      "required": true
    }
  }
}
```


Useful tips (best practices)

- Use *onComplete* arguments in workitems
- Debug logging in report log
- Use eTransmit or zipped archives for linked files
- Template and static data in *appbundles*
- Separate out DB logic from UI (and Network)

```
{
  "arguments": {
    "inputFile": {
      "url": "https://myhost.com/mybucket/myFile.inv"
    },
    "onComplete": {
      "url": "https://myhost.com/workitemcomplete",
      "verb": "post"
    }
  }
}

try
{
  ExportViewsToFBX(document);
}
catch (Exception e)
{
  Console.WriteLine($"Exception occurred while trying to export views {e.Message}");
}
```

New features

- Nuget package for DesignAutomationBridge
- Support for Revit 2020
- Support for locale
- IFC support
- Ability to override the timeout limit
- Ability to dynamically insert workitem id into output URL
- Added onDemand feature

Where to go from here



Forge Accelerator

<http://autodeskcloudaccelerator.com>

Autodesk Forge Accelerator Programs

Expand your business with Autodesk Forge

A series of Accelerator programs for creative developers to spend quality time working on a Forge project.

Current offering for 2019

Paris, France – December 9-13

Current offering for 2020

San Francisco, CA – January 13-17

Bogota, Colombia – February 3-7

London, United Kingdom – February 17-21

Sydney, Australia – March 9-13

São Paulo, Brazil – April 6-10

Boston, MA – April 27-May 1

Barcelona, Spain – June 8-12

[Contact us](#) if you are interested in a future Accelerator.

Autodesk Forge

- Home
- Before you start coding
- Tools
- OAuth
- View your models
 - Create a server
 - Authenticate
 - Upload file to OSS
 - Translate the file
 - Show on Viewer
- View BIM 360 & Fusion models

 Create a server

<https://learnforge.autodesk.io/>

Learn Autodesk Forge

Learn the basics of authentication, data management, file translation, and model rendering, with our quickstart guides.

What is it?

Forge enables companies to leverage design and engineering data to develop custom software applications and connected workflows for manufacturing, media/entertainment, architecture, engineering, and construction.

- **View 3D models directly in your browser:** The Viewer allows you to embed, interact with, and retrieve meta data about design files in over 50 formats right in your browser, without having to install extra software.
- **Centrally manage data:** The Data Management API allows you to access your data

Visit us at the Forge center, in the middle of the ADSK in the Expo space all week...



Forge Center

Meet The Forge Leaders
Networking Happy Hour
(Wednesday 5.30-7.30PM)

Participate in the Forge Scavenger Hunt and be entered into a daily raffle for some cool prizes!



Forge Dev Zone

Hands-on technical support on specific APIs/Services (Tuesday-Thursday)

| | | |
|-----------|---------------|--|
| Tuesday | 12:00-2:00PM | Forge Design Automation |
| Tuesday | 5:30-7:30PM | Forge Viewer |
| Wednesday | 10:00-12:00PM | Forge Data Management & Model Derivative |
| Wednesday | 12:00-2:00PM | Forge and BIM 360 |
| Thursday | 12:00-2:00PM | Forge Design Automation |



Forge Answer Bar

Get answers from our experts
(Tuesday-Thursday)

Questions





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.

© 2019 Autodesk. All rights reserved.

