

Forge Design Automation for Revit Building Web RVT Applications on Forge

Jeremy Tammik

The Building Coder, Forge and BIM Technical Evangelism, Autodesk



Disclaimer

We may make statements regarding planned or future development efforts for our existing or new products and services. These statements 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. These planned and future development efforts may change without notice. Purchasing decisions should not be made based upon reliance on these statements.

These statements are being made as of Tuesday, October 16, 2018, and we assume no obligation to update these forward-looking statements to reflect events that occur or circumstances that exist or change after the date on which they were made. If this presentation is reviewed after this date, these statements may no longer contain current or accurate information.

We are in an invitation-only beta!

- The system is not available right now
- We'll tell you what you do to get ready today

Questions we'll answer today

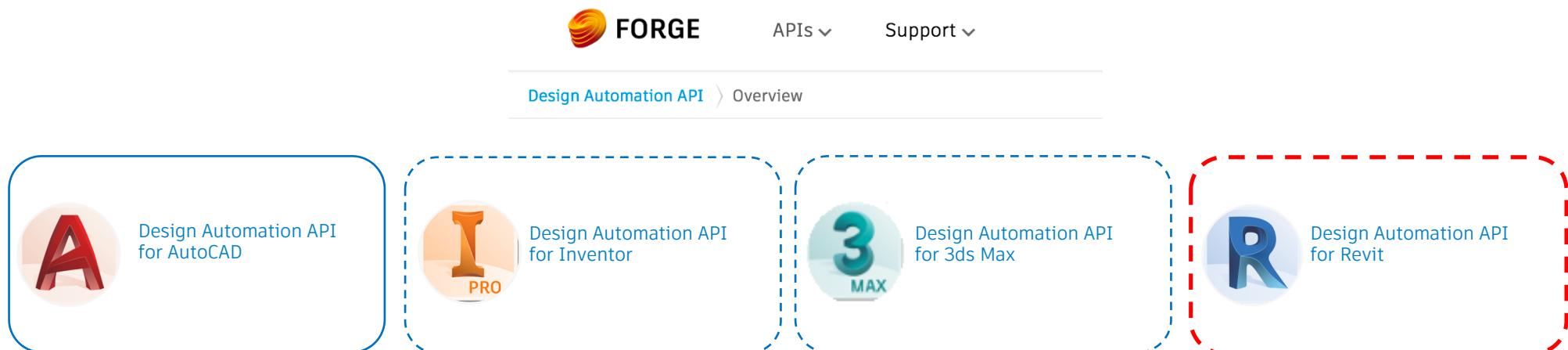
- What is Design Automation API for Revit?
- How does it fit into the Forge ecosystem?
- What **workflows** can you support with it?
- How do you **write an application** for the system?
- What's on the roadmap?
- How do you contact us?



Design Automation API for Revit

Forge Design Automation API for Revit is...

- ... the Revit engine on the cloud, which allows you to ...
 - customize Revit workflows
 - automate tasks
 - create cloud-based solutions to problems

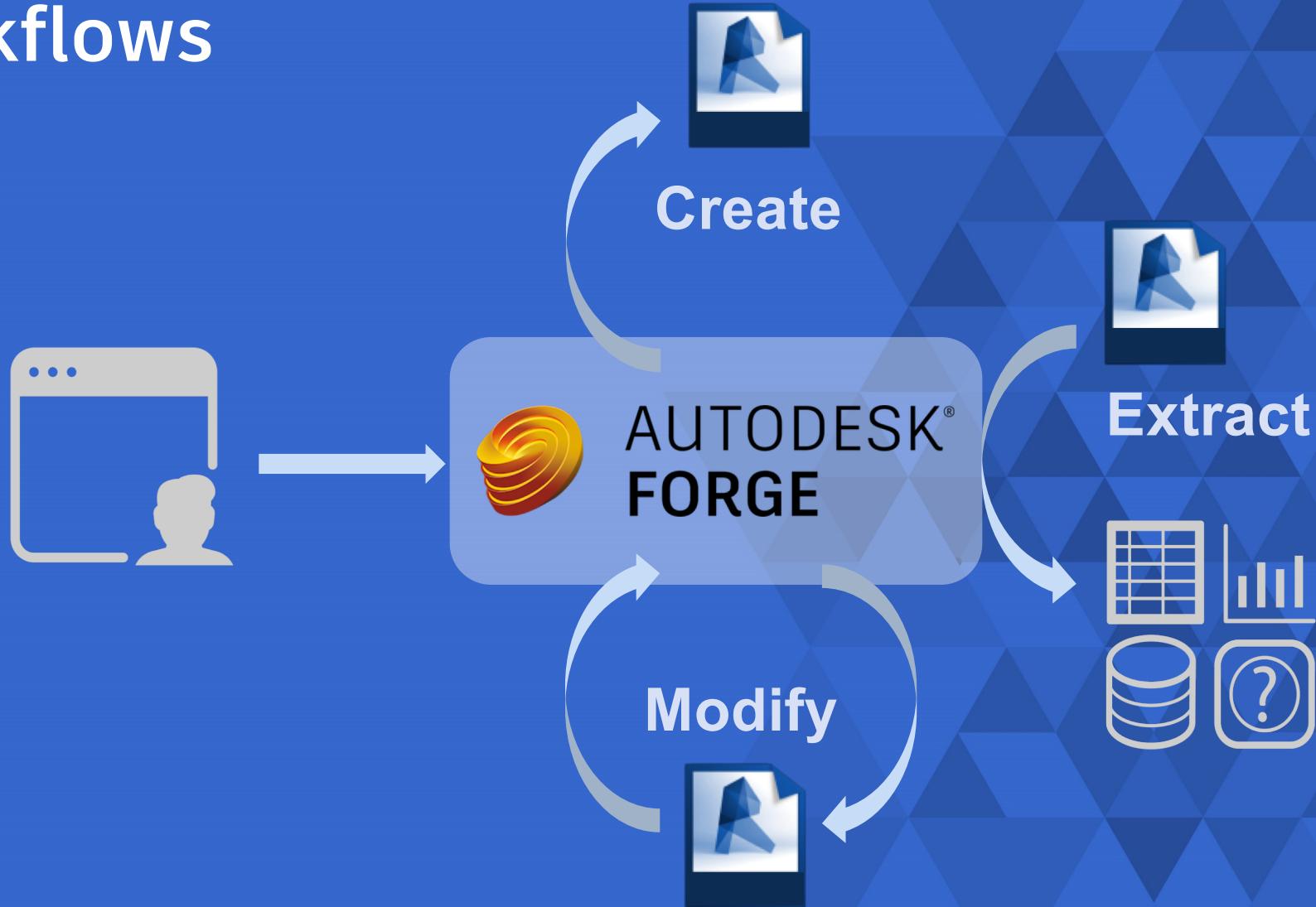


Service Basics

- Similar to Design Automation API for AutoCAD
- REST API endpoints
- Run automated tasks
- Access to Revit's "business logic" – no UI



Workflows

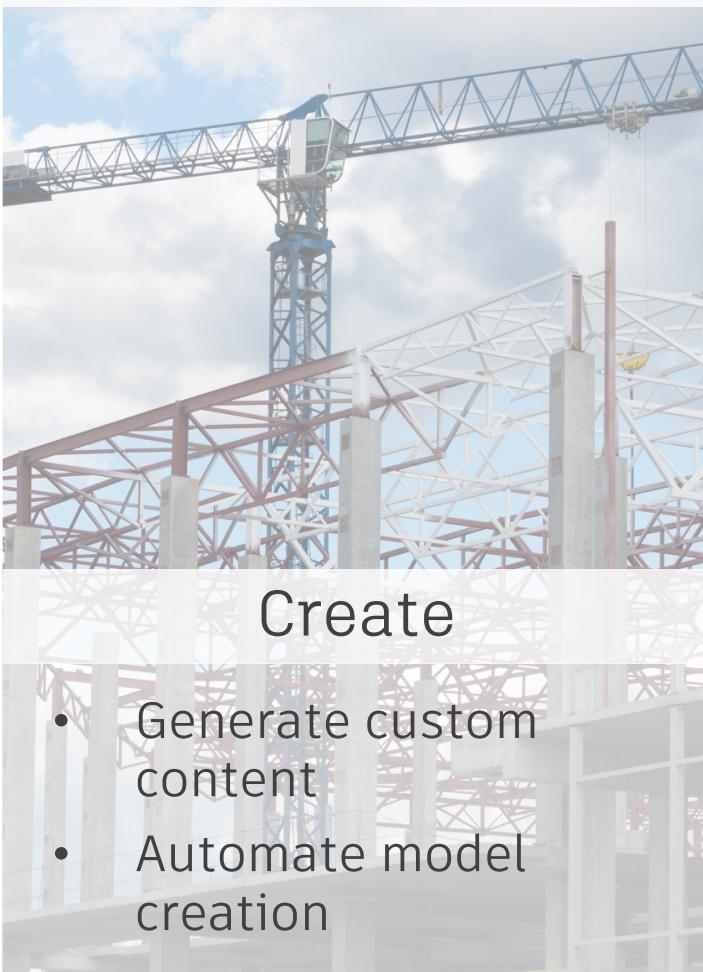




AUTODESK®
FORGE

Design Automation API for Revit

Use Revit data in cloud-native applications to automate at scale



Create

- Generate custom content
- Automate model creation



Extract

- Explore & analyze model data
- Produce automated reports

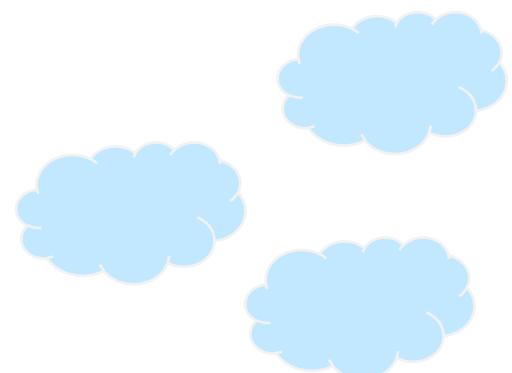


Modify

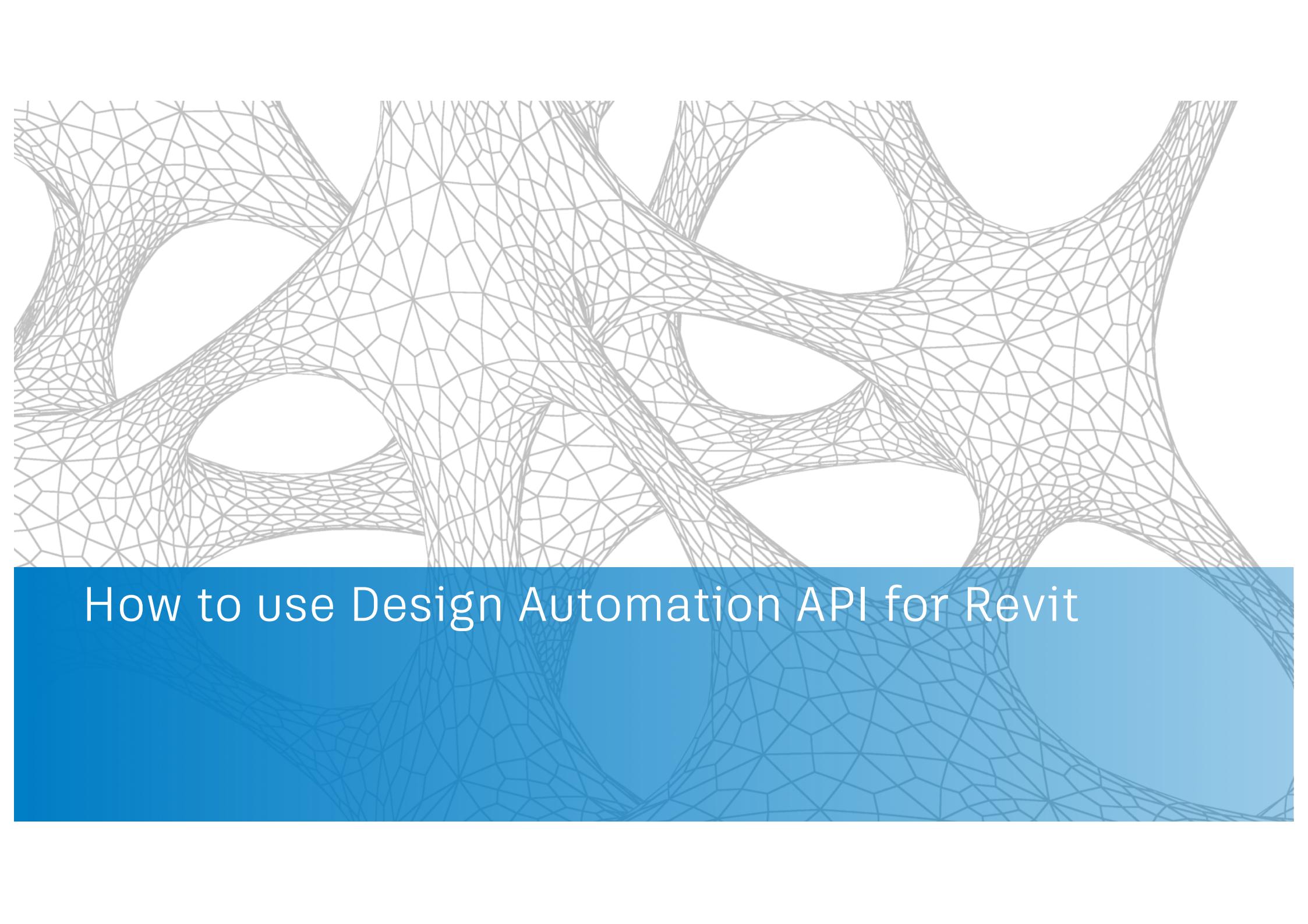
- Maintain company standards
- Automatically create documentation

Benefits of the cloud

- Cloud hosted apps
- Create new apps for end users without Revit
- Work with models already in the cloud
- Higher capacity than a desktop machine
- Let us handle hardware maintenance and infrastructure



Demo

The background of the slide features a complex, abstract wireframe mesh pattern composed of numerous thin, light-grey lines forming a organic, flowing structure.

How to use Design Automation API for Revit

Design Automation API terms

Name	What is it	How you use it	Example
engine	Version of Revit engine in the cloud	Apps and activities use a specific engine version	Revit 2018's engine
app	Zip file containing your Revit code, cf. AppStore bundle	Upload via the DA API and access later to run your Revit add-in	Zipped add-in which generates Revit stairs from a spreadsheet
activity	Definition of an action for Revit to run	Define via the DA API	"MyStairsGeneratorActivity" which you invoke to run your apps
workitem	Job submitted to cloud and run on Revit	POST to run your Revit add-in	REST API call to run the stairs generator against a specific input spreadsheet

Using the service

Phase 1: Setup

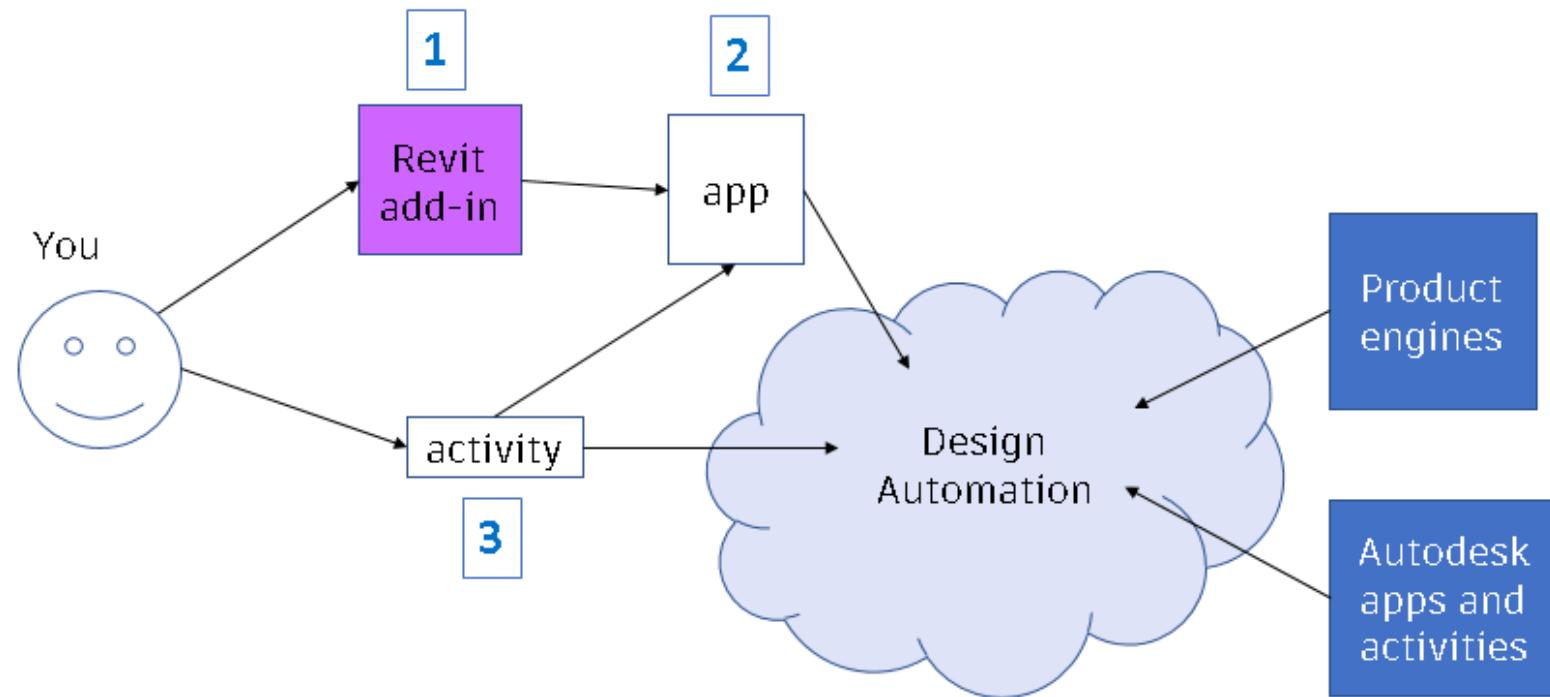
Set up a Forge account	Create or convert Revit add-in	Create code bundle	Upload code bundle	Define the activity
------------------------------	--------------------------------------	-----------------------	-----------------------	------------------------

Phase 2: Execution

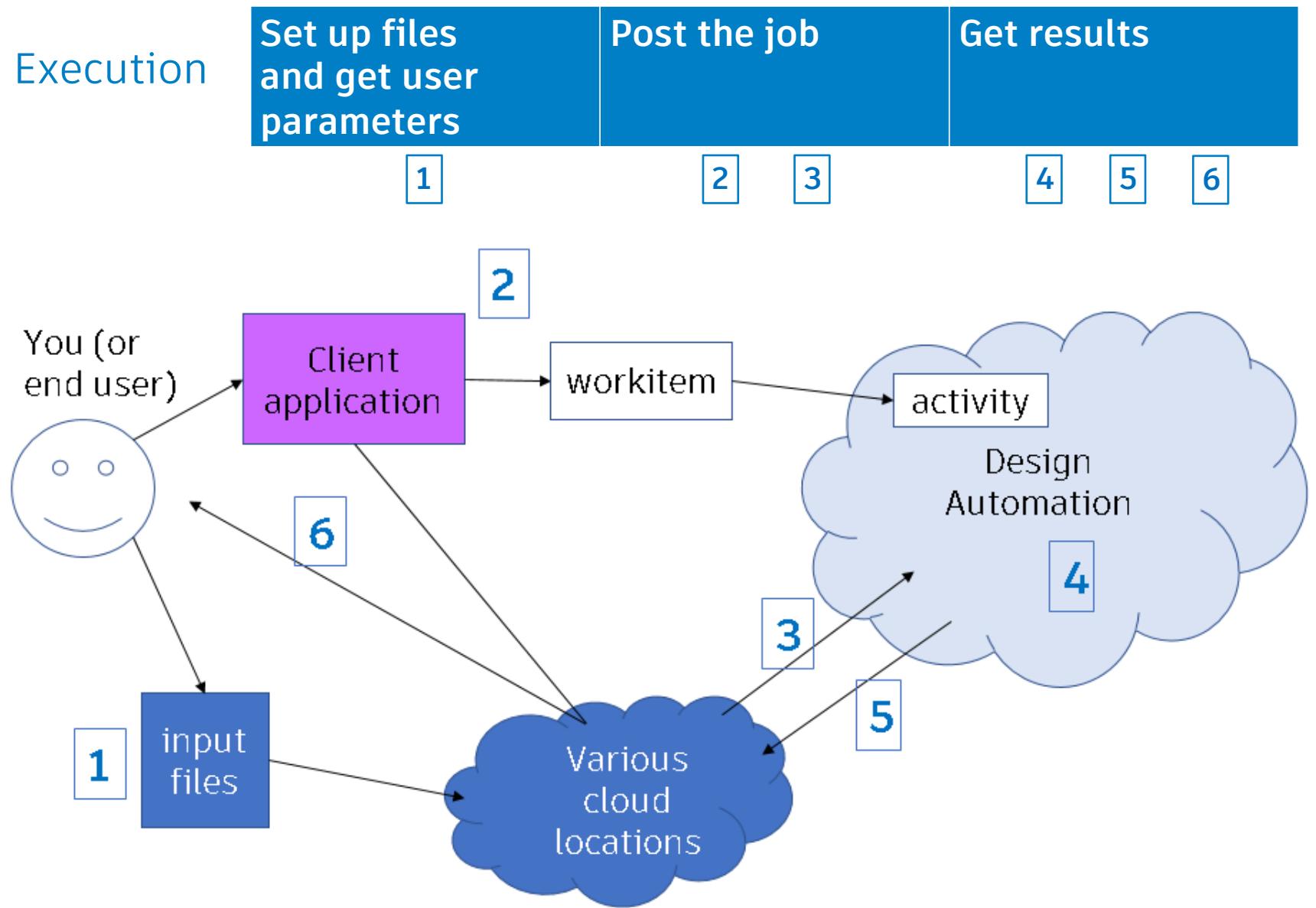
Set up files and get user parameters	Post the job	Get results
--	--------------	-------------

Phase 1: Setup

Set up a Forge account	Create or convert Revit add-in	Create code bundle	Upload code bundle	Define the activity
	1	2	2	3



Phase 2: Execution



Using the service

Phase 1: Setup

Set up a Forge account	Create or convert Revit add-in	Create code bundle	Upload code bundle	Define the activity
------------------------------	--------------------------------------	-----------------------	-----------------------	------------------------

Phase 2: Execution

Set up files and get user parameters	Post the job	Get results
--	--------------	-------------

Create code bundle (“app”)

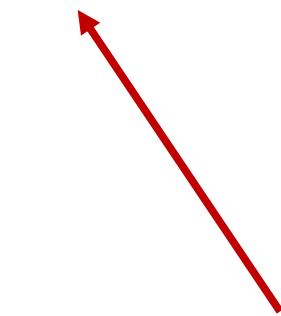
Set up a Forge account	Create or convert Revit add-in	Create code bundle	Upload code bundle	Define the activity
------------------------	--------------------------------	---------------------------	--------------------	---------------------

```
MyStairsGenerator.zip
| --MyStairsGenerator.bundle
| --Contents
|   | --MyStairsGenerator.addin
|   | --MyStairsGenerator.dll
|   | --<other needed references>
| --PackageContents.xml
```

Upload code bundle (“app”)

Set up a Forge account	Create or convert Revit add-in	Create code bundle	Upload code bundle	Define the activity
------------------------	--------------------------------	--------------------	---------------------------	---------------------

1. Call the “get upload URL” endpoint to get a signed URL for your app
2. POST your app to that URL



Don't forget this!

Define the activity

Set up a Forge account	Create or convert Revit add-in	Create code bundle	Upload code bundle	Define the activity
------------------------	--------------------------------	--------------------	--------------------	----------------------------

- Call the “define activity” endpoint to define your job
- Specify engine version
- Specify output file name
- Specify associated apps

Command type: POST

Headers: ‘Content-Type: application/json’

‘Authorization: Bearer [INCLUDE_TOKEN_HERE]’

Data: {
 “body”: {
 “id”: “**MyStairsGeneratorActivity**”,
 “commandLine”: “\$(engine.path)\\\\\\accoreconsole.exe
/i \$(args[**HostDwg**].path) /al
\$(apps[**MyStairsGenerator**].path)”,
 “parameters”: {
 “**HostDwg**”: {
 “localName”: “\$(HostDwg)”
 },
 “**Result**”: {
 “localName”: “**result.rvt**”
 }
 },
 “engine”: “**Autodesk.Revit+2018**”,
 } }

Post the job (“POST workitem”)

Set up files and
get user input

Post the job

Get results

- Call the “post job” endpoint to send a task to run
- Specify input and output arguments

Command type: POST

Headers: ‘Content-Type: application/json’
‘Authorization: Bearer [INCLUDE_TOKEN_HERE]’

Data: {
"activityId" : "Revit.MyStairsGeneratorActivity+prod",
"arguments" :
{
"HostDwg" : { "url" :
<https://path/to/signed/url/input.csv> },
"Result" : { "url" :
<https://path/to/signed/url/result> , "verb" : "put" }
}
}'

Get results

Set up files and
get user input

Post the job

Get results

- Check job status with the “check status” endpoint
- Get results with the “workitem” endpoint

Endpoint includes the WORKITEM_ID
Command type: GET
Headers: ‘Content-Type: application/json’
‘Authorization: Bearer [INSERT_TOKEN_HERE]’
Data: None

Converting a Revit add-in



1. Remove UI
 - Add-in must implement IExternalDBApplication
 - Separate into UI and DB layers to keep running on desktop
2. Use our “bridge” library to subscribe to the “Design Automation is Ready” event
 - Run your code in the event handler

Converting a Revit add-in

Set up a Forge account

Create or convert Revit add-in

Create code bundle

Upload code bundle

Define the activity

```
[Autodesk.Revit.Attributes.Regeneration(Autodesk.Revit.Attributes.RegenerationOption.Manual)]
[Autodesk.Revit.Attributes.Transaction(Autodesk.Revit.Attributes.TransactionMode.Manual)]
public class MyStairsGenerator : IExternalDBApplication
{
    public ExternalDBApplicationResult OnStartup(
        Autodesk.Revit.ApplicationServices.ControlledApplication app)
    {
        DesignAutomationBridge.DesignAutomationReadyEvent += HandleDesignAutomationReadyEvent;
        return ExternalDBApplicationResult.Succeeded;
    }

    public void HandleDesignAutomationReadyEvent(object sender, DesignAutomationEventArgs e)
    {
        Autodesk.Revit.ApplicationServices.Application rvtApp = e.DesignAutomationData.RevitApp;
        bool stairsCreated = LayoutUtils.GenerateStairs(rvtApp, LayoutUtils.ReadSpreadsheetData());
        if (stairsCreated)
            e.Succeeded = true;
    }
    // more code ...
}
```

Considerations Converting a Revit add-in

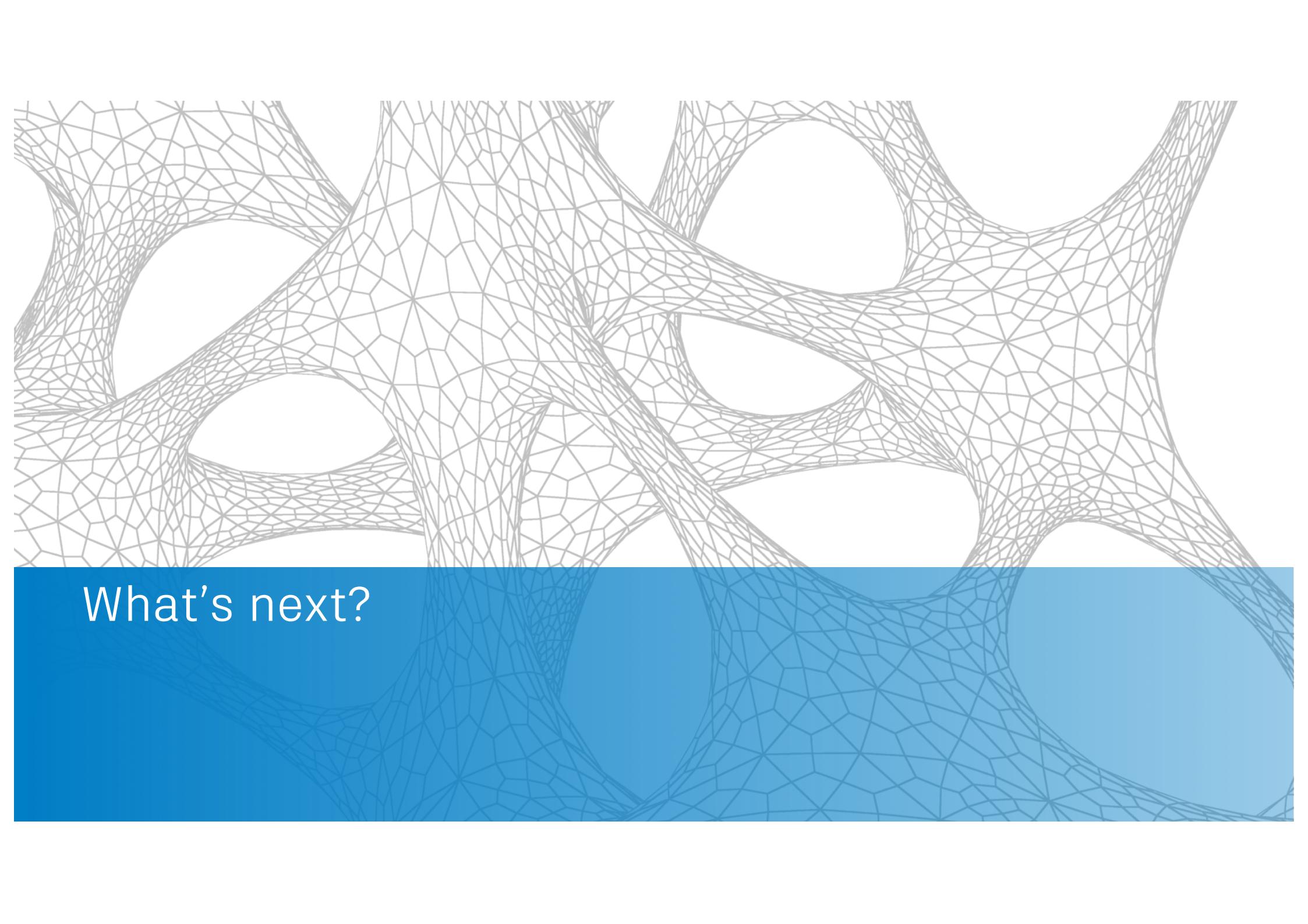
Set up a Forge account	Create or convert Revit add-in	Create code bundle	Upload code bundle	Define the job
------------------------	---------------------------------------	--------------------	--------------------	----------------

Potential issue	You should...
No active view	Call functions which explicitly take a view
No automatic prompting user for errors	Write a failure handler
Controlled disk access	Create files only in or under the working directory
No interaction during execution	Collect all input up front
No network access during execution	Do network calls before or after you send a job to the service

More demos!

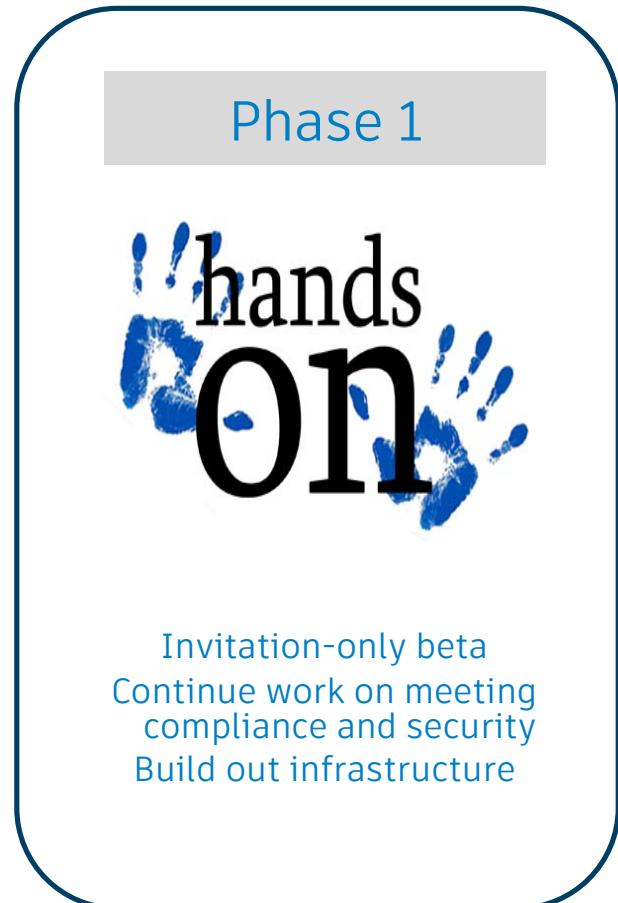
Run DA4R App on Model in BIM360

- Store model in BIM 360
- Display model in custom viewer
- Modify properties and save changes in custom viewer extension
- Launch DA4R app from custom viewer button
- Update BIM360 seed model using DA4R

The background features a complex, organic geometric pattern composed of numerous thin, light gray lines forming a mesh of triangles and quadrilaterals. This pattern is highly symmetrical and resembles a molecular or cellular structure. A solid blue rectangular area is overlaid on the bottom half of the image, containing the text "What's next?".

What's next?

Roadmap



Public preview (beta)
Compliance and security
addressed
Infrastructure testing and
enhancements

Public release
API changes based on
customer feedback

Getting started – what you can do now

- Remove UI dependencies
- Decide how to handle errors
- Decide what user input you need
- Splitting your application: one job or several?
- The Building Coder:

- [Swallowing StairsAutomation Warnings](#)

<http://thebuildingcoder.typepad.com/blog/2018/09/swallowing-stairsautomation-warnings.html>

- [Auto-Run an Add-In for Design Automation](#)

<http://thebuildingcoder.typepad.com/blog/2018/09/auto-run-an-add-in-for-design-automation.html>

How to contact us

- Talk with me here
- Find us at the Revit Idea Exchange
- Take the survey: www.autodesk.com/revitonforge
- E-mail us: revitonforge@autodesk.com

Useful links

- Design Automation API v2 Documentation:
<https://developer.autodesk.com/en/docs/design-automation/v2/overview/>
- My first Revit add-in (UI!): <https://www.autodesk.com/myfirstrevitplugin>
- AU 2017 Classes
 - [SD125457](#) – Design Automation: Building Web Applications with Revit Data on Forge
 - [SD124720](#) – The New and Expanded Forge Design Automation API – Generic Design Automation API class
 - [FDC124076](#) – Moving to Forge and the Cloud with Your Existing .NET Experience



AUTODESK®

Make anything.

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 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.
© 2017 Autodesk. All rights reserved.

