

Bringing your Inventor Add-in into Forge Design Automation

Andrew Akenson

Software Architect | Autodesk



```
if (!you.IntoCode()) {  
    System.Console.WriteLine("This may not be for you");  
}
```



Design Automation Overview



Inventor C# Add-in example

During the course of this presentation, we may make statements regarding future events and/or statements regarding planned or future development efforts for our existing or new products and services. We wish to caution you that such statements reflect our current expectations, estimates and assumptions based on factors currently known to us and that actual events or results could differ materially. Also, these statements are not intended to be a promise or guarantee of future delivery of products, services or features but merely reflect our current plans, which may change. Purchasing decisions should not be made based upon reliance on these statements. The statements made in this presentation are being made as of the time and date of its live presentation. We do not assume any obligation to update any statements we make to reflect events that occur or circumstances that exist after the date of this presentation.



FORGE OFFERING



Data Management API



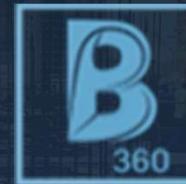
Design Automation API



Model Derivative API &
Viewer



Reality Capture API



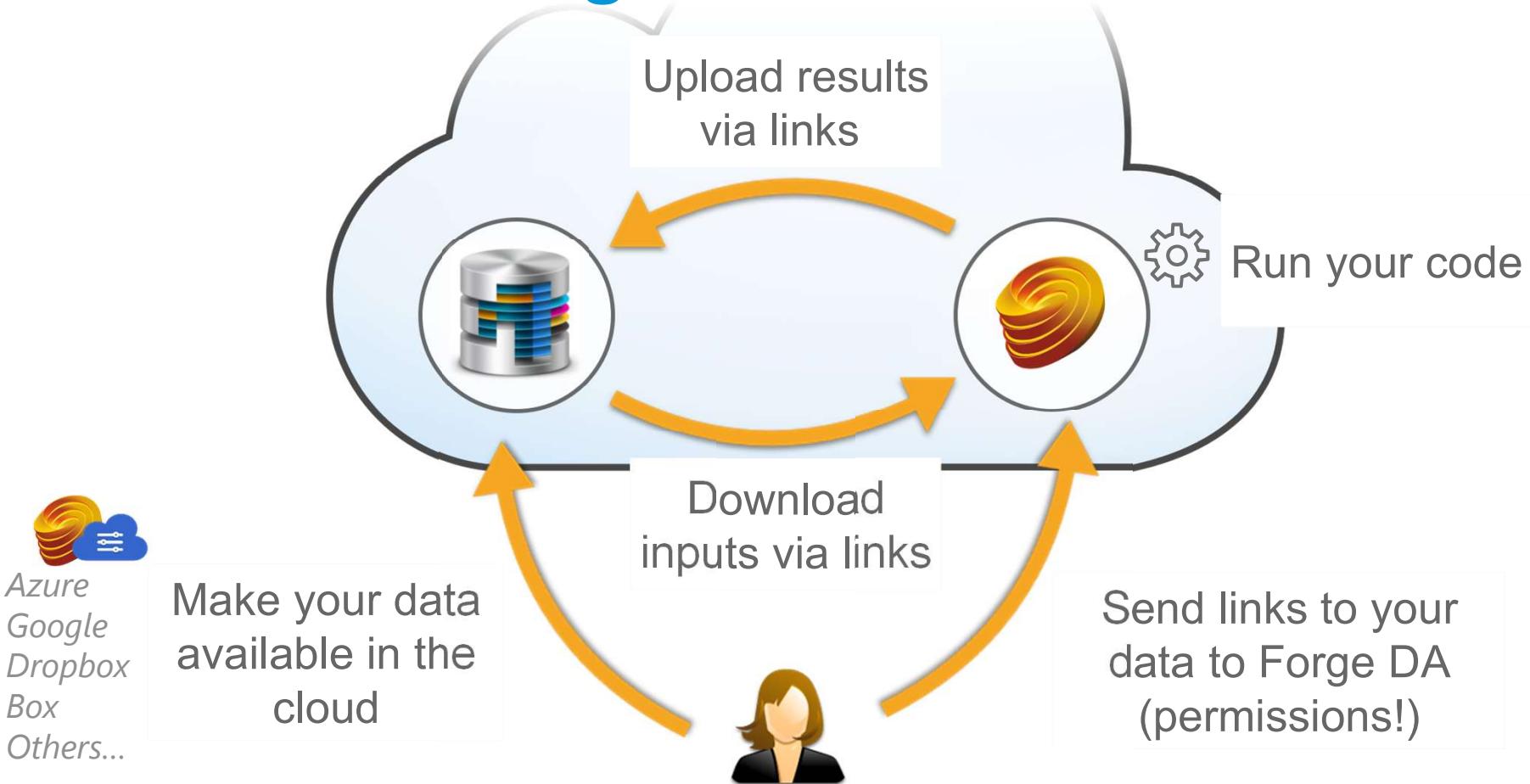
BIM 360 API



Webhooks API

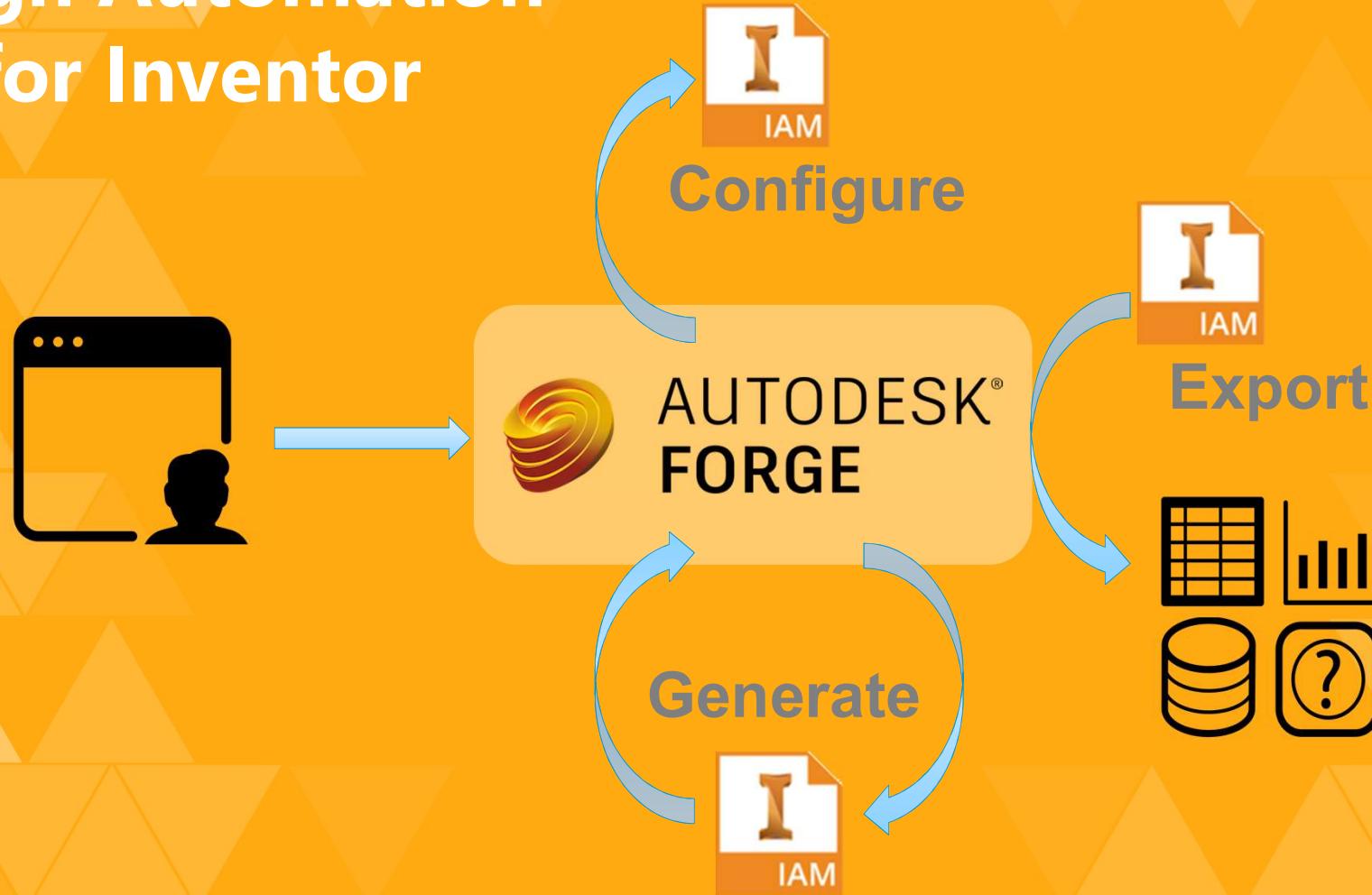
Design Automation for Inventor
now in Public Beta!

Data Processing on Demand



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

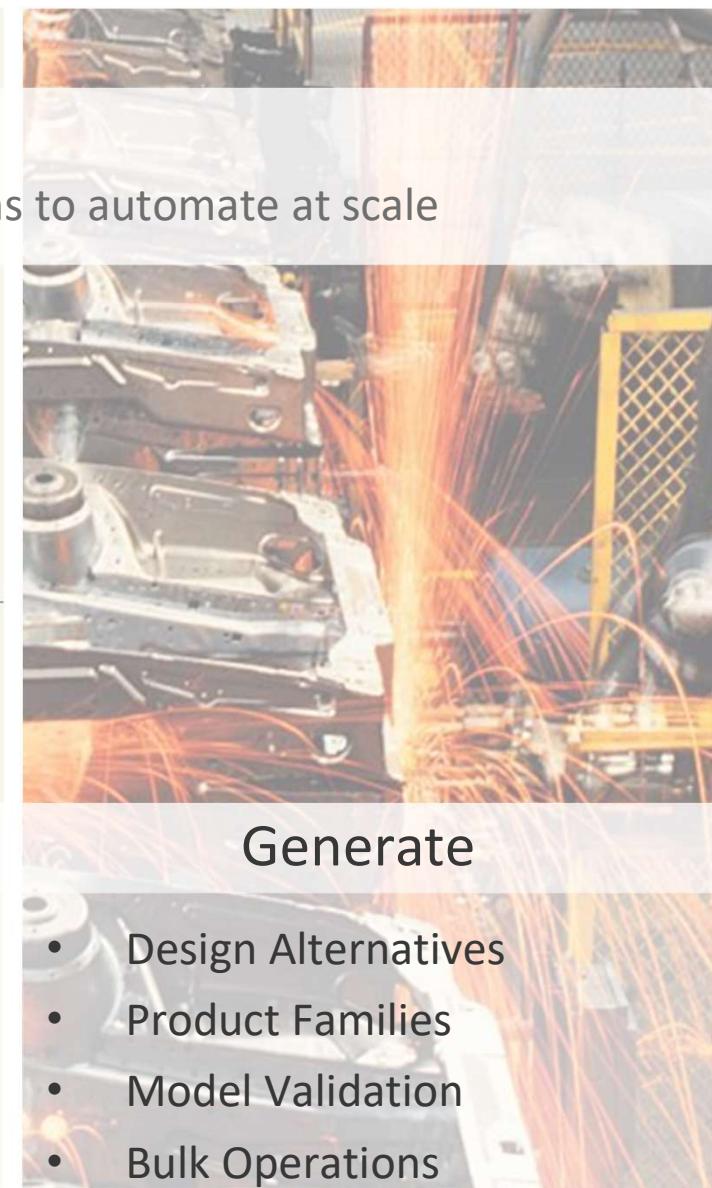
Design Automation API for Inventor



AUTODESK® FORGE

Configure

- Engineering Automation
- Sales Configuration
- iLogic



Add-in Proof of Concept

Who

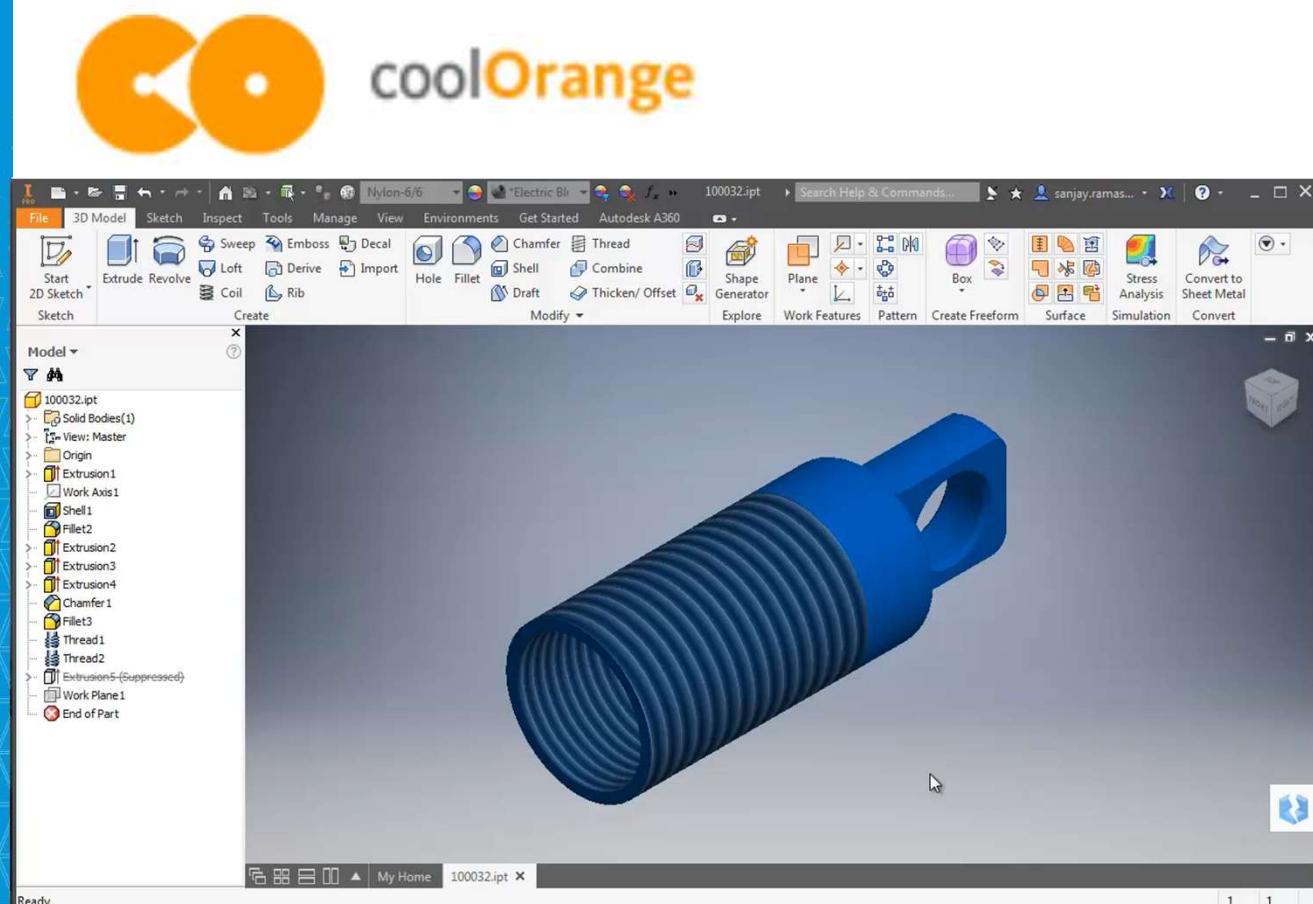
Development partner working with customers in data management and custom Add-ins.

Situation

Work with the Design Automation team to bring an Add-in into Design Automation for Demos

Solution

Starting with an existing Inventor Add-in that takes cosmetic threads and creates models threads for printing, convert this to a Design Automation plugin to show how to take existing Inventor Add-ins and automate that workflow in Forge.



APIs Used:



Design
Automation



Viewer

GENERATE

Design Automation Terminology



The API

V3 HTTP Endpoint	Programming concept	Product concept	V2 HTTP Endpoint
/workitems	Function call	Product execution, job	/WorkItems
/activities	Function definition	Input/Output types, parameters	/Activities
/appbundles	Shared library	Plugin	/AppPackages
/engines	Instruction set	Product (inventor, autocad..) to use	/Engines

JS4 This seems to be to deeply technical. But it depends on the auditorium - who it is targeted at.

Jirka Stejskal, 6/6/2018

Design Automation in Inventor Terminology



APPBUNDLES

Inventor Plugins



ACTIVITIES

Interface between AppBundle and Work Item



WORKITEMS

Job to run on InventorServer in Forge



ENGINE

InventorServer – multiple versions



Inventor AppBundles

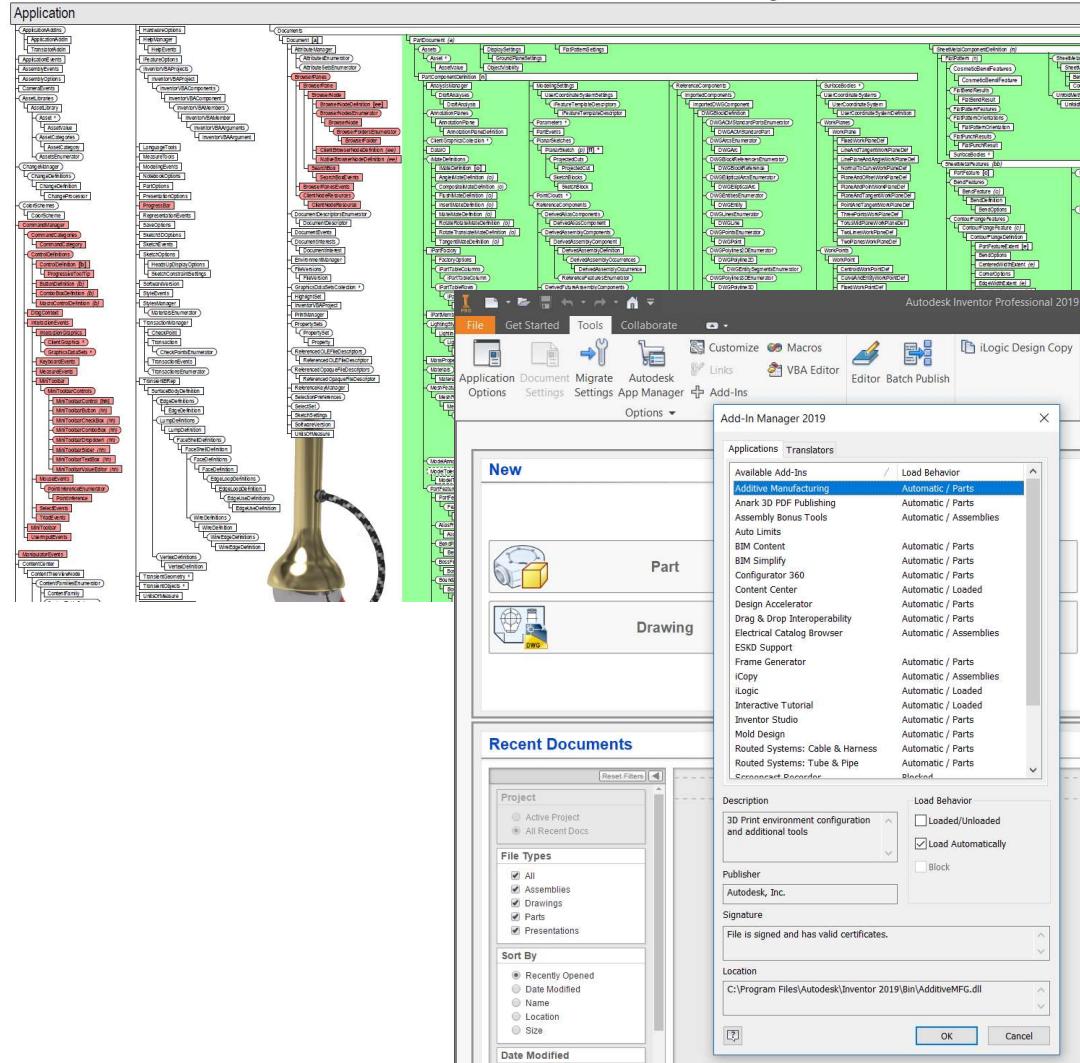
Upload Inventor plug-in code

- Plug-ins are like Add-ins without UI.
- Supports all Inventor API languages (C++, C#, VB.NET)
- Upload your own code or use shared AppBundles
- Specify Inventor version
- Zip file containing plug-in and any other dependencies

Can also be:

- **Shared** – By Forge ID or everyone
- **Aliased** – Production, Beta, Alpha
- **Versioned**

AUTODESK® INVENTOR® 2019 API Object Model



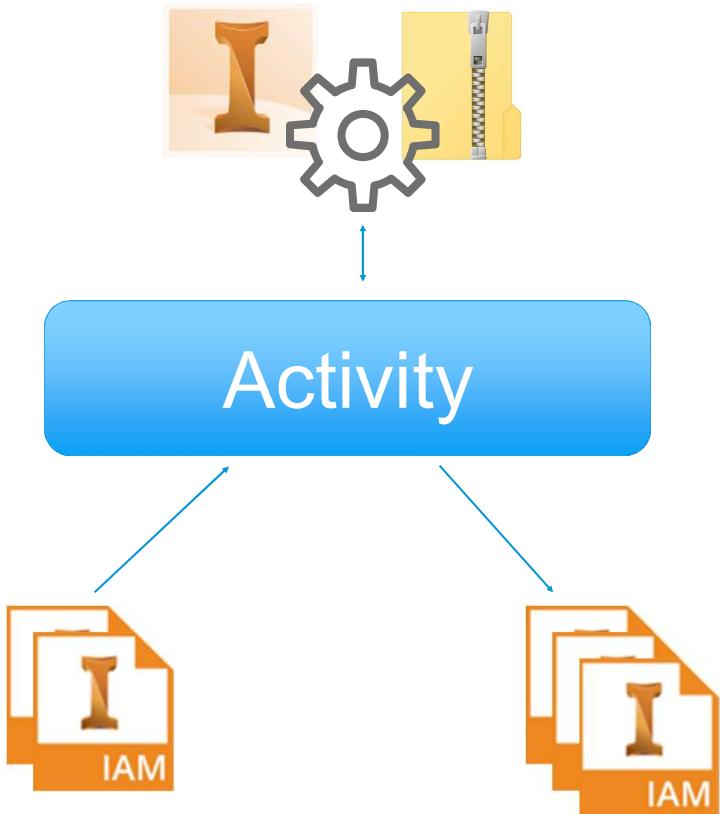
Inventor Activities

Describes:

- Which **AppBundle** to use
- How many **input** files
- **Zip** files (data sets)
- **Type of Parameter**
- How many **outputs**

Can also be:

- **Shared** – By Forge ID or everyone
- **Aliased** – Production, Beta, Alpha
- **Versioned**



JS3 It would be great to have some real life example -

commented JSON?

Jirka Stejskal, 6/6/2018

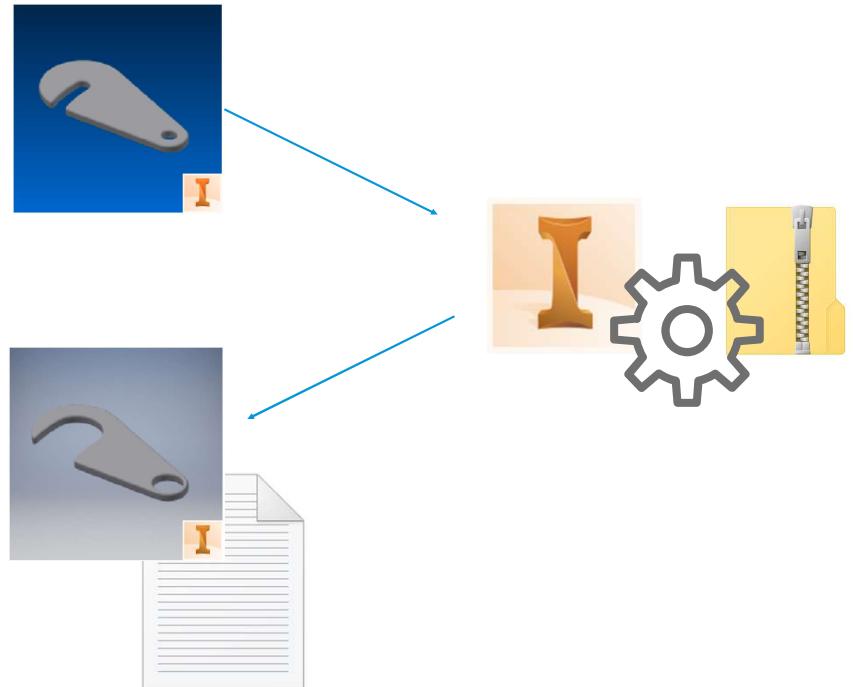
Inventor WorkItems

Describes:

- **Link(s)** to input file(s)
- Link to output file **location(s)**
- Parameter **values**

Results

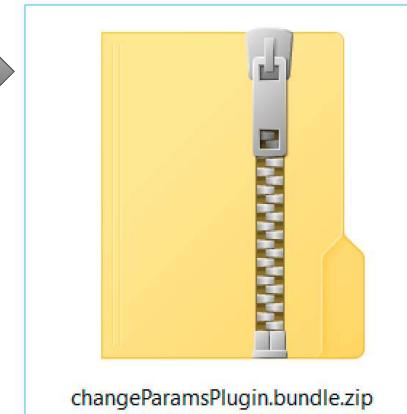
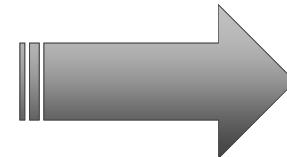
- **Status** – Complete, Error, Pending
 - **Polling or Callback**
- **Log file** – Debug output, error handling



Inventor AppBundle Example

```
\---changeParamsPlugin.bundle
    | PackageContents.xml
    |
    \---Contents
        changeParamsPlugin.dll
        changeParamsPlugin.Inventor.addin
```

```
Platform="Inventor"
ModuleName=
    "./Contents/changeParamsPlugin.Inventor.addin"
```



changeParamsPlugin.bundle.zip

Inventor Engines



- Multiple Versions
- Run jobs in parallel
- Optimized for Inventor workflows
- iLogic and translator plugins loaded
- Custom plug-ins
- No license needed, just pay for use



Runs in a “sandbox”

- One job per worker, nothing shared with anyone else
- Protects your code, protects our infrastructure
- No access outside of working directory

Quotas and Limits

- Input and Output file size
- Job duration

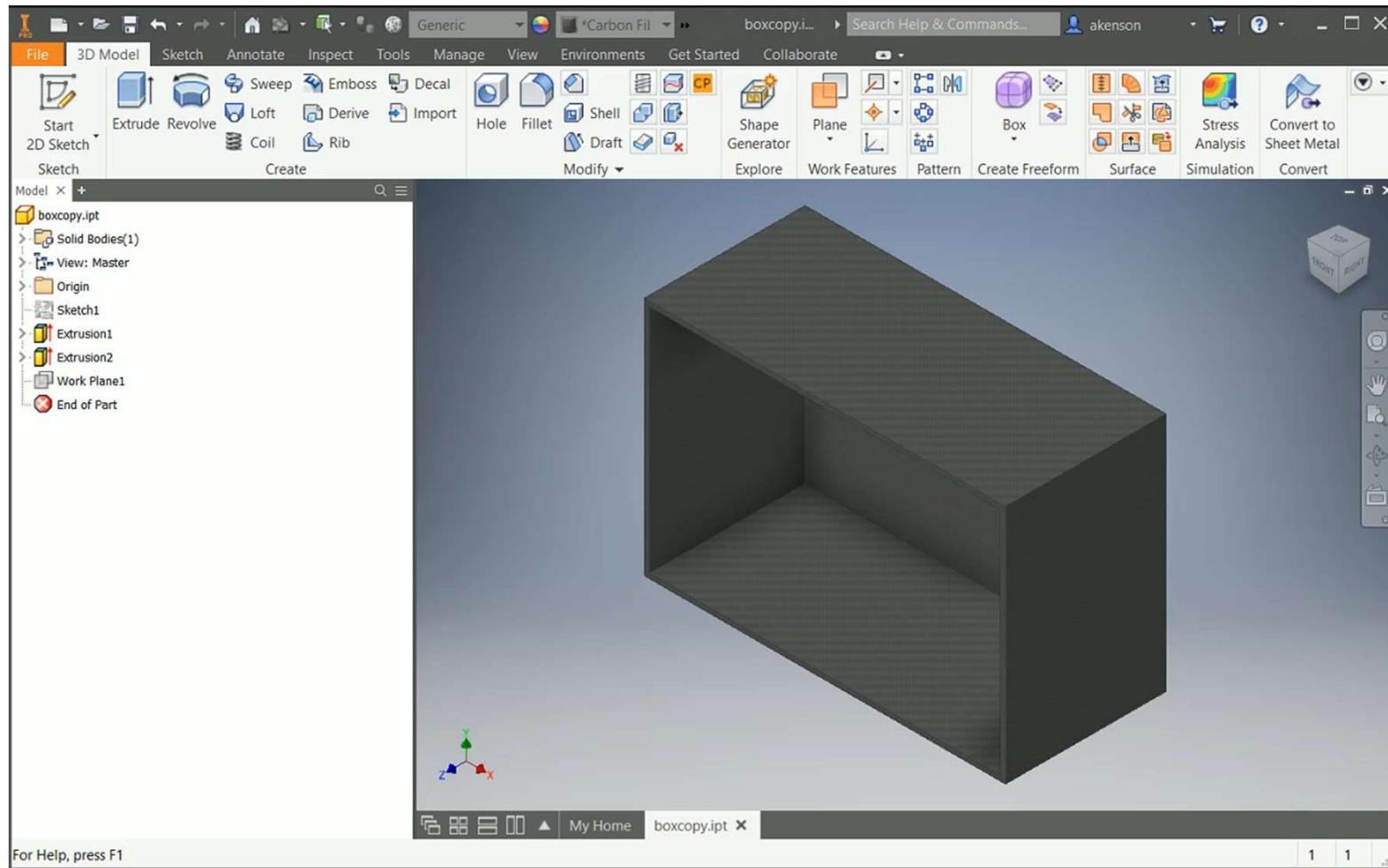


- Limited Excel
- No sticky sessions
- No Application Add-ins (Frame Generator, T&P...)
- No ETO

Debug locally before uploading



Simple Add-in example: “Change Param”



“Change Param” source Github

```
$ git clone https://github.com/akenson/change-param-addin.git
```

change-param-addin

This Add-in is intened as a very simple Inventor add-in to show how to take an existing Inventor Add-in and turn it into an AppBundle as a learning excersize for AU 2018.

1. Start Visual Studio 2017 or later as Administrator
2. Open TurnBlueAddIn.sln in Visual Studio
3. Build solution
4. Copy Autodesk.ChangeParam.Inventor.addin to %PROGRAMDATA%/Autodesk/Inventor <version>/Addins
5. Copy the ouptut dll in bin//ChangeParamAddIn.dll that you built in step 2 to %PROGRAMFILES%/Autodesk/Inventor <version>/Bin
6. Run Inventor, you should see a Message Box stating that the Change Param add-in is enabled.
7. Open a Part
8. Got to the Model Tab and Modify Pane.
9. Click on the "Change Param" button and it should change the "height" parameter to "1"

Get the “Debug Local” project from Github

```
$ git clone https://github.com/Developer-Autodesk/design.automation.inventor-csharp-basics.git
```

The screenshot shows the GitHub repository page for 'design.automation.inventor-csharp-basics'. The repository is owned by 'Developer-Autodesk' and has 11 stars, 2 forks, and 1 issue. It contains 9 commits, 3 branches, and 1 release. The repository is licensed under MIT. The commit history includes several initial commits and a note about adding double quotes. The README.md file is visible at the bottom.

This sample is .NET console app and demonstrates how one can process Inventor Assemblies or Parts on Design Automation.

design-automation csharp data-management inventor

9 commits 3 branches 1 release 2 contributors MIT

Branch: master ▾ New pull request Create new file Upload files Find file Clone or download ▾

File	Description	Time
pelakm and vasicem INVGEN-22618 added double quotes (#6)	Latest commit dc6deef 3 days ago	
Docs	INVGEN-21355: added developer notes	a month ago
Solution	INVGEN-22618 added double quotes (#6)	3 days ago
.gitignore	Initial commit.	a month ago
LICENSE	Initial commit.	a month ago
README.md	Added a new project "debugPluginLocally" (#3)	26 days ago
thumbnail.png	Initial commit.	a month ago

README.md

ChangeParamAddIn - Microsoft Visual Studio (Administrator)

File Edit View VAssistX Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU Start

Solution Explorer StandardAddInServer.cs ChangeParamButton.cs

ChangeParamAddIn

```
6
7     namespace ChangeParamAddIn
8     {
9         /// <summary>
10        /// ChangeParamButton class
11        /// </summary>
12        internal class ChangeParamUI
13        {
14            private ChangeParamButton m_ChangeParamButton;
15
16            //user interface event
17            private UserInterfaceEvents m_userInterfaceEvents;
18
19            Inventor.Application m_inventorApplication;
20
21            public ChangeParamUI()
22            {
23            }
24
25
26            public void InitUI(AddInServer addIn, bool firstTime, Inventor.Application inventorApplication)
27            {
28                m_inventorApplication = inventorApplication;
29                Button.InventorApplication = m_inventorApplication;
30
31                //load image icons for UI items
32                Icon ChangeParamIcon = new Icon(this.GetType(), "ChangeParam.ico");
33
34                //retrieve the GUID for this class
35                GuidAttribute addInCLSID;
36                addInCLSID = (GuidAttribute)GuidAttribute.GetCustomAttribute(typeof(AddInServer), typeof(GuidAttribute));
37                string addInCLSIDString;
38                addInCLSIDString = "{" + addInCLSID.Value + "}";
39
40                m_ChangeParamButton = new ChangeParamButton(addIn,
```

VA View VA Outline Solution Explorer Team Explorer 100 %

This item does not support previewing

Ln 42 Col 56 Ch 56 INS ↑ 0 ↪ 0 change-param-addin master

ChangeParamAddIn - Microsoft Visual Studio (Administrator)

File Edit View VAssistX Project Build Debug Team Tools Test Analyze Window Help

Quick Launch (Ctrl+Q) Andrew Akenson AA

Solution Explorer StandardAddInServer.cs ChangeParamButton.cs

Search Solution Explorer (Ctrl+F)

ChangeParamAddIn (2 projects)

- ChangeParamAddIn
 - References
 - AssemblyInfo.cs
 - Button.cs
 - ChangeParam.ico
 - ChangeParamButton.cs
 - StandardAddInServer.cs
 - debugPluginLocally
 - Properties
 - References
 - Analyzers
 - Autodesk.Inventor.Interop
 - Microsoft.CSharp
 - System
 - System.Core
 - System.Data
 - System.Data.DataSetExtensions
 - System.Net.Http
 - System.Xml
 - System.Xml.Linq

StandardAddInServer.cs

```
6  namespace ChangeParamAddIn
7  {
8      /// <summary>
9      /// ChangeParamButton class
10     /// </summary>
11     internal class ChangeParamUI
12     {
13         private ChangeParamButton m_ChangeParamButton;
14
15         //user interface event
16         private UserInterfaceEvents m_userInterfaceEvents;
17
18         Inventor.Application m_inventorApplication;
19
20         public ChangeParamUI()
21         {
22
23         }
24
25
26         public void InitUI(AddInServer addIn, bool firstTime, Inventor.Application inventorApplication)
27         {
28             m_inventorApplication = inventorApplication;
29             Button.InventorApplication = m_inventorApplication;
30
31             //load image icons for UI items
32             Icon ChangeParamIcon = new Icon(this.GetType(), "ChangeParam.ico");
33
34             //retrieve the GUID for this class
35             GuidAttribute addInCLSID;
36             addInCLSID = (GuidAttribute)GuidAttribute.GetCustomAttribute(typeof(AddInServer), typeof(GuidAttribute));
37             string addInCLSIDString;
38             addInCLSIDString = "{" + addInCLSID.Value + "}";
39
40             m_ChangeParamButton = new ChangeParamButton(addIn,
```

VA View VA Outline Solution Explorer Team Explorer 100 %

Ready ↑ 0 ↗ 1 change-param-addin master

ChangeParamAddIn - Microsoft Visual Studio (Administrator)

File Edit View VAssistX Project Build Debug Team Tools Test Analyze Window Help

Quick Launch (Ctrl+Q) Andrew Akenson AA

Solution Explorer

Search Solution Explorer (Ctrl+F)

Solution 'ChangeParamAddIn' (2 projects)

- ChangeParamAddIn
 - References
 - Analyzers
 - autodesk.inventor.interop
 - Microsoft.VisualBasic.Compatibility
 - stdole
 - System
 - System.Data
 - System.Drawing
 - System.Windows.Forms
 - System.XML
 - AssemblyInfo.cs
 - Button.cs
 - ChangeParam.ico
 - ChangeParamButton.cs
 - StandardAddInServer.cs
 - debugPluginLocally
 - Properties
 - References
 - Analyzers
 - Autodesk.Inventor.Interop
 - Microsoft.CSharp
 - System
 - System.Core
 - System.Data
 - System.Data.DataSetExtensions
 - System.Net.Http
 - System.Xml
 - System.Xml.Linq
 - inputFiles
 - App.config
 - InventorConnector.cs
 - Program.cs

StandardAddInServer.cs

```
1  using System;
2  using System.Runtime.InteropServices;
3  using System.Drawing;
4  using System.Windows.Forms;
5  using Inventor;
6  using Microsoft.Win32;
7
8  namespace ChangeParamAddIn
9  {
10     /// <summary>
11     /// This is the primary AddIn Server class that implements the ApplicationAddInServer interface
12     /// that all Inventor AddIns are required to implement. The communication between Inventor and
13     /// the AddIn is via the methods on this interface.
14     /// </summary>
15
16     [GuidAttribute("B99DB61B-F61E-4A56-AE2C-3FB608A2547D")]
17     public class AddInServer : Inventor.ApplicationAddInServer
18     {
19         #region Data Members
20
21         //Inventor application object
22         private Inventor.Application m_inventorApplication;
23         private ChangeParamUI m_ChangeParamUI;
24
25         #endregion
26
27         #region
28         0 references | Andrew Akenson, 11 hours ago | 1 author, 1 change
29         public AddInServer()
30         {
31         }
32
33         #region ApplicationAddInServer Members
34
35         0 references | Andrew Akenson, 11 hours ago | 1 author, 1 change
36         public void Activate(Inventor.ApplicationAddInSite addInSiteObject, bool firstTime)
37         {
38         }
39     }
40 }
```

VA View VA Outline Solution Explorer Team Explorer 100 % Ln 23 Col 9 Ch 9 INS ↑ 0 ↗ 2 ⌂ change-param-addin Y master

ChangeParamAddIn - Microsoft Visual Studio (Administrator)

File Edit View VAssistX Project Build Debug Team Tools Test Analyze Window Help

Quick Launch (Ctrl+Q) Andrew Akenson AA

Solution Explorer

Search Solution Explorer (Ctrl+F)

Solution 'ChangeParamAddIn' (2 projects)

- ChangeParamAddIn
 - References
 - Analyzers
 - autodesk.inventor.interop
 - Microsoft.VisualBasic.Compatibility
 - stdole
 - System
 - System.Data
 - System.Drawing
 - System.Windows.Forms
 - System.XML
 - AssemblyInfo.cs
 - StandardAddInServer.cs
- debugPluginLocally
 - Properties
 - References
 - Analyzers
 - Autodesk.Inventor.Interop
 - Microsoft.CSharp
 - System
 - System.Core
 - System.Data
 - System.Data.DataSetExtensions
 - System.Net.Http
 - System.Xml
 - System.Xml.Linq
 - inputFiles
 - App.config
- InventorConnector.cs
- Program.cs

StandardAddInServer.cs

```
→ ChangeParamAddIn.AddInServer → public class AddInServer : Inventor.ApplicationAddInServer
  ↗ ChangeParamAddIn
    ↗ ChangeParamAddIn.AddInServer
      ↗ m_inventorApplication

1  using System;
2  using System.Runtime.InteropServices;
3  using System.Drawing;
4  using System.Windows.Forms;
5  using Inventor;
6  using Microsoft.Win32;
7
8  namespace ChangeParamAddIn
9  {
10  	/// <summary>
11  	/// This is the primary AddIn Server class that implements the ApplicationAddInServer interface
12  	/// that all Inventor AddIns are required to implement. The communication between Inventor and
13  	/// the Addin is via the methods on this interface.
14  	/// </summary>
15
16  [GuidAttribute("B99DB61B-F61E-4A56-AE2C-3FB608A2547D")]
17  1 reference | Andrew Akenson, 11 hours ago | 1 author, 1 change
18  public class AddInServer : Inventor.ApplicationAddInServer
19  {
20  	#region Data Members
21
22  	//Inventor application object
23  	private Inventor.Application m_inventorApplication;
24  	//private ChangeParamUI m_ChangeParamUI;
25
26  	#endregion
27
28  0 references | Andrew Akenson, 11 hours ago | 1 author, 1 change
29  public AddInServer()
30  {
31  }
32
33  #region ApplicationAddInServer Members
34
35  0 references | Andrew Akenson, 11 hours ago | 1 author, 1 change
36  public void Activate(Inventor.ApplicationAddInSite addInSiteObject, bool firstTime)
37  {
38  }
```

VA View VA Outline Solution Explorer Team Explorer 100 %

Ln 20 Col 9 Ch 3 INS ↑ 0 ↲ 3 ⌂ change-param-addin Y master

ChangeParamAddIn - Microsoft Visual Studio (Administrator)

File Edit View VAssistX Project Build Debug Team Tools Test Analyze Window Help

Any CPU Start Stop Debug

Solution Explorer StandardAddInServer.cs

Search Solution Explorer (Ctrl+Shift+F)

ChangeParamAddIn (2 projects)

ChangeParamAddIn

- References
 - Analyzers
 - autodesk.inventor.interop
 - Microsoft.VisualBasic.Compatibility
 - stdole
 - System
 - System.Data
 - System.Drawing
 - System.Windows.Forms
 - System.XML
- AssemblyInfo.cs
- StandardAddInServer.cs

debugPluginLocally

- Properties
- References
 - Analyzers
 - Autodesk.Inventor.Interop
 - Microsoft.CSharp
 - System
 - System.Core
 - System.Data
 - System.Data.DataSetExtensions
 - System.Net.Http
 - System.Xml
 - System.Xml.Linq

inputFiles

App.config

InventorConnector.cs

Program.cs

StandardAddInServer.cs

```
61 //the Deactivate method is called by Inventor when the AddIn is unloaded
62 //the AddIn will be unloaded either manually by the user or
63 //when the Inventor session is terminated
64
65 try
66 {
67     //release inventor Application object
68     Marshal.ReleaseComObject(m_inventorApplication);
69     m_inventorApplication = null;
70
71     GC.WaitForPendingFinalizers();
72     GC.Collect();
73 }
74 catch(Exception e)
75 {
76     MessageBox.Show(e.ToString());
77 }
78
79
80 public void ChangeParam(Document doc)
81 {
82     // Get the active document
83     //Document doc = m_inventorApplication.ActiveDocument;
84     if (doc.DocumentType == DocumentTypeEnum.kPartDocumentObject)
85     {
86         PartDocument partDoc = (PartDocument)doc;
87         try
88         {
89             Parameters docParams = partDoc.ComponentDefinition.Parameters;
90             Parameter param = docParams["height"];
91             param.Expression = "1";
92             partDoc.Update();
93         }
94         catch (Exception e)
95         {
96             MessageBox.Show("error setting param: " + e.Message);
97         }
98     }
99 }
```

VA View VA Outline Solution Explorer Team Explorer 100 %

Ln 67 Col 54 Ch 42 INS ↑ 0 ↻ 3 ⌂ change-param-addin ⌂ master

ChangeParamAddIn - Microsoft Visual Studio (Administrator)

File Edit View VAssistX Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU Start Stop Task List Solution Explorer

Quick Launch (Ctrl+Q) Andrew Akenson AA

Solution Explorer

Search Solution Explorer (Ctrl+F)

Diagnostic Tools

Toolbox Data Sources

Program.cs StandardAddInServer.cs

```
debugPluginLocally.Program → private static void DebugSamplePlugin(InventorServer app)
debugPluginLocally → debugPluginLocally.Program → DebugSamplePlugin(InventorServer app)

62     catch (IOException e)
63     {
64         Console.WriteLine("The specified file is in use. It might be open by Inventor");
65         return;
66     }
67
68     // create a copy
69     System.IO.File.Copy(boxPath, boxPathCopy);
70
71     // open box.ipt by Inventor
72     Document doc = app.Documents.Open(boxPathCopy);
73
74     // get params.json absolute path
75     string paramsPath = System.IO.Path.Combine(projectdir, @"inputFiles\", "params.json");
76
77     // create a name value map
78     Inventor.NameValueMap map = app.TransientObjects.CreateNameValueMap();
79
80     // add parameters into the map, do not change "_1". You may add more parameters "_2", "_3"...
81     map.Add("_1", paramsPath);
82
83     // create an instance of samplePlugin
84     //samplePlugin.SampleAutomation plugin = new samplePlugin.SampleAutomation(app);
85     ChangeParamAddIn.AddInServer plugin = new ChangeParamAddIn.AddInServer(app);
86
87     // run the plugin
88     plugin.RunWithArguments(doc, map);
89
90 }
91
92 }
```

VA View VA Outline Solution Explorer Team Explorer 100 %

Ready Ln 89 Col 9 Ch 9 INS ↑ 0 ↗ 3 ⌂ change-param-addin Y master

A little more work to do on the environment...

```
[11/09/2018 23:29:31] Starting work item b1f8908ced664131b6746c1750df9f80
[11/09/2018 23:29:31] Start download phase.
[11/09/2018 23:29:31] Start downloading file https://s3-us-west-2.amazonaws.com/inventorio-
prod/documentation/APIBasics/Box.upt.
[11/09/2018 23:29:31] Start preparing AppPackage ChangeParam.
[11/09/2018 23:29:31] Download bits and install app to local cache.
[11/09/2018 23:29:31] Embedded resource [{"height":"16 in", "width":"10 in"}] is saved as file:
T:\Aces\Jobs\b1f8908ced664131b6746c1750df9f80\params.json.
[11/09/2018 23:29:31] End downloading file https://s3-us-west-2.amazonaws.com/inventorio-
prod/documentation/APIBasics/Box.upt. 117760 bytes have been written to
T:\Aces\Jobs\b1f8908ced664131b6746c1750df9f80\Box.upt.
[11/09/2018 23:29:31] End download phase.
[11/09/2018 23:29:32] Error: Failed to prepare app package(s).
[11/09/2018 23:29:32] Error: An unexpected error happened during phase Downloading of job.
[11/09/2018 23:29:32] Job finished with result FailedEnvironmentSetup
```

ChangeParamAddIn - Microsoft Visual Studio (Administrator)

File Edit View VAssistX Project Build Debug Team Tools Test Analyze Window Help

Any CPU Start

Solution Explorer StandardAddInServer.cs

ChangeParamAddIn ChangeParamAddIn.AddInServer m_inventorApplication

```
11  /// This is the primary AddIn Server class that implements the ApplicationAddInServer interface
12  /// that all Inventor AddIns are required to implement. The communication between Inventor and
13  /// the AddIn is via the methods on this interface.
14  /// </summary>
15
16 [GuidAttribute("B99DB61B-F61E-4A56-AE2C-3FB608A2547D")]
17 4 references | Andrew Akenson, 12 hours ago | 1 author, 1 change
18 public class AddInServer : Inventor.ApplicationAddInServer
19 {
20     #region Data Members
21
22     //Inventor application object
23     private InventorServer m_inventorApplication;
24
25     //private ChangeParamUI m_ChangeParamUI;
26
27     #endregion
28
29     0 references | Andrew Akenson, 12 hours ago | 1 author, 1 change
30     public AddInServer()
31     {
32     }
33
34     1 reference | 0 changes | 0 authors, 0 changes
35     public AddInServer(InventorServer app)
36     {
37         m_inventorApplication = app;
38     }
39
40     #region ApplicationAddInServer Members
41
42     0 references | Andrew Akenson, 12 hours ago | 1 author, 1 change
43     public void Activate(Inventor.ApplicationAddInSite addInSiteObject, bool firstTime)
44     {
45         try
46         {
47             //The Activate method is called by Inventor when it loads the addin
48         }
49     }
50 }
```

VA View VA Outline Solution Explorer Team Explorer 100 %

Ready Ln 16 Col 60 Ch 57 INS ↑ 0 ⌂ 4 ⌂ change-param-addin ⌂ master

Still some UI to remove...

```
[11/10/2018 00:42:48] InventorCoreConsole.exe Information: 0 : Activating plug-in: ChangeParamAddIn
[11/10/2018 00:42:48] InventorCoreConsole.exe Information: 0 : Deactivating plug-in: ChangeParamAddIn
[11/10/2018 00:42:48] Showing a modal dialog box or form when the application is not running in
UserInteractive mode is not a valid operation. Specify the ServiceNotification or DefaultDesktopOnly style to
display a notification from a service application.
[11/10/2018 00:42:48]
[11/10/2018 00:42:48] Process exit code: -1
[11/10/2018 00:42:48]
[11/10/2018 00:42:48] End Inventor Core Engine standard output dump.
[11/10/2018 00:42:48] Error: Inventor Core Engine Core Console output contains error(s).
[11/10/2018 00:42:48] End script phase.
[11/10/2018 00:42:48] Error: An unexpected error happened during phase CoreEngineExecution of job.
[11/10/2018 00:42:48] Job finished with result FailedExecution
```

ChangeParamAddIn - Microsoft Visual Studio (Administrator)

File Edit View VAssistX Project Build Debug Team Tools Test Analyze Window Help

Any CPU Start Debug

Solution Explorer

Search Solution Explorer (Ctrl+.)

Solution 'ChangeParamAddIn' (2 projects)

- ChangeParamAddIn
 - References
 - AssemblyInfo.cs
 - StandardAddInServer.cs
 - debugPluginLocally

StandardAddInServer.cs

C:\public_repos\change-param-addin\StandardAddInServer.cs

ChangeParamAddIn

```
1  using System;
2  using System.Runtime.InteropServices;
3  using System.Drawing;
4  using System.Windows.Forms;
5  using Inventor;
6  using Microsoft.Win32;
7
8  namespace ChangeParamAddIn
9  {
10     /// <summary>
11     /// This is the primary AddIn Server class that implements the ApplicationAddInServer interface
12     /// that all Inventor AddIns are required to implement. The communication between Inventor and
13     /// the AddIn is via the methods on this interface.
14     /// </summary>
15
16     [GuidAttribute("B99DB61B-F61E-4A56-AE2C-3FB608A2547D")]
17     [ComVisible(true)]
18     public class AddInServer : Inventor.ApplicationAddInServer
19     {
20         #region Data Members
21
22         //Inventor application object
23         private InventorServer m_inventorApplication;
24         private AddInServer m_automation;
25
26         //private ChangeParamUI m_ChangeParamUI;
27
28         #endregion
29
30
31         public AddInServer()
32         {
33
34             2 references | Andrew Akenson, 12 hours ago | 1 author, 1 change
35             public AddInServer(InventorServer app)
```

VA View VA Outline Solution Explorer Team Explorer 100 %

Ln 7 Col 1 Ch 1 INS ↑ 0 ↗ 4 ⌂ change-param-addin master

Make your AppBundle



1

Create a new folder ending in “.bundle”

OSDisk (C:) > public_repos > change-param-addin

<input type="checkbox"/>	Name	Date modified	Type
	bin	11/10/2018 9:23 A...	File folder
	obj	11/10/2018 10:19 ...	File folder
<input checked="" type="checkbox"/>	ChangeParam.bundle	11/10/2018 10:23 ...	File folder
	.gitignore	11/9/2018 8:38 PM	Text Document
	bom int	11/9/2018 8:38 PM	Autodesk Inventor

2

Copy template PackageContents.xml

C# Sample PackageContents.xml

OSDisk (C:) > public_repos > design.automation.inventor-csharp-basics > Solution > samplePlugin

<input type="checkbox"/>	Name	Date modified	Type	Size
	bin	11/9/2018 8:33 AM	File folder	
	obj	11/9/2018 8:33 AM	File folder	
	Properties	11/8/2018 4:31 PM	File folder	
	PackageContents.xml	11/8/2018 4:31 PM	XML Document	1 KB
	packages.config	11/6/2018 4:31 PM	XNA Game Studio...	1 KB
	PluginServer.cs	11/8/2018 4:31 PM	Visual C# Source F...	3 KB
	README.md	11/8/2018 4:31 PM	MD File	1 KB
	SampleAutomation.cs	11/8/2018 4:31 PM	Visual C# Source F...	16 KB
	samplePlugin.csproj	11/8/2018 4:31 PM	Visual C# Project F...	6 KB
	samplePlugin.Inventor.addin	11/8/2018 4:31 PM	ADDIN File	1 KB
	samplePlugin.X.manifest	11/8/2018 4:31 PM	MANIFEST File	1 KB

COPY

New ".bundle" Folder

OSDisk (C:) > public_repos > change-param-addin > ChangeParam.bundle

<input type="checkbox"/>	Name	Date modified	Type	Size
<input checked="" type="checkbox"/>	PackageContents.xml	11/10/2018 8:28 A...	XML Document	1 KB

3 Update the PackageContents.xml

```
<ApplicationPackage SchemaVersion="1.0" Version="1.0" ProductCode=
'{B99DB61B-F61E-4A56-AE2C-3FB608A2547D}' Name="ChangeParam" Description="AU2018 Demo Plugin" Author=
"Andrew Akenson">
  <CompanyDetails Name="Autodesk, Inc" Phone="415.555.5555" Url="www.autodesk.com" Email=
"inventoronforge@autodesk.com" />
  <Components>
    <!-- For Inventor Engine, "Platform" attribute must be "Inventor" -->
    <RuntimeRequirements OS="Win64" Platform="Inventor" />
    <!-- For Inventor Plug-in, the "Module" attribute must point to the .addin manifest file.
    -->
    <ComponentEntry LoadOnAutoCADStartup="False" LoadOnCommandInvocation="False"
      AppDescription="AU2018 demo App Package."
      ModuleName=".Content/Autodesk.ChangeParam.Inventor.addin" AppName="ChangeParam"/>
  </Components>
  <EnvironmentVariables>
  </EnvironmentVariables>
</ApplicationPackage>
```

4

Add content to the AppBundle

Create "Contents" folder

OSDisk (C:) > public_repos > change-param-addin > ChangeParam.bundle

<input type="checkbox"/>	Name	Date modified	Type
<input checked="" type="checkbox"/>	Contents	11/10/2018 10:39 ...	File folder
<input type="checkbox"/>	PackageContents.xml	11/10/2018 10:34 ...	XML Document

Add your assembly and .addin file

OSDisk (C:) > public_repos > change-param-addin > ChangeParam.bundle > Contents

<input type="checkbox"/>	Name	Date modified	Type	Size
<input type="checkbox"/>	Autodesk.ChangeParam.Inventor.addin	11/9/2018 8:55 PM	ADDIN File	1 KB
<input type="checkbox"/>	ChangeParamAddIn.dll	11/10/2018 10:19 ...	Application extens...	92 KB

5

Update your .addin file

```
<!-- Type attribute is same as Type registry key (Standard, Translator, Plugin (Server only)
-->
<Addin Type="Plugin">
<ClassId>{B99DB61B-F01E-4A56-AE2C-3FB608A2547D}</ClassId>
<ClientId>{B99DB61B-F01E-4A56-AE2C-3FB608A2547D}</ClientId>

<!-- Both of the following fields should be translated. NO OTHER FIELDS SHOULD BE TRANSLATED!
-->
<DisplayName>ChangeParamAddIn</DisplayName>
<Description>ChangeParamAddIn</Description>

<!-- Assumes that SimpleAddIn.dll is underneath Inventor\bin -->
<Assembly>ChangeParamAddIn.dll</Assembly>

<SupportedSoftwareVersionGreater Than>17..</SupportedSoftwareVersionGreater Than>
<LoadOnStartUp>1</LoadOnStartUp>
<Hidden>0</Hidden>
</Addin>
```

6 Zip it up!

Name	Date modified	Type	Size
<input checked="" type="checkbox"/> ChangeParam.bundle.zip	11/10/2018 10:47 ...	Compressed (zipped) Folder	30 KB
<input type="checkbox"/> StandardAddInServer.cs	11/10/2018 10:19 ...	Visual C# Source File	4 KB
<input type="checkbox"/> AssemblyInfo.cs	11/10/2018 10:05 ...	Visual C# Source File	3 KB
<input type="checkbox"/> ChangeParamAddIn.csproj	11/10/2018 9:23 A...	Visual C# Project File	7 KB
<input type="checkbox"/> ChangeParamAddIn.sln	11/10/2018 9:23 A...	Visual Studio Solution	2 KB
<input type="checkbox"/> ChangeParamButton.cs	11/10/2018 9:22 A...	Visual C# Source File	5 KB
<input type="checkbox"/> ChangeParam.ico	11/9/2018 8:58 PM	Icon	2 KB
<input type="checkbox"/> Autodesk.ChangeParam.Inventor.addin	11/9/2018 8:55 PM	ADDIN File	1 KB
<input type="checkbox"/> Button.cs	11/9/2018 8:43 PM	Visual C# Source File	4 KB
<input type="checkbox"/> .gitignore	11/9/2018 8:38 PM	Text Document	1 KB

Test it out in Design Automation



A screenshot of a web browser window displaying the Autodesk Forge "My Apps" interface. The URL in the address bar is <https://forge.autodesk.com/myapps>. The page has a dark header with the Autodesk Forge logo and navigation links: Success Stories, Solutions, Getting Started, Documentation, Community, Support, and Pricing. On the right side of the header is a user profile icon. Below the header is a blue banner with the text "My Apps". To the right of the banner is a large orange button labeled "CREATE APP" with a counter showing "0" and a right-pointing arrow. The main content area is white and displays the message "You don't have any apps yet" in bold, followed by the subtext "Apps that you have created will appear here". A vertical scrollbar is visible on the right edge of the content area.

So close, but need to save the output...

```
[11/10/2018 19:26:22] InventorCoreConsole.exe Information: 0 : Opening document:  
T:\Aces\Jobs\7b147563930242b591e145c5c4e12fa6\Box.ipt  
[11/10/2018 19:26:22] Process exit code: 0  
[11/10/2018 19:26:22] End Inventor Core Engine standard output dump.  
[11/10/2018 19:26:22] End script phase.  
[11/10/2018 19:26:22] Start upload phase.  
[11/10/2018 19:26:22] Error: Non-optional output [Result.ipt] is missing .  
[11/10/2018 19:26:22] Error: An unexpected error happened during phase Publishing of job.  
[11/10/2018 19:26:22] Job finished with result FailedMissingOutput
```

ChangeParamAddIn - Microsoft Visual Studio (Administrator)

File Edit View VAssistX Project Build Debug Team Tools Test Analyze Window Help

Any CPU Start Debug Diagnostic Tools

Andrew Akenson AA

Solution Explorer

Search Solution Explorer (Ctrl + F)

Solution 'ChangeParamAddIn' (2 projects)

- ChangeParamAddIn
 - References
 - AssemblyInfo.cs
 - StandardAddInServer.cs
- debugPluginLocally

StandardAddInServer.cs

```
public void RunWithArguments(Document doc, NameValueMap map)
{
    Trace.TraceInformation(">>> ChangeParam for doc: " + doc.DisplayName);
    ChangeParam(doc);
}

1 reference | 0 changes | 0 authors, 0 changes
public void ChangeParam(Document doc)
{
    // Get the active document
    //Document doc = m_inventorApplication.ActiveDocument;
    if (doc.DocumentType == DocumentTypeEnum.kPartDocumentObject)
    {
        PartDocument partDoc = (PartDocument)doc;
        try
        {
            Parameters docParams = partDoc.ComponentDefinition.Parameters;
            Parameter param = docParams["height"];
            param.Expression = "1";
            partDoc.Update();
        }
        catch (Exception e)
        {
            Trace.TraceInformation("error setting param: " + e.Message);
        }
    }
}

0 references | Andrew Akenson, 14 hours ago | 1 author, 1 change
public void ExecuteCommand(int CommandID)
{
    //this method was used to notify when an AddIn command was executed
    //the CommandID parameter identifies the command that was executed

    //Note: this method is now obsolete, you should use the new
    //ControlDefinition objects to implement commands, they have
    //their own event sinks to notify when the command is executed
}
```

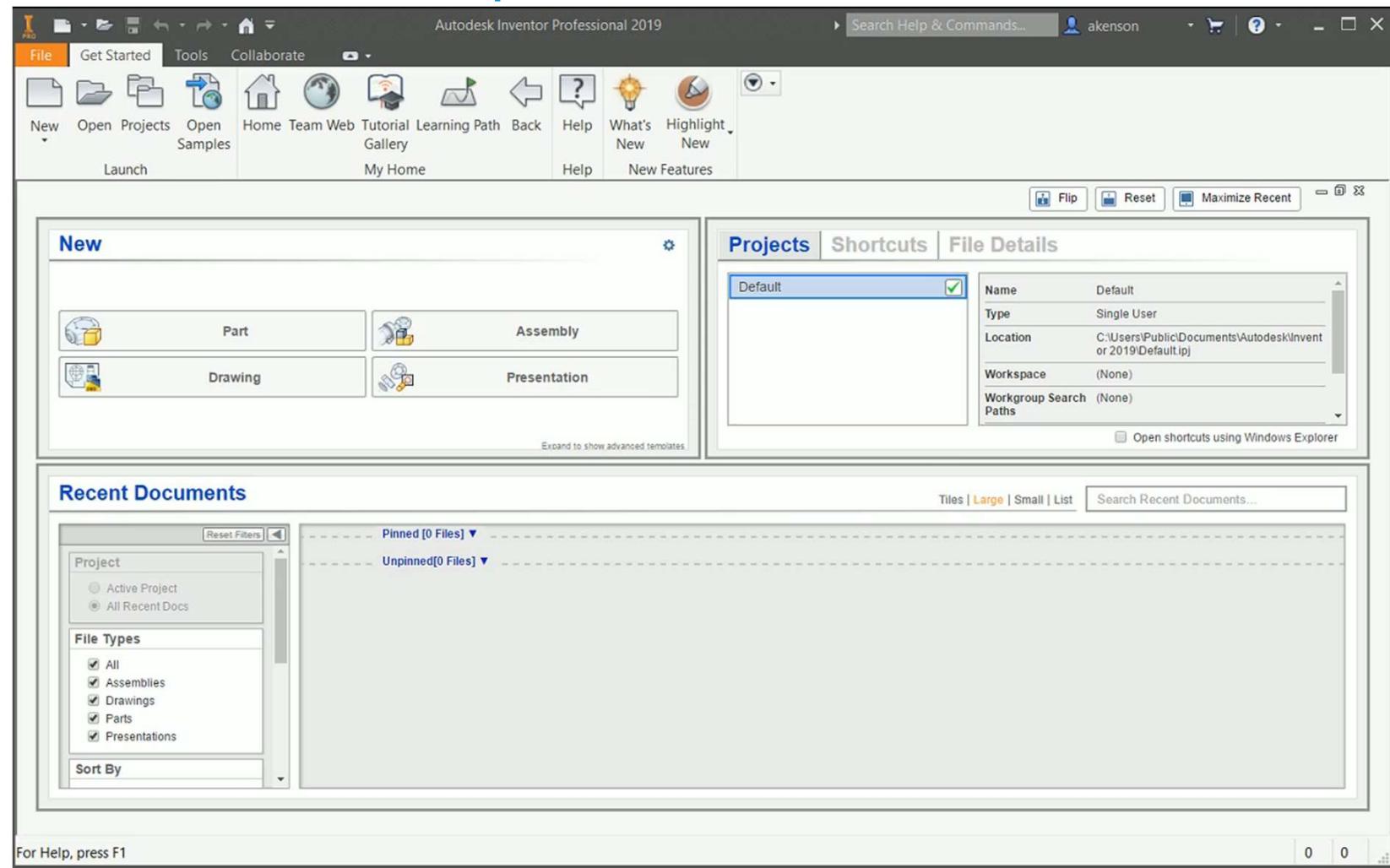
VA View VA Outline Solution Explorer Team Explorer

100 %

Item(s) Saved

Ln 114 Col 21 Ch 21 INS ↑ 0 ⌂ 8 change-param-addin master

Check out the Result.ipt!



Digging a bit deeper...



C# Sample Application and AppBundle

<https://github.com/Developer-Autodesk/design.automation.inventor-csharp-basics>

```
LogTrace("Zipping up updated Assembly.");

// assembly lives in own folder under WorkingDir. Get the WorkingDir
var workingDir = Path.GetDirectoryName(docDir);
var fileName = Path.Combine(workingDir, "Result.zip"); // the name must be in sync with OutputIam localName

if (File.Exists(fileName)) File.Delete(fileName);

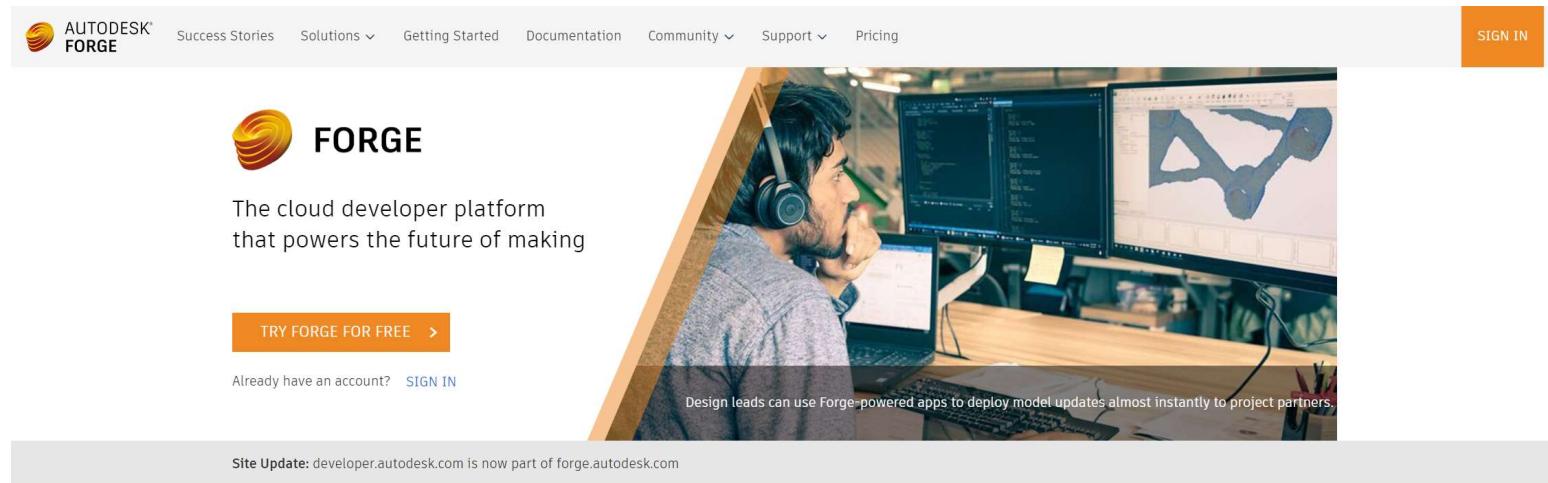
// start HeartBeat around ZipFile, it could be a long operation
using (new HeartBeat())
{
    ZipFile.CreateFromDirectory(Path.GetDirectoryName(pathName), fileName, CompressionLevel.Fastest, false)
}
```

Where to get started



General Resources

<https://forge.autodesk.com>



The screenshot shows the Autodesk Forge homepage. At the top, there's a navigation bar with links for Success Stories, Solutions, Getting Started, Documentation, Community, Support, and Pricing. On the far right of the bar is a "SIGN IN" button. Below the navigation bar, the Autodesk Forge logo is prominently displayed, featuring a stylized orange and yellow swirl icon followed by the word "FORGE". A large photograph of a man wearing headphones and working on a computer is centered on the page. To the left of the photo, the text "The cloud developer platform that powers the future of making" is written. Below the photo is a call-to-action button labeled "TRY FORGE FOR FREE >". Underneath the button, there's a link for users who already have an account: "Already have an account? SIGN IN". A small note at the bottom of the page states: "Design leads can use Forge-powered apps to deploy model updates almost instantly to project partners." At the very bottom of the main content area, a grey banner contains the text "Site Update: developer.autodesk.com is now part of forge.autodesk.com".

Forge: the cloud developer platform from Autodesk

Forge gives companies the tools to develop custom, cloud-based software applications that connect workflows for manufacturing, media/entertainment, architecture, engineering, and construction.



BIM 360 API



VIEWER



DATA MANAGEMENT API



MODEL DERIVATIVE API



DESIGN AUTOMATION API



REALITY CAPTURE API

Build applications and custom integrations

Develop apps that connect construction workflows with the BIM 360 API. Integrate with, and extend the functionality of, the BIM 360 platform to do things like automate



Learn Forge

<https://forge.autodesk.com/LearnForge>

Type to search

Autodesk Forge

Home

- What is it?
- What's in this tutorial?

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
- Authorize
- List hubs & projects
- User information
- Show on Viewer

Run & Debug

Viewer extension

Basic skeleton

 Handling selection

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 across A360, Fusion, and the Object Storage Service.
- **Convert your design files to formats that unlock their potential:** Use the Model Derivative API to prepare files for the Viewer, extract geometries, retrieve metadata, and more, from over 50 different industry-standard formats.

What's in this tutorial?

- **Before you start coding:** create and activate your account.
- **Tools:** development tools you need to create a webapp that uses Autodesk Forge.
- **OAuth:** a few words about security and authentication.
- Step-by-step tutorials for:
 - **View your models:** upload & show 3D models on the web.
 - **View BIM 360 & Fusion models:** access & show BIM 360 and Fusion models on your own webapp.
- **Run & debug:** executing the code locally and tips & tricks.
- **Viewer extension:** add buttons and panels to Viewer.
- **Deployment:** AWS, Heroku and AppHarbor step-by-step deployment.

Ready to get started?

Forge on Github

<https://forge.autodesk.com/GitHub>

The screenshot shows the GitHub organization page for 'Forge Platform'. The page features a header with a search bar, pull requests, issues, marketplace, and explore links. Below the header, there's a logo for 'Forge Platform' and a brief description: 'Forge is a set of Autodesk APIs and services for software developers to build innovative cloud-powered apps.' A link to the developer documentation and support email is also provided.

The main content area displays five repository cards:

- bim360-csharp-issues**: Describes the Document Issues API for BIM 360 using the PushPin Viewer extension. It includes tags for jquery, csharp, viewer, data-management, net-core, and jstree. Stats: 5 stars, 4 forks, MIT license, updated 7 hours ago.
- forge-api-java-client**: Provides Java SDK to help easily integrate Forge REST APIs into the application. It includes tags for Java, viewer, csharp, model-derivative, nodejs, and autocad. Stats: 17 stars, 17 forks, Apache-2.0 license, updated 2 days ago.
- data.management-csharp-webhook**: Webhooks for Data Management API: Use a database to store refresh token and access files (on BIM 360) later. It includes tags for csharp, webhook, asp-net, data-management, mongolab, and net-core. Stats: 0 stars, 0 forks, MIT license, updated 3 days ago.
- oauth-walkthrough-2.legged.auth**: 2-Legged Authentication walkthrough: Learn how to use two-legged authentication. It includes tags for Javascript and MIT license. Stats: 0 stars, 0 forks, MIT license, updated 3 days ago.
- recap-walkthrough-photo.to.3d**: Described as 'Recap of the 2D Walkthrough Idea that Results in the API to process photos.'

On the right side, there are three sidebar boxes: 'Top languages' (JavaScript, C#, Java, HTML, CSS), 'Most used topics' (viewer, csharp, model-derivative, nodejs, autocad), and 'People' (0 members).

Find answers on StackOverflow

<https://forge.autodesk.com/Stack>

The screenshot shows the Autodesk Forge website with a blue header bar. The header includes the Autodesk Forge logo, navigation links for Success Stories, Solutions, Getting Started, Documentation, Community, Support, and Pricing, and a 'SIGN IN' button. Below the header is a large blue banner with the text 'Get Help'. The main content area has a white background. At the top of this area, there is a sub-header 'Find answers from StackOverflow' followed by a note: 'The Forge support team monitors StackOverflow for, and will answer, all questions marked with Forge-related tags. In many cases, members of the Forge developer community will also give you advice.' Below this note is a text input field with placeholder text 'Ask a question now!' and a 'TIPS' section containing the text 'Here's how to get better answers.' To the right of the input field is a search bar with a 'SEARCH' button. A dropdown menu is open over the search bar, listing various API documentation pages: 'About which API?', 'Authentication (OAuth)', 'Data Management API', 'Design Automation API' (which is highlighted in blue), 'Model Derivative API', 'Viewer', 'BIM 360 API', 'Reality Capture API', and 'None'. To the right of the search bar is a sidebar with sections for 'ABOUT', including links to 'About Forge', 'Pricing', 'Success Stories', 'Partners', 'Forge Fund', 'DevCon 2018', and 'DevCon 2017'. At the bottom of the page, there are three columns: 'FOLLOW FORGE' with links to Twitter and Facebook; 'SOLUTIONS' listing Data Management, Design Automation, Model Derivative & Viewer, Reality Capture, BIM 360, and Webhooks; and 'DOCUMENTATION' listing Authentication (OAuth), Data Management, Design Automation, Model Derivative, Viewer, Reality Capture, and BIM 360.

Join an Accelerator

<https://forge.autodesk.com/accelerator>

The screenshot shows the Autodesk Forge Accelerators landing page. At the top, there's a navigation bar with links for Home, Forge Accelerator, APIs, Upcoming Events, Resources, Apply, Contact, and a search icon. Below the navigation is a large blue header section with the title "Autodesk Forge Accelerators" and a subtitle "Fast track your cloud development". It also mentions "A special program to help developers leverage Autodesk Forge APIs to build next generation solutions". The main content area has two sections: "Autodesk Forge Accelerator Programs" on the left and "Expand your business with Autodesk Forge" on the right. The "Programs" section lists three locations: Tel Aviv, Israel; Moscow, Russia; and Boston, MA. The "Expand" section provides details about the accelerator programs. At the bottom, there's a section titled "Develop for the Cloud with Autodesk Forge" featuring three cards: "THE PROGRAM" (with a lightbulb icon), "FORGE TECHNOLOGIES" (with a cloud icon), and "HOW TO APPLY" (with a person icon).

AUTODESK®
FORGE

Home Forge Accelerator APIs Upcoming Events Resources Apply Contact

Autodesk Forge Accelerators

Fast track your cloud development

A special program to help developers leverage Autodesk Forge APIs to build next generation solutions

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

Tel Aviv, Israel – December 10-13, 2018

Moscow, Russia – January 21-25, 2019

Boston, MA – February 4-8, 2019

Contact us if you are interested in a future Accelerator.

Develop for the Cloud with Autodesk Forge

THE PROGRAM FORGE TECHNOLOGIES HOW TO APPLY

Forge Community

Community

- Twitter: @AutodeskForge
- Facebook @AdskForge
- YouTube: <https://forge.autodesk.com/YouTube>
- Blog: <https://forge.autodesk.com/blog>
- Design Automation for Inventor:
InventorOnForge@autodesk.com

More here at DevCon and AU



Please Visit - Forge Experts Available!

- Forge Village at DevCon
- Autodesk Answer Bar in the AU Exhibit Hall
- Forge Quad in the AU Exhibit Hall

Forge DevLab - Forge Experts Available!

- **Tuesday, November 13, 1:30 p.m. – 5:30 p.m. in Galileo 1002 (Level 1)**
- Drop-in workshop / informal lab for software developers of all skill levels - come and go as you please
- **Get one-to-one** help and advice on using the Forge APIs
- DevLab is open to all **Forge DevCon and Autodesk University attendees**
- Book a 30 minute slot in the Class Catalog (enter Forge DevLab in the search bar) or drop by
- Forge experts available all afternoon to answer your questions or help you get started

Classes here at DevCon and AU

Design Automation

- [AS226079 - Manage Revit Digital Content using Forge Design Automation API for Revit.](#)

iLogic and Inventor Add-ins

- [PM224007 - Taking It to the Next Level: Drawing Automation with Inventor](#)
- [IM227119 - Making the Leap from iLogic to Add-Ins—or Not?](#)
- [IM221410 - The Power of iLogic Design Automation: How Did We Get Here?](#)



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.

© 2018 Autodesk. All rights reserved.

