AUTODESK UNIVERSITY

PD20446-L

Configurator 360 Hands-on: Create and Deploy Your Parametric Designs to Anyone

Pierre Masson
Autodesk

Learning Objectives

- Learn how to upload a model in Configurator 360.
- Learn how to use various methods to create parametrically driven assemblies.
- Learn how to set your Configurator 360 environment.
- Learn how to deploy your design to colleagues or customers.

Description

In this hands-on lab, we will use Inventor software and Configurator 360 software to create a fully configurable model available in the cloud. We will start with an Inventor model and explore different strategies for driving it parametrically, like iLogic rules, forms, and deriving parts. We will fine-tune the model to configure it successfully in Configurator 360 software, and we will learn how to display user messages for customers. Finally, we will upload the model and make it ready for deployment everywhere. This session features Configurator 360 and Inventor Professional.

Your AU Expert(s)

Pierre Masson is currently employed as a Premium Support Specialist within Autodesk's Global services Department.

He works on various manufacturing products like Fusion 360, Inventor, ETO, C360, Autocad or Netfabb. He has also some knowledge with the simulation products like Moldflow, Nastran In-CAD and Within.

He holds a Process and Machine Design Engineer Degree from the University of Technology of Belfort (UTBM).

He worked for different companies in France, Germany and Switzerland. He has held different roles, from Project Management to Research or Mechanical Design.

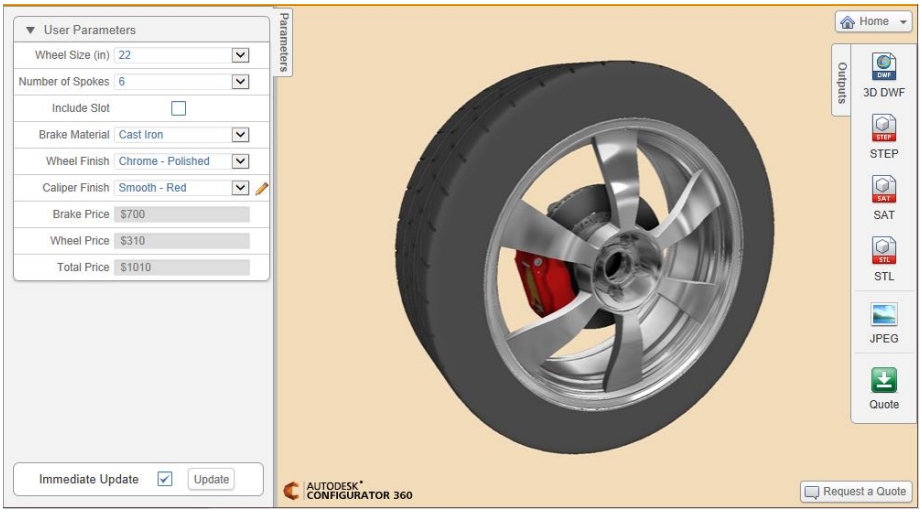
What is Autodesk® Configurator 360™ ?

The demands of consumers in both B2B and B2C situations have changed significantly over the last few years, with increasing expectations of customization of manufactured products. To keep up, manufacturers have to respond with quick and accurate quotes. Getting quotes in early drives up chances of winning bids, and providing accurate quotes is critical for the bottom line of a business. In order to set them apart from other manufacturers and provide compelling differentiation, manufacturers are now seeing the need to provide real time views of configured products and instantaneous sales drawings, in addition to proposal/quote documents.

Autodesk Configurator 360, a *Software-as-a-Service* with Autodesk® Inventor® at its core, has been created with all of the above in mind. Configurator 360 is available as on add-on subscription service to Inventor. This service not only provides a compelling browser-based front end for online product configurations, it also seamlessly connects an organization by providing CAD models and drawings as output, for engineering and production. Being able to drive 3D models, drawings and quotations using a common rules engine is a great benefit.

So how would you go about deploying this online configurator? Four steps…

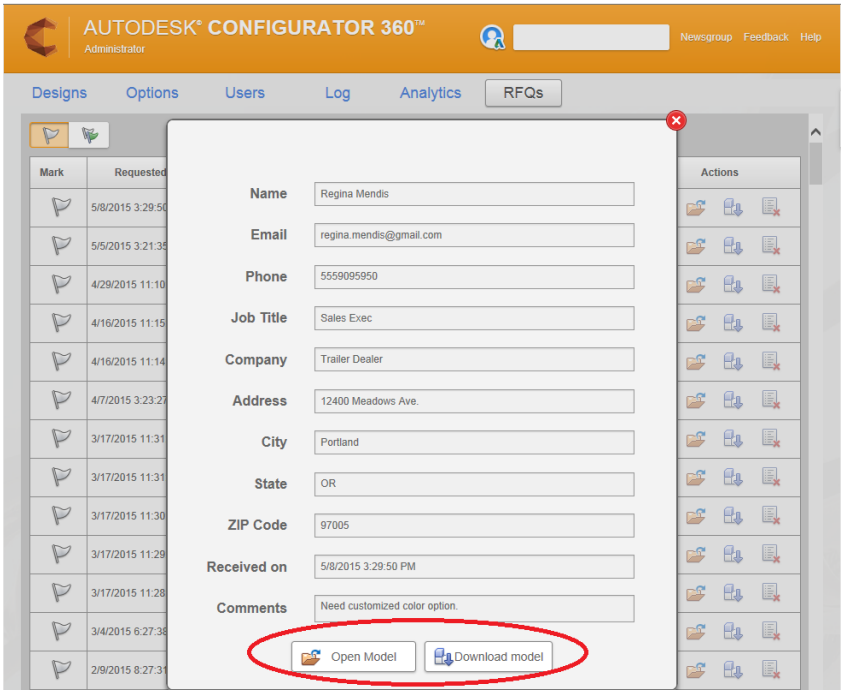
1. Create robust rules-based 3D designs using Autodesk Inventor.
2. With a push of a button, publish these designs to Configurator 360 right from within Inventor. Test out the URLs provided by Configurator 360 for the designs.
3. Embed the configurator within your own company website.
4. Optionally, connect to other external systems such as Autodesk® PLM 360™, Salesforce.com or other CRM/ERP systems for business processes.



CONFIGURATOR 360 INTERFACE

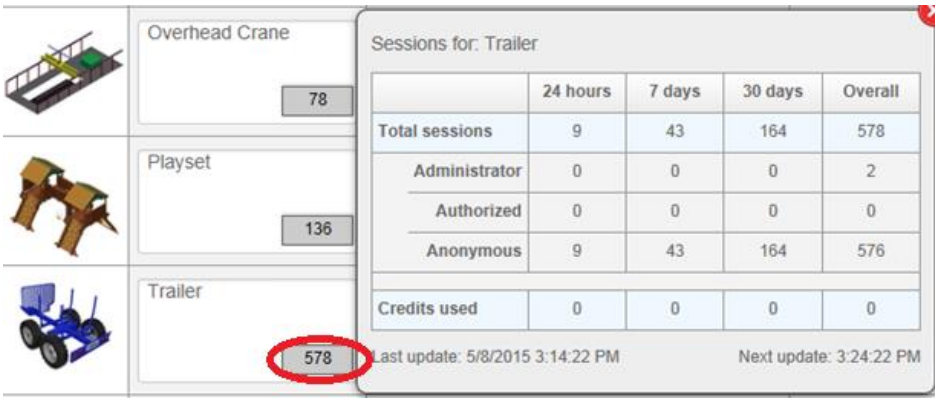
You are now ready to impress your customers with your website by providing dynamic 3D views of your products, associated drawings and if you so wish, an instant quote. You can also deploy this configurator just to your sales/dealer channel.

Any ‘Request for Quote’ action by a user sends you an email and logs the request in the Configurator 360 admin interface. You can then download the ‘as configured’ Inventor model for further engineering activities and or send it straight to production after a review.



REQUEST FOR QUOTES

The admin interface also provides analytics per design and for the entire catalog, to help you make informed decisions on popular products and where you should be investing more.



ADMINISTRATOR VIEW FROM C360

Step 1: Making a “parametric driven” design

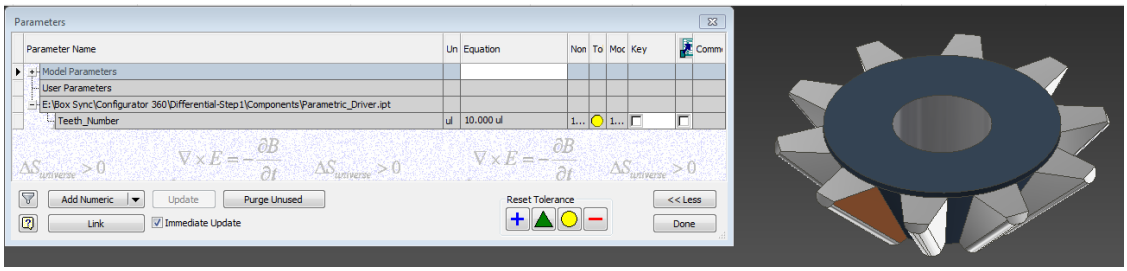
In this first step, we’ll explore different ways to configure a design using it parameters. With a well-structured inventor design, you can drive the geometry by just varying some parameters value at the top-level.

We will be working on the differential files from the STEP 1 folder in this class lab files.
Enable the project "..\Differential-Step1\Differential-Step1.ipj"

Deriving part parameters in another part

“Parametric_Driver.ipt” contains the main parameter to drive our assembly. It will be derived into the differential parts.
Only the relevant parameters for each part are derived.

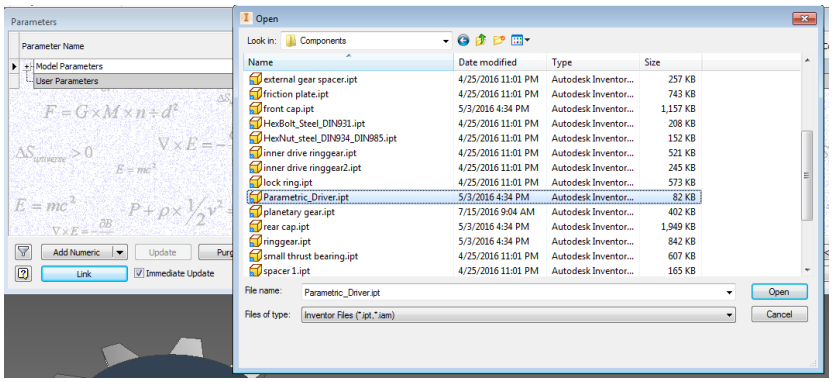
The differential parts will be updated only when the derived parameter has been modified.
This is a key difference with Excel derived parameters where any Excel change triggers a part update

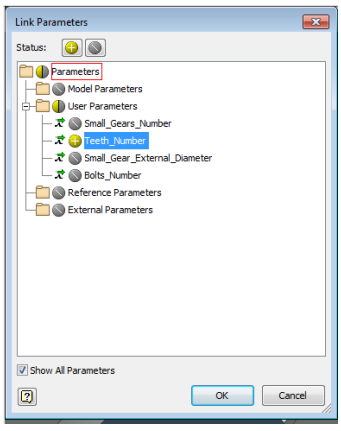


TEETH NUMBER PARAMETER IN PLANETARY GEAR

Driving Teeth Number of the Planetary Gears

- Open “Planetary Gear.ipt”. Go to Parameters and select Link.
- Ensure files of Type IPT is enabled and select “Parametric_Driver.ipt”.
- Link the “Teeth_Number” parameter only.
- Make the “Teeth” parameter from the part equal to “Teeth_Number”.





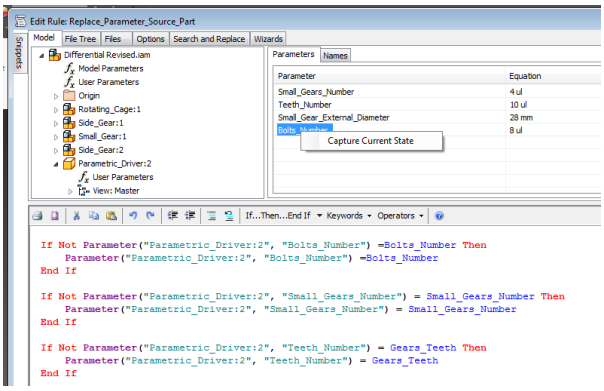
DERIVING PART PARAMETERS

Driving the parameters from the top assembly

- Open “Differential Revised.iam”.
- Check the user parameters.
- We will copy these values in “Parametric_Driver.ipt” using iLogic.
- Create a new iLogic Rule.
- Make “Bolts_Number” from “Parametric_Driver.ipt” equal to the one of the assembly.
- Add a check to ensure it is unequal before modifying it.
- Do the same for the other parameters.

The iLogic code :

```
If Not Parameter("Parametric_Driver:2", "Bolts_Number") =Bolts_Number Then
    Parameter("Parametric_Driver:2", "Bolts_Number") =Bolts_Number
End If

If Not Parameter("Parametric_Driver:2", "Small_Gears_Number") =
Small_Gears_Number Then
    Parameter("Parametric_Driver:2", "Small_Gears_Number") =
Small_Gears_Number
End If

If Not Parameter("Parametric_Driver:2", "Teeth_Number") = Gears_Teeth Then
    Parameter("Parametric_Driver:2", "Teeth_Number") = Gears_Teeth
End If
```



ILOGIC RULE TO MODIFY PART PARAMETERS FROM AN ASSEMBLY

Step 1 is complete !

Step 1 is completed.
You can open “Parametric_Driver.ipt”, change the Teeth number and check the change in the planetary Gear. You can also change parameters at the top assembly, and observe the update in the parametric driver and then in the other parts.

Step 2: Using iLogic forms and images

In this first step, we'll explore different ways to configure a design using it parameters. With a well-structured inventor design, you can drive the geometry by just varying some parameters value at the top-level.

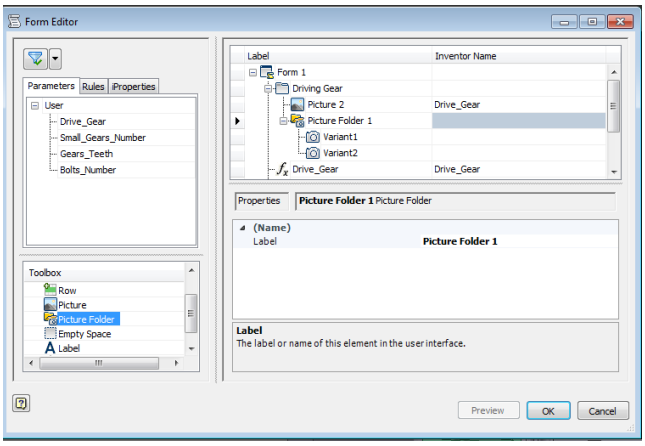
We will be working on the differential files from the STEP 2 folder in this class lab files.
Enable the project "...\Differential-Step2\Differential-Step2.ipj"

About iLogic forms and images

iLogic forms and images are supported in Configurator 360.
They will be available in your configurator and can be really helpful to guide the user in making the right choices for his configuration.
Just make sure to upload them with the Inventor files to C360.

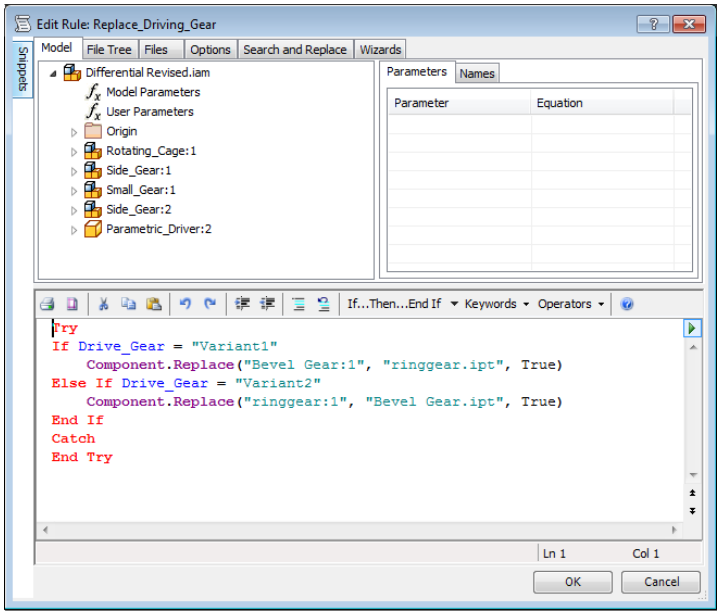
Create iLogic form to switch gear

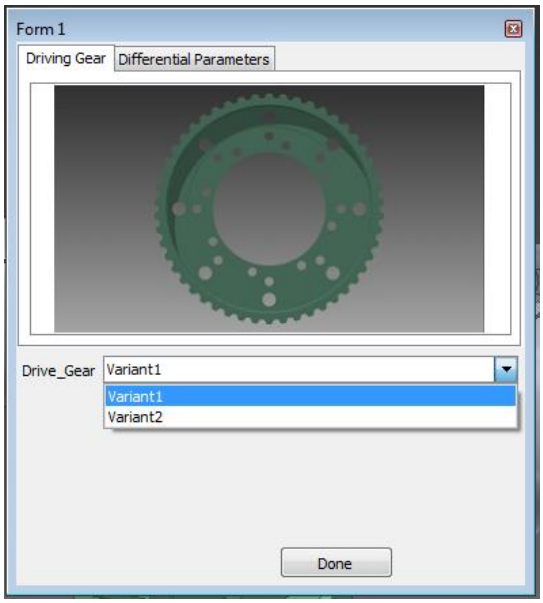
- Create a new iLogic form.
- Add a Tab Group called "Driving Gear".
- Drag 2 pictures in it, link them to gears pictures in the picture folder from the Step 2 files.
- Add parameter "Drive_Gear" to the tab.
- The name of the pictures must match the "Drive_Gear" values to change accordingly.



ILOGIC FORM EDITOR

Test the form to see the result.The assembly has an iLogic rule to replace the gear component depending on Drive_Gear value.
The iLogic form picture folder will switch the picture depending on Drive_Gear value.

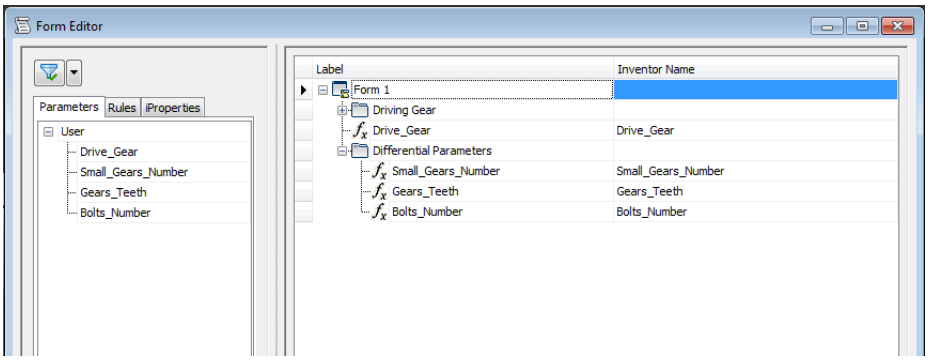




ILOGIC RULE TO REPLACE THE GEAR, AND ILOGIC FORM TO CHANGE THE PARAMETER VALUE

Add the user parameters to the form

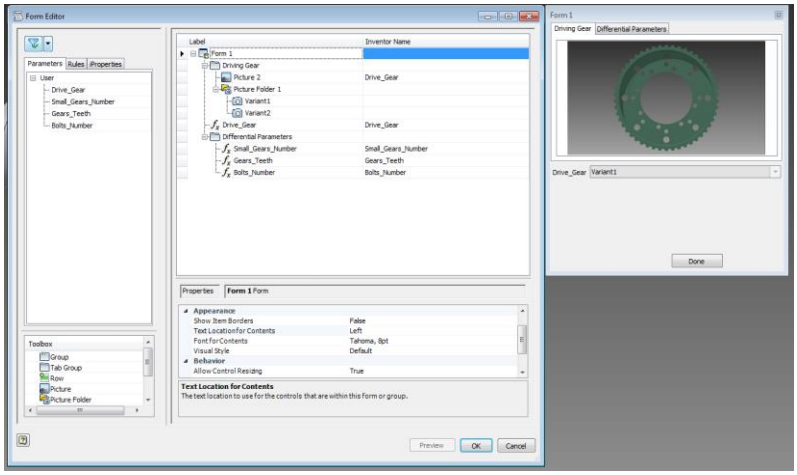
- Go back to the iLogic form.
- Add a Tab Group called “Differential Parameters”.
- Drag the user parameters of the main assembly.
- Though we will be able to edit them directly without having to go in the parameters window.



ADDING A TAB TO DRIVE THE OTHER PARAMETERS IN THE FORM

Step 2 is complete !!

Step 2 is completed.
You have created an iLogic form with images that can drive your assembly configuration.
This kind of solution can be very useful to guide your users in their choices.



ILOGIC FORM IS COMPLETE

Step 3: Setting up the C360 environment

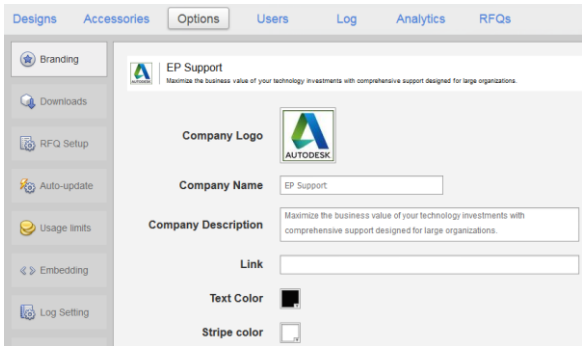
In this step, we will log into Configurator 360 and explore the environment.
We will define the most important settings like company description, setting the company logo...
All kinds of things that need to be set at the beginning, when you are implementing Configurator 360.

We will have to navigate to **"configurator360.autodesk.com".** Make sure you have access to it. You can start the trial period if needed.

Setting up the branding options

Here we'll define the different settings that are available to describe your company in the C360 Option.

- Navigate to "configurator360.autodesk.com".
- Sign-In.
- Go to the Options Tab / Branding.
- Set the company name, logo and other settings for your catalog.



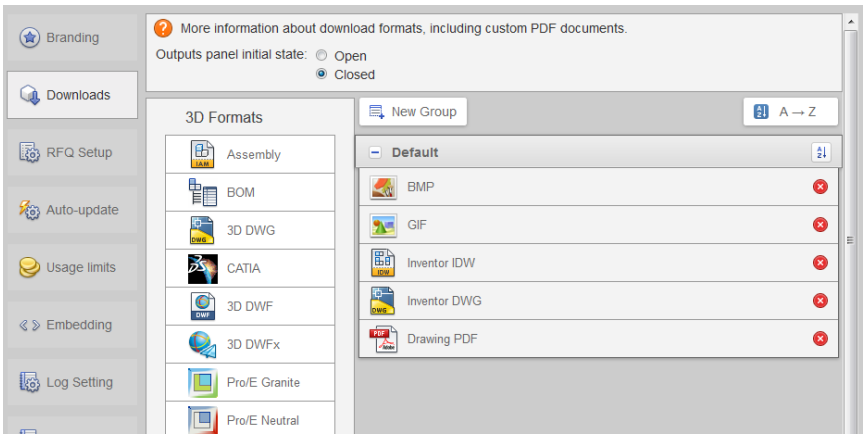
OPTIONS TAB TO CONFIGURE THE CATALOG

Setting up the download options

Download types are very important to define. Depending on who is going to access your online catalog, you will choose which download types are enabled.

For example, if this catalog is for your internal colleagues, you can enable the download of Inventor files. Though, they will be able to manipulate them afterwards.
But if your catalog will be available to anyone on the internet, you might want to give the ability to download images only.

- You might want to enable / disable the drawing download.
- Go to the Downloads tab, and drag and drop some file types.

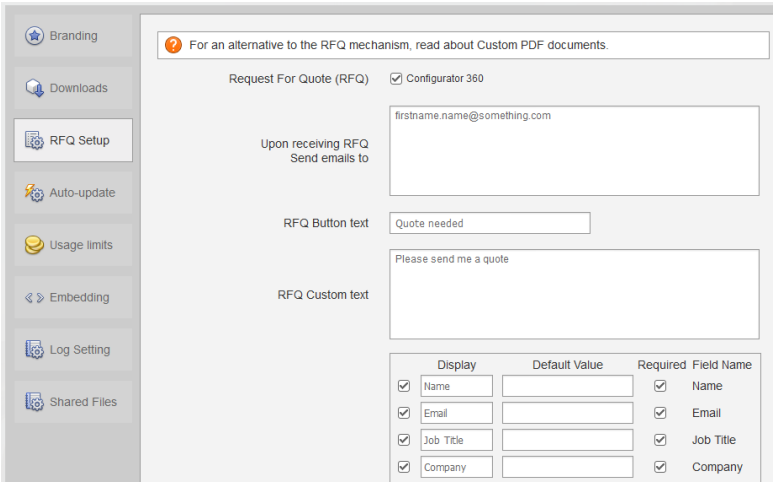


AVAILABLE DOWNLOAD OPTIONS FROM C360

Setting up the quote options

In C360, you can customize your quote. You can create custom word templates, using merge fields. We will not go too much in detail in this lab.
For more information, see this link.

- Go to the RFQ Tab in the C360 Options.
- Type some text in the fields where needed, so you will be able to test it once there is a design to configure.



RFQ SETUP IN C360

Step 3 is complete !!

Step 3 is completed.
You have configured your C360 Catalog like an administrator would do.

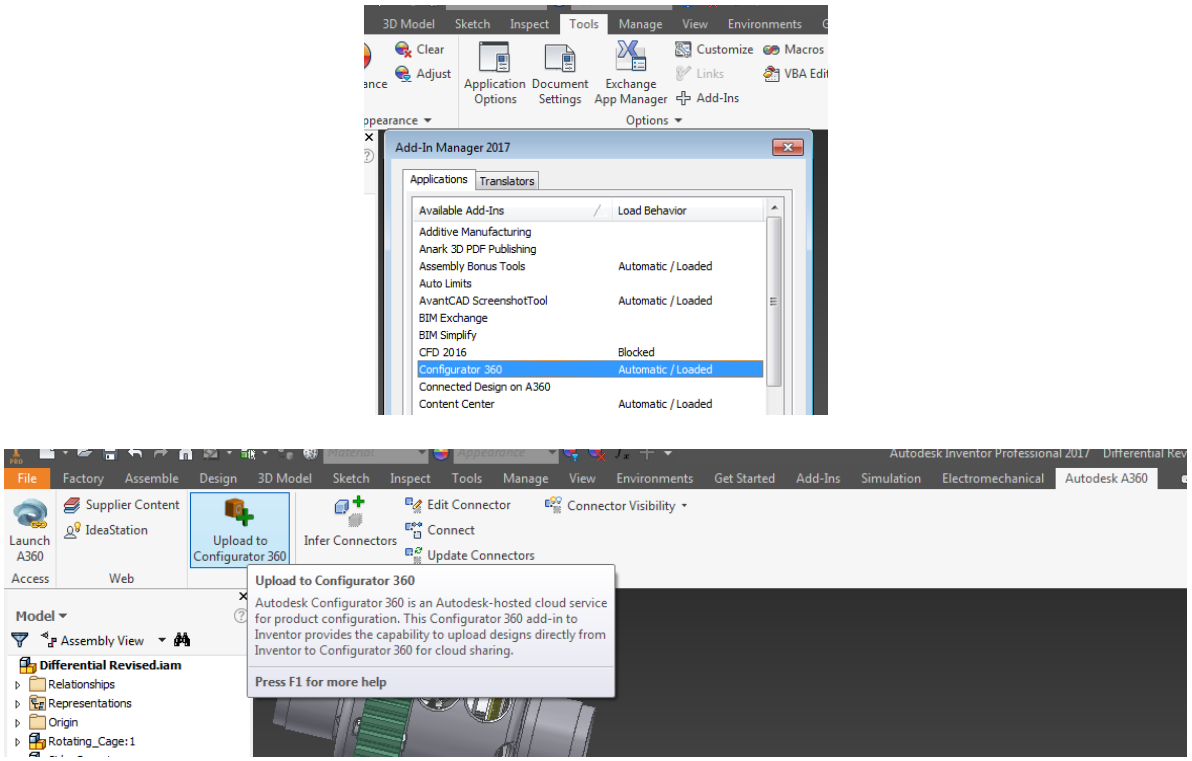
Step 4: Loading the model in C360

In this step, we upload our model to Configurator 360. The files can be uploaded from Inventor, using the Configurator 360 Plugin.
You can also directly the files from the configurator 360 portal. For an assembly, all the files are loaded as a ZIP File.
When working with configurable designs, it is important to load all the files needed, and not only the active ones in the assembly. Sometimes, the assembly might have to load other files depending on its configuration.
For example, in our exercise files, we use an iLogic form which is loading images. It is important to make sure these image files are also loaded onto C360.

Uploading the model from Inventor

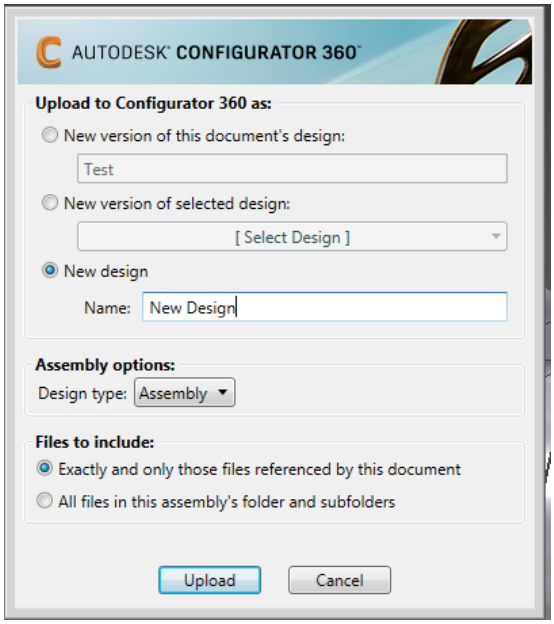
We will be using the C360 Plugin from Inventor. **Make sure to enable the project "...\Differential-Step4\Differential-Step4.ipj"**

- Open “Differential Revised.iam” from the step 4 files.
- Go to Tools / Add-Ins and ensure the configurator 360 Add-In is loaded.
- Go to the “A360” tab and select “Upload to Configurator 360”.



UPLOADING DESIGN FROM THE C360 PLUGIN

- Select “New Design”, give it a name.
- The design type will be assembly.
- Upload all files in this assembly’s folder, otherwise the gear variants will not be loaded.

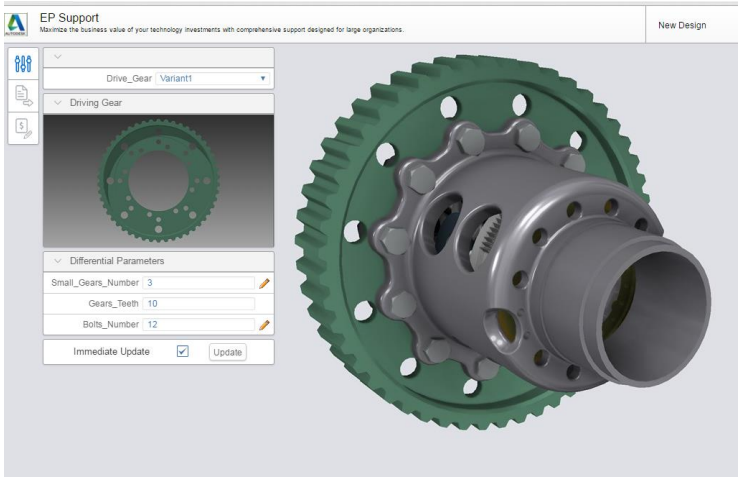


UPLOADING THE DESIGN TO C360 IF REFERENCED FILE ONLY

Navigating in the configurator environment

We will log back again in Configurator 360, and open this design to configure it.

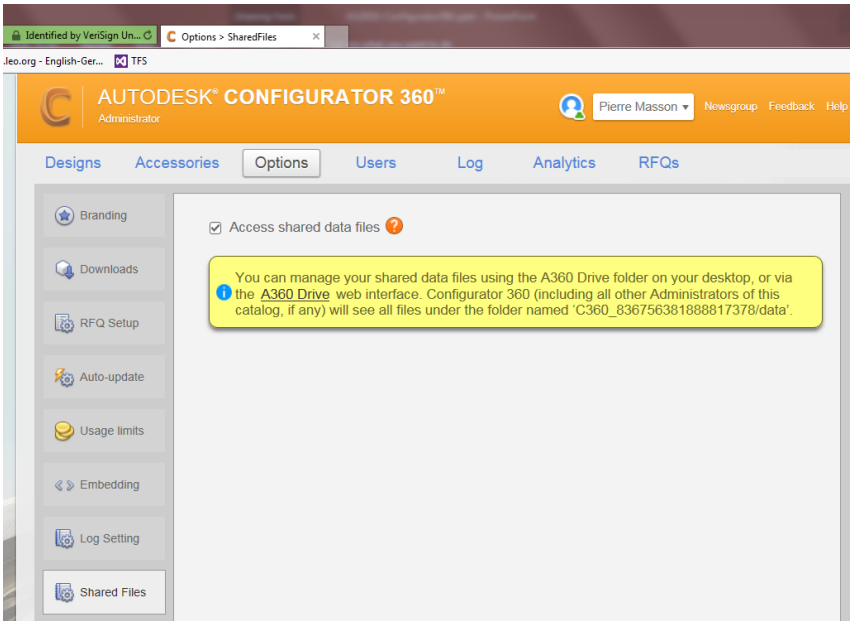
- Open the design in C360.
- The iLogic form is loaded with the pictures.
- You can also drive the parameters we exposed in the iLogic form.
- Navigating in 3D is similar to Inventor (Mouse Wheel pressed and Shift for Orbit).
- Play with the parameters to change the configuration.



DIFFERENTIAL LOADED IN C360

Sharing Files across design

It is possible to define common files for all your designs.
For example you have a central excel sheet containing data for your design components. It is also possible to put a database like an SQLite for example.
You can enable this in the Options, tab “Shared Files”.



ENABLING SHARED FILES ON C360

Step 4 is complete !!

Step 4 is completed.
You have uploaded your first design on C360, and configured it to verify it is working.

Step 5: Displaying a message in C360

By configuring special iLogic rules in Inventor, you can trigger a message in C360. This is useful when you want to say something to your user, depending on what he choose.
For example, the user might have chosen a set of parameters that cannot work together, and you want to warn him about that.

C360 uses a custom iProperty with a special name to detect if user messages must be displayed.

With a bolts number below 8, our differential design is not ok.
We will display a message in C360 if the user chooses a bolt number outside a range. We will also check that the number of gears is acceptable.

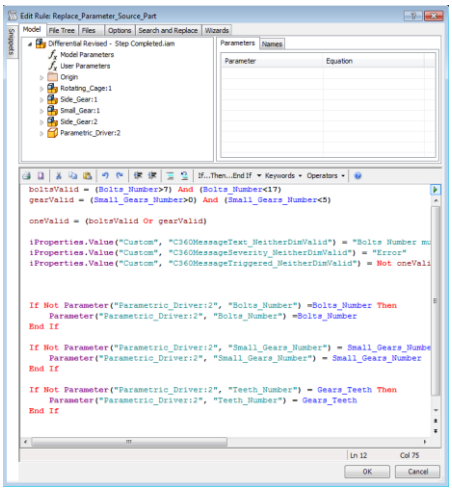
Creating the iLogic code to display a message

Enable "..\Differential-Step5\Differential-Step5.ipj".

- Open the differential revised assembly.
- Edit the iLogic rule “Replace_Parameter_Source_Part”.
- Add the following code at the beginning of the rule.
- OneValid will be a Boolean variable that is true if the number of bolts and the number of gear is ok.
- The iProp “C360MessageTriggered” will be the one to trigger the message.

Use this iLogic code :

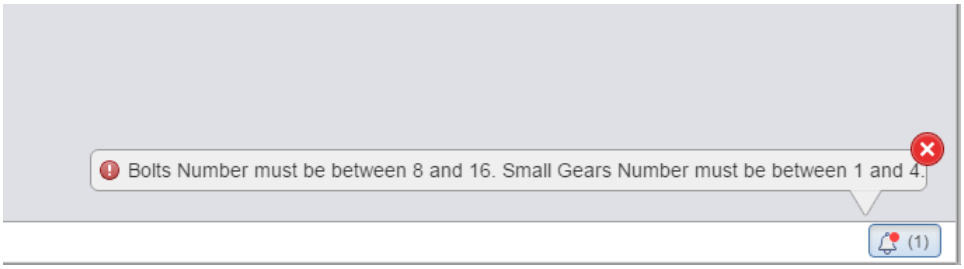
```
boltsValid = (Bolts_Number>7) And (Bolts_Number<17)
gearValid = (Small_Gears_Number>0) And (Small_Gears_Number<5)

oneValid = (boltsValid And gearValid)

iProperties.Value("Custom", "C360MessageText_DimValid") = "Bolts Number must be
between 8 and 16. Small Gears Number must be between 1 and 4."
iProperties.Value("Custom", "C360MessageSeverity_DimValid") = "Error"
iProperties.Value("Custom", "C360MessageTriggered_DimValid") = Not oneValid
If Not oneValid Then Exit Sub
```



ILOGIC RULE TO DISPLAY A WARNING IN C360

Uploading the design and checking the message

- Upload your design from the C360 Inventor plugin.
- Create a new version of the previously uploaded design.
- Upload all files in the folder.
- Enable the latest version of the design and open it.
- Test the message.



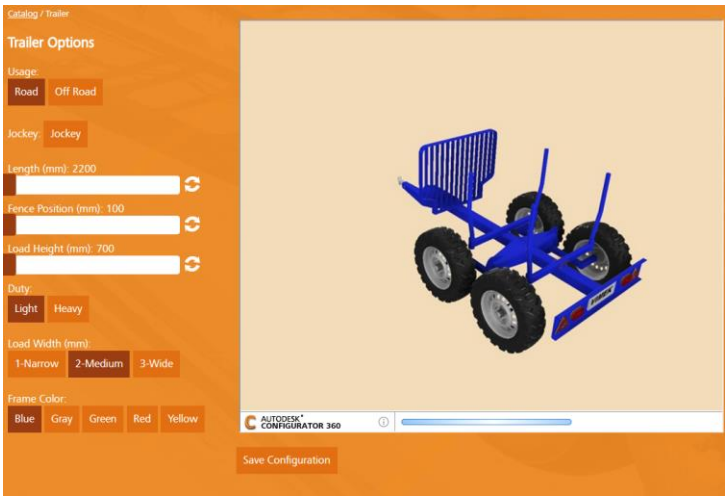
WARNING DISPLAYED IN C360

Step 5 is complete !!

Step 5 is completed.
You have learned how to display a message in C360 using iLogic rules.

Step 6: Embedding the design in your webpage

Configurator 360 has an API that you can call from your own webpage.
So you can embed the C360 Window into you own webpage. C360 gives you the ability to view and configure the model, and you can add the controls you want, the information your users need on the webpage.
This task is usually done by web programmers, but we'll explore the basics together, so you have and understanding on how it's supposed to work.

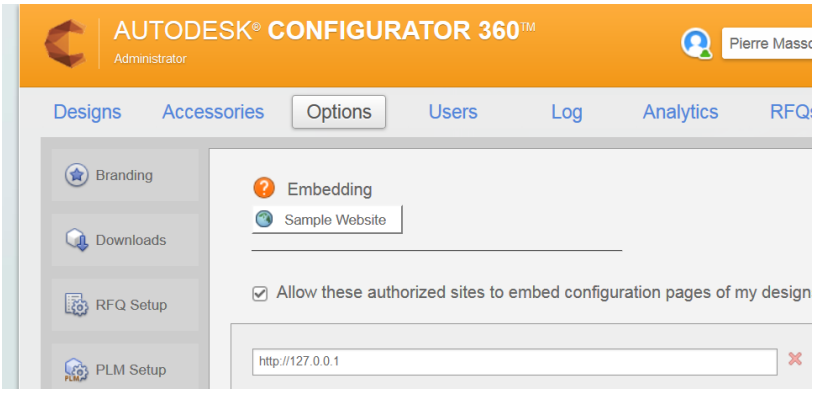


EMBEDDED C360 DESIGN IN A CUSTOM WEB PAGE

Preparing C360 for embedding

Navigate back to the Configurator 360 Admin page. We have to enable some settings so other webpages can call the C360 configuration.
In our exercise we will use the local address of the machine, because this is where we will test the embedding.

- In C360 Options, allow these sites to embed your configurations :
- http://127.0.0.1
- http://localhost
- Enable your design for unrestricted viewing.

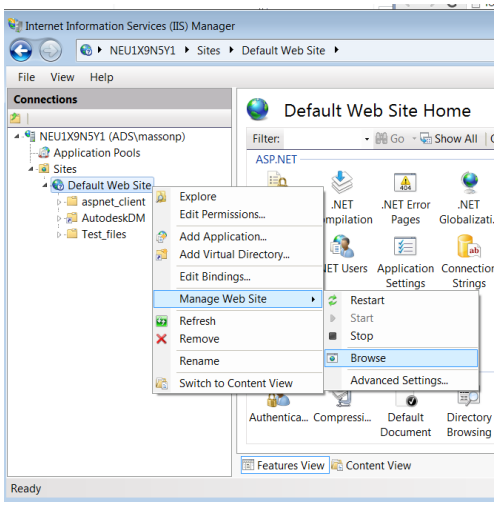


ENABLING EMBEDDING IN C360

Installing and enabling IIS

IIS is not mandatory for what we want to do. But this is the tool we will use in order to run webpages locally on the computer we are using.

- Install IIS on your computer (from the official Microsoft IIS site).
- Open it, go to default Web Site, then Manage Web Site.
- Make sure it is started and go to browse.
- A Welcome page is opened.



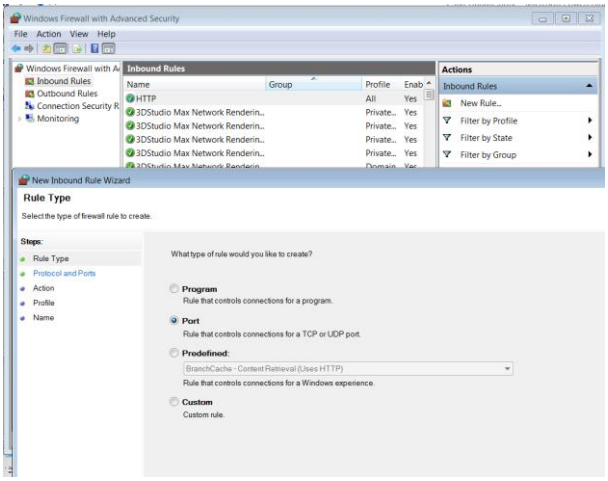
IIS MANAGER

SUCCESSFUL IMPLEMENTATION OF IIS

Opening Port 80 in the firewall

Port 80 is used for communication over the internet. It is possible that it is already opened.

- Go to Windows Firewall.
- Advanced Parameters.
- Create an Inboud Rule and select Port.
- Open Port 80.



FIREWALL SETTINGS IN WINDOWS 7

Creating your HTML Page

There is good online documentation on how to embed the C360 configuration. We will use the samples and simply modify the reference ID to make if work with our configuration.

- Navigate to : http://embedding.configurator360.autodesk.com/doc.html
- Create a new NotePad File
- Copy the content of “Putting it all together” in the Notepad file.
- Save it as an HTM File.

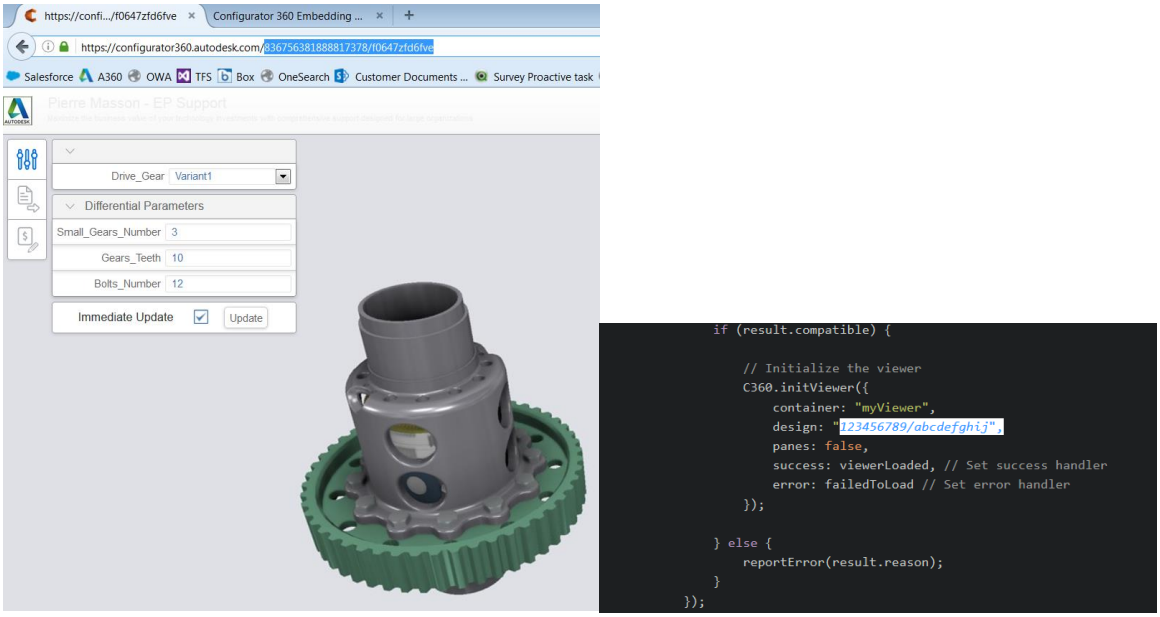
Putting it all together

Example:

```
<!DOCTYPE html>
<html lang="en" >
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Simple Embed Test</title>
    <style type="text/css">
        div#myViewer {
            position: relative;
            width: 1132px;
            height: 768px;
        }
    </style>
    <script src="https://configurator360.autodesk.com/Script/v1/EmbeddedViewer"></script>
</head>
<body>
<h1>Test</h1>

<div id="myViewer"></div>

<script type="text/javascript">
    (function (window) {
        "use strict";
        var C360 = window.ADSK && window.ADSK.C360;

        // callback for getPropertyValues.
        function listProperties(result) {
            window.console.log(window.JSON.stringify(result, null, '  '));
        }

        // dummy implementation for handling error codes.
        function reportError(code) {
            var state = C360.loadedState;
            if (code === state.GPUAccelerationDisabled) {
                window.alert('GPU hardware acceleration is disabled.');
```

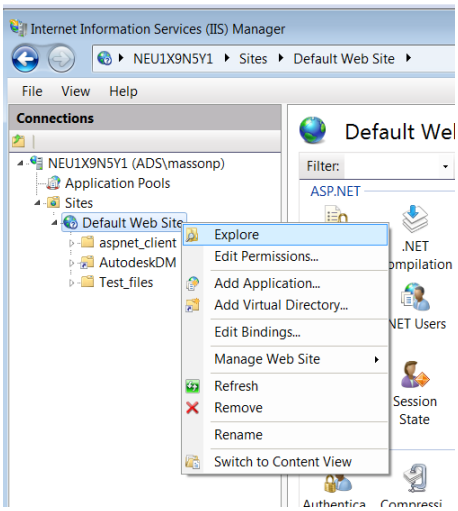
C360 EMBEDDING EXAMPLE

- Go back to the administrator page of Configurator 360.
- Navigate to your C360 Design and copy the codes after “autodesk.com/”
- Paste it in the Design Code in the HTM File.



COPY – PASTING THE CODE OF YOUR DESIGN

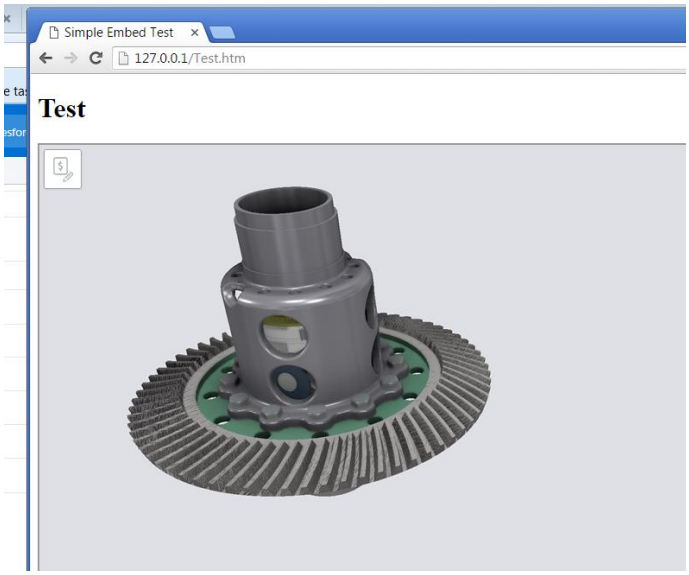
- Back to IIS, right click Default Web Site and click Explore.
- Paste the HTM File in this directory.
- Click the HTM File to open it in a browser.



BROWSING TO THE IIS DEFAULT WEB SITE

- Open the HTM file with a Browser.
- Make sure 127.0.0.1 or localhost is in the address.
- Rotate and check the file.

As you can notice the configuration controls are not available. This is because panes is set to false in our html code.



WEB PAGE WITH 127.0.0.1 ADDRESS CALLING YOUR C360 DESIGN

Step 6 is complete!!

Step 6 is completed.
You have learned how to enable a quick test environment for webpages using IIS. You have learned how to call your C360 Configuration from another webpage.

Step 7: Adding elements to the HTML Page

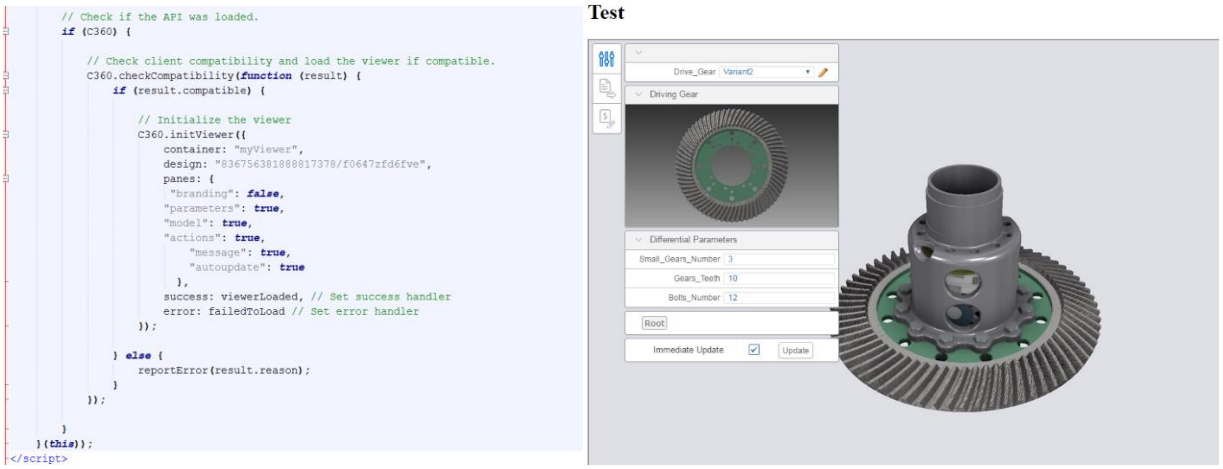
We will now discover more of the C360 API.
Using this API, we will see how to show or hide the C360 panes. You might want to let only the 3D Models visible and handle all the configuration with your own webpage controls.
We will also see how to modify the parameters of your design using the C360 API. For example, you might want to put a control some dimensions with a slider on your webpage. Then using the API, you can send the new dimension to C360 to update the model accordingly.

Display and Hide Panes

We will work again on the webpage we created in the last exercise.

- Modify the Page using NotePad.
- Locate the C360.initViewer function.
- Paste the text below to display the panes. The syntax is very important.
- You can decide to hide or display specific components.

```
// Initialize the viewer
C360.initViewer({
container: "myViewer",
design: HEREYOURDESIGNNUMBER,
panes: {
"branding": false,
"parameters": true,
"model": true,
"actions": true,
"message": true,
"autoupdate": true
},
success: viewerLoaded, // Set success handler
error: failedToLoad // Set error handler
});
```

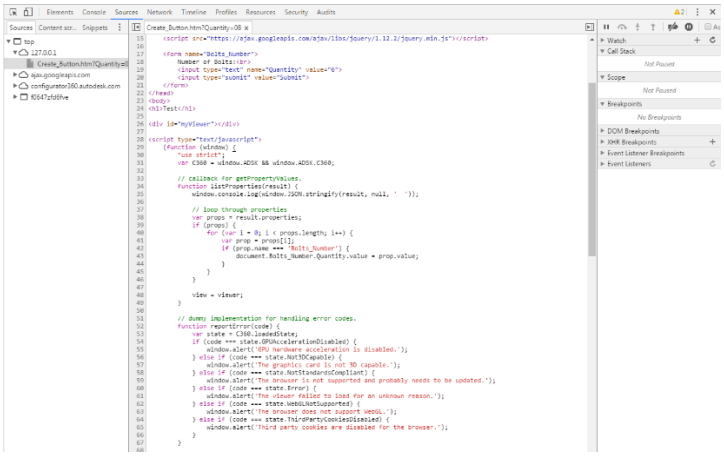


SHOWING AND HIDING PANELS USING THE HTML CODE

Debugging in Google Chrome

We will use Google Chrome to explore how the webpage is loaded and if it works properly. This is just for giving a basic idea on how to troubleshoot your webpage, if you're having troubles with the C360 Design.

- Start Google Chrome.
- Drag and Drop the HTML page from the wwwroot folder.
- If needed, change the address to 127.0.0.1
- Press F12.
- Navigate through the different tabs like Console, Sources…

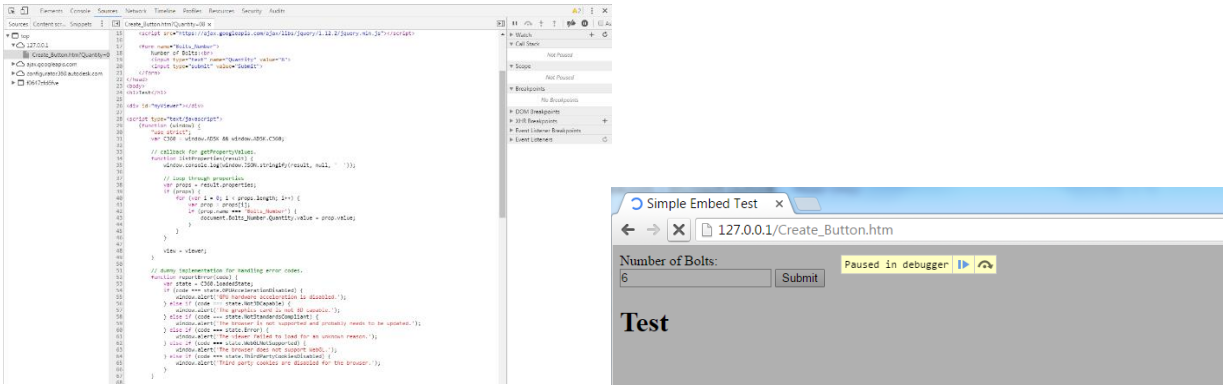


DEBUG ENVIRONMENT IN GOOGLE CHROME

We will now add a breakpoint to stop the execution of the webpage at a specific point. So we can study what is happening at that time of the execution.

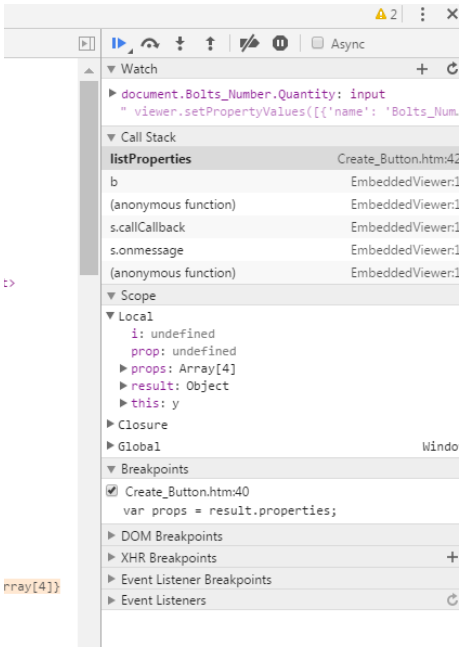
- Go to Sources Tab.
- Edit the HTM file under 127.0.0.1
- Locate the listProperties function.
- Click on the grey area where lines number are listed.

- It will create a breakpoint.
- Refresh the page with F5.
- The design will be paused in that line and we can do some diagnostics.



BREAKPOINT IN GOOGLE CHROME. THE BROWSER SHOWS « PAUSED IN DEBUGGER »

Once the execution is stopped, on the right pane of the browser, we can go step by step and observe the code execution of each line. We can also see the status of each local variable.



LOCAL VARIABLE STATE IN CHROME WHILE EXECUTING STEP BY STEP

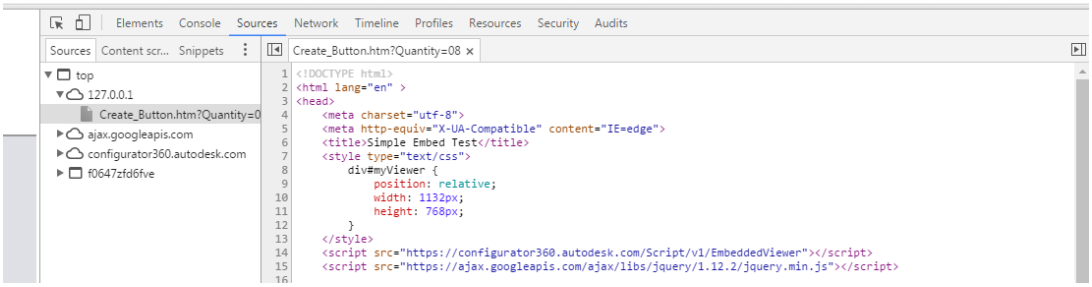
Remove the breakpoint so we can work normally again.

Creating a button

We will add a button in the webpage, and use this button to manipulate a parameter of our design. We will use a special library called jQuery.

In our case, we will manipulate the number of Bolts of our design.

Add this line to call the jQuery library :
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.2/jquery.min.js"></script>



CALLING THE JQUERY AND THE C360 API IN OUR WEBPAGE

- Place a Form called Bolts_Number.
- Put a text field with default value to 6.
- Add a submit button.

This is the text of the form:
<form name="Bolts_Number">
*Number of Bolts:
*
<input type="text" name="Quantity" value="6">
<input type="submit" value="Submit">
</form>

```
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Simple Embed Test</title>
    <style type="text/css">
        div#myViewer {
            position: relative;
            width: 1132px;
            height: 768px;
        }
    </style>
    <script src="https://configurator360.autodesk.com/Script/v1/EmbeddedViewer"></script>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.2/jquery.min.js"></script>

    <form name="Bolts_Number">
        Number of Bolts:<br>
        <input type="text" name="Quantity" value="6">
        <input type="submit" value="Submit">
    </form>
</head>
```

HTML CODE CALLING THE JQUERY LIBRARY AND ADDING A BOLTS_NUMBER FORM

Now we will write the code to handle the value of this button. When loading the page, we want the button value to be updated accordingly to the current parameter.

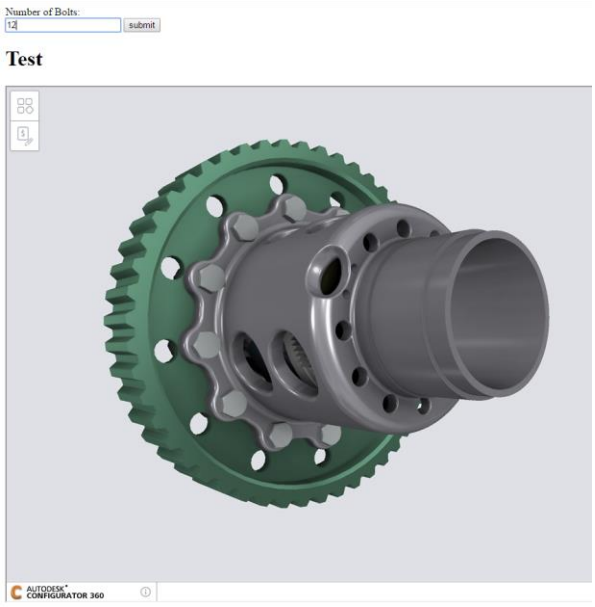
- When the design is loaded we use list properties as callback to get all properties of the design.
- Edit this function to loop through all props.
- When Bolts_Number is found, take its value to our button.

This is the code :

```
// callback for getPropertyValues.
      function listProperties(result) {
         window.console.log(window.JSON.stringify(result, null, '  '));

                              // loop through properties
                              var props = result.properties;
                              if (props) {
                                    for (var i = 0; i < props.length; i++) {
                                          var prop = props[i];
                                          if (prop.name === 'Bolts_Number') {
                                                document.Bolts_Number.Quantity.value =
prop.value;
                                          }
                                    }
                              }
```

Now let's test this.

- Copy the HTML file to the wwwroot folder.
- Then drag and drop the file in your browser. Change the address to 127.0.0.1
- The number of Bolts value should be updated.
- Change it and see the number of bolts updating.



NUMBER OF BOLTS INPUT FIELD UPDATING THE C360 DESIGN

Congrat’s, you’re done !!

Thanks a lot for taking the time to go through this course. Together, during the course we saw how to :

- Drive parameters in Inventor using derived part.
- Drive parameters and assembly components in Inventor using iLogic.
- Create an iLogic form with images.
- Configure your C360 catalog.
- Upload and share designs on C360.
- Display warnings on C360.
- Embed the configurator in your webpage.