

Discover the Flexibility of iLogic in Inventor

Pascale Brassat

Dipl.-Ing.(FH) | www.claudiuspeters.com

General Information

Who am I?

Pascale Brassat

Very short summary

Claudius Peters Projects GmbH

**Since 2007 I'm using Inventor and since version
2010 I'm using iLogic.**

**Technical draftsman and afterwards university
degree in mechanical engineering and informatics.**



Autodesk Inventor

Parametrical 3D CAD System



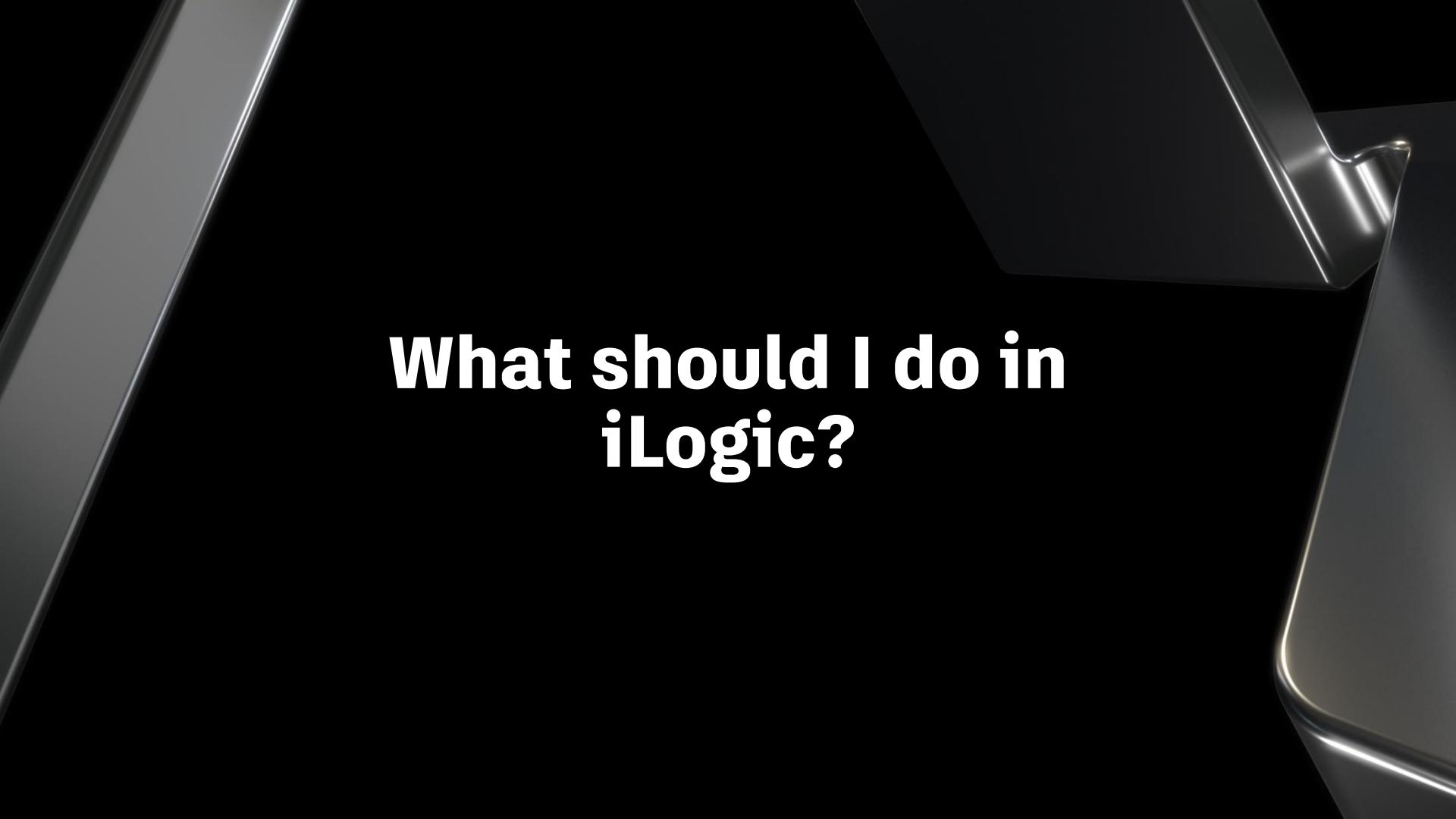
iLogic

Programming environment in Inventor

- Easy to learn
- Flexible usage
- Possible to combine with external tools
- Event triggering is easy to use
- Tons of possibilities

The screenshot shows the Inventor iLogic programming interface. On the left, the 'Snippets' browser displays a hierarchical list of available snippets under 'System | Custom'. The 'Features' section is expanded, showing various snippets like 'IsActive', 'Color', and 'SetThread All'. Other sections include 'Parameters', 'Components (classic)', and 'iLogic Assemblies/Components'. On the right, the main workspace shows a code editor with the following VBA-like script:

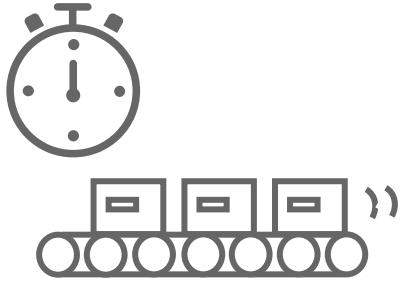
```
If ThisDoc.Document.DocumentType = DocumentTypeEnum.kPartDocument Then
    Dim Teil As PartDocument = ThisDoc.Document
    If Teil.ComponentDefinition.SurfaceBodies.Count > 1 Then
        Dim Aktion As Transaction = ThisApplication.TransactionManager.CreateTransaction("Tool Bodies")
        Dim oToolBodies As ObjectCollection = ThisApplication.TransactionManager.CreateObjectCollection("Tool Bodies")
        Dim i As Integer = 1
        For Each Element As SurfaceBody In Teil.ComponentDefinition.SurfaceBodies
            If i > 1 Then
                oToolBodies.Add(Element)
            End If
            i = i + 1
        Next
        Aktion.Commit()
    End If
End If
```



**What should I do in
iLogic?**

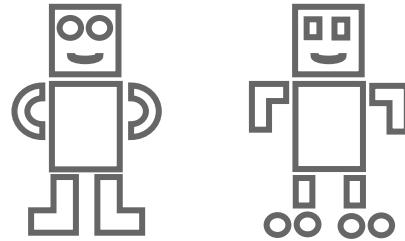
Tasks for iLogic

How to identify possible tasks for iLogic

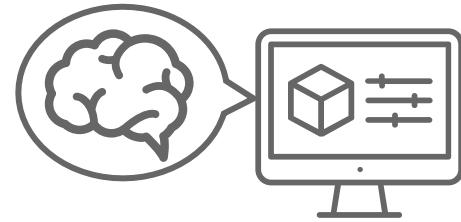


Repeating and easy to solve tasks.

3 recommendations



Library parts, which has a large number of variants.

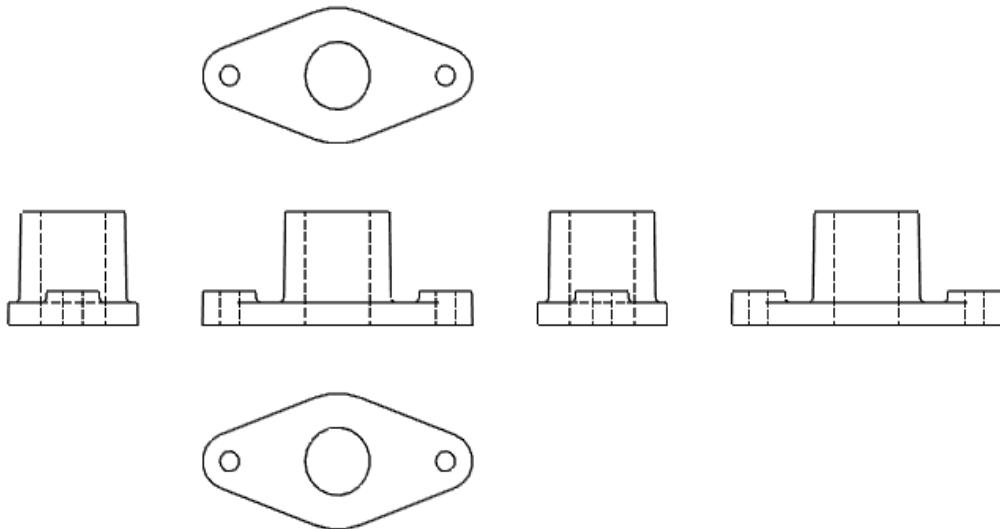


Tasks, which need very specific knowledge.
Preserve knowledge into programming.

Simple Examples

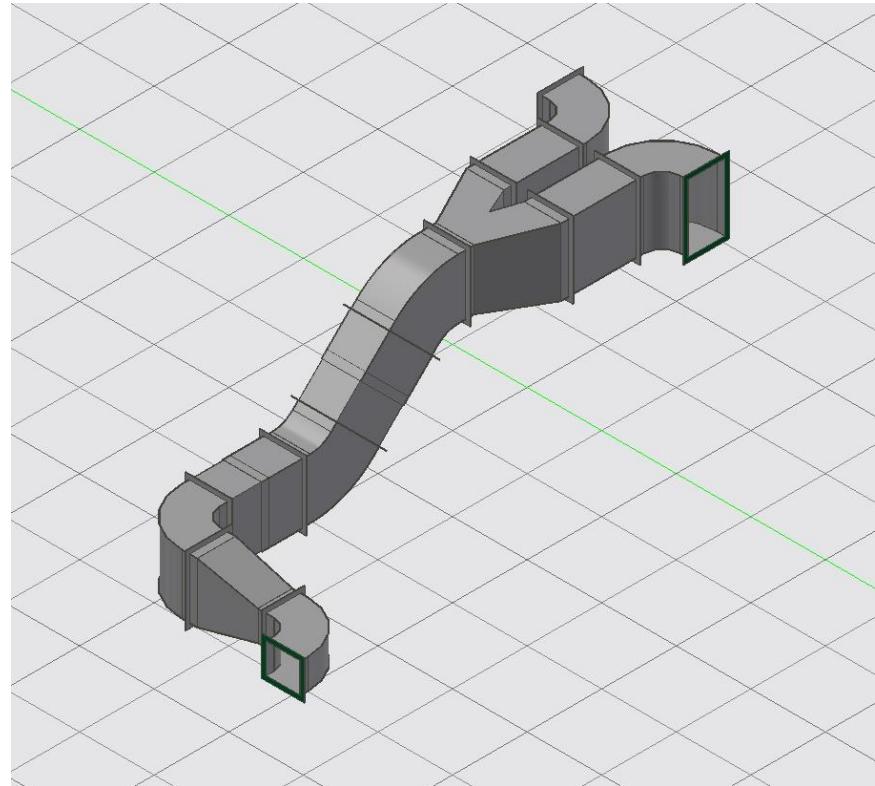
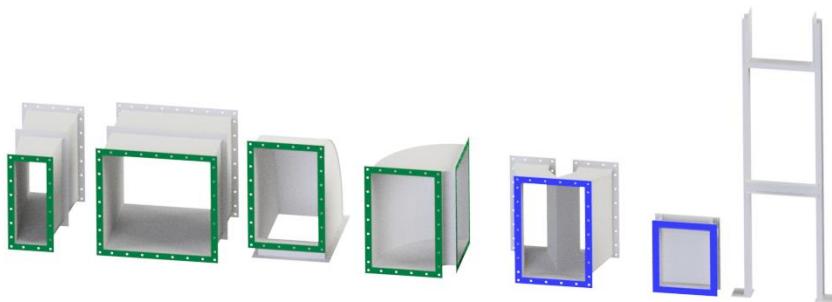
Repeating simple tasks

Create basic drawing views based on a selected main drawing view



Library

Create component libraries (with lots of variants)



Knowledge base

Put knowledge into programming

It can also be phrased differently:

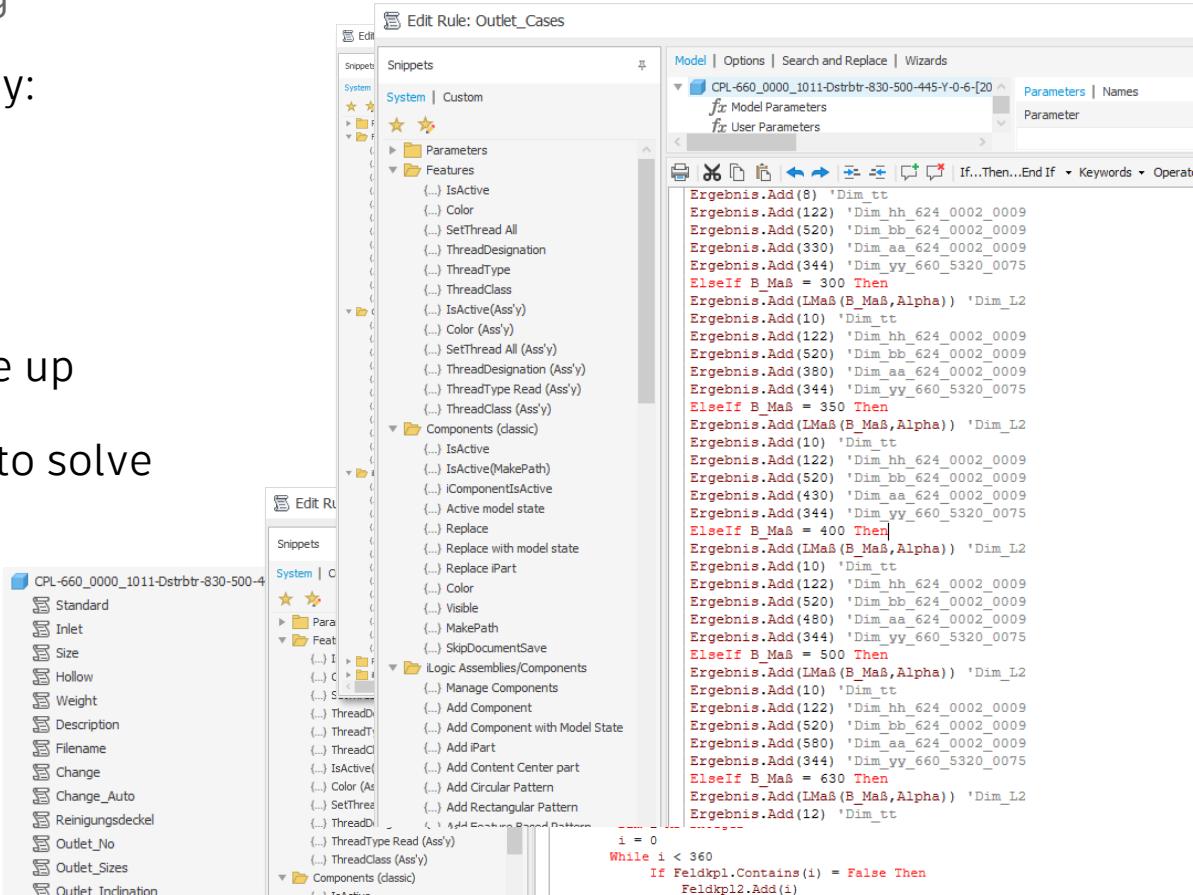
These are tasks that are not

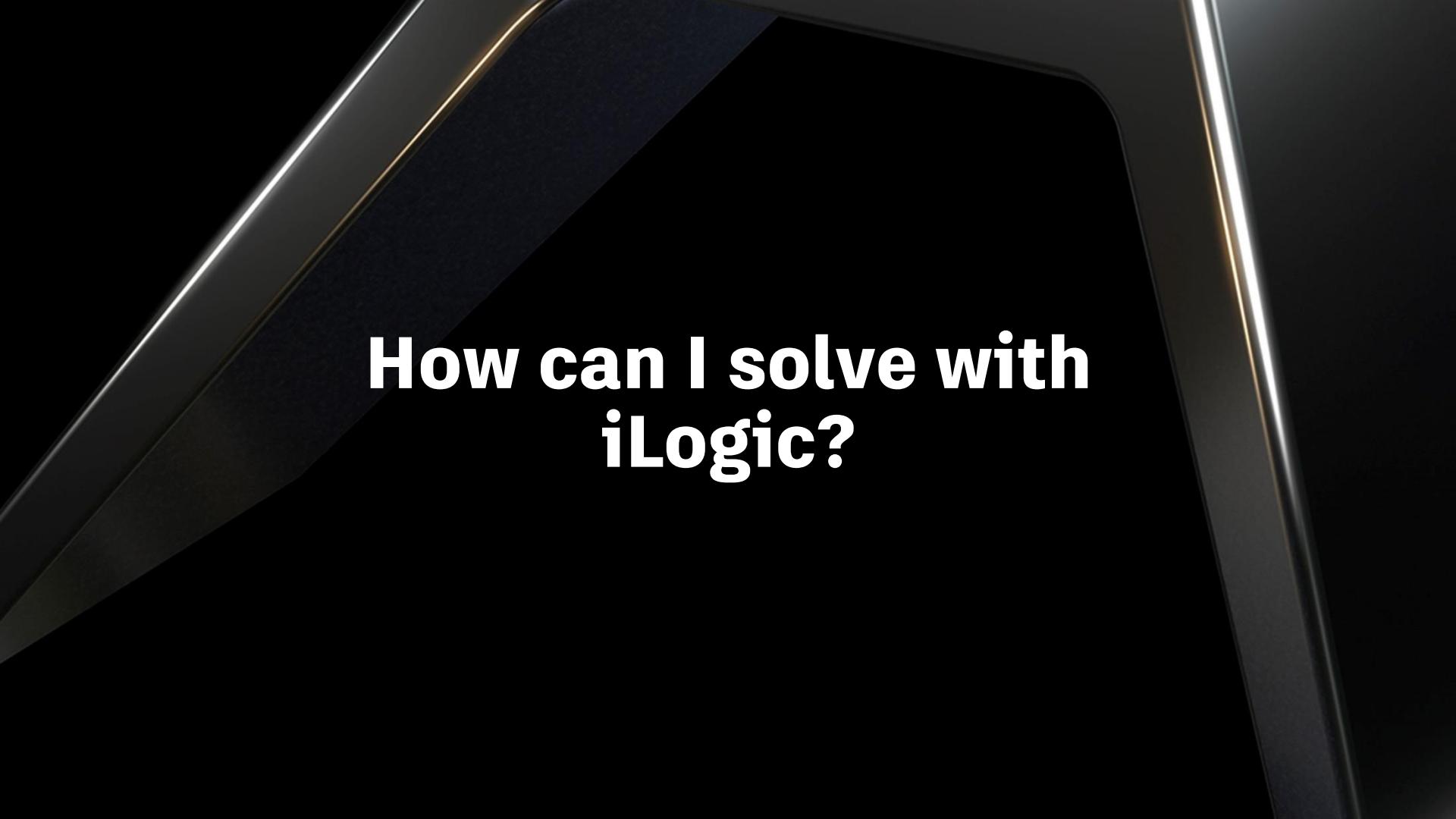
often get done, but when they're up

Landing on the table they need to solve

much time.

(Research, inquiries, etc.)





**How can I solve with
iLogic?**

iLogic Structure

Where to begin?

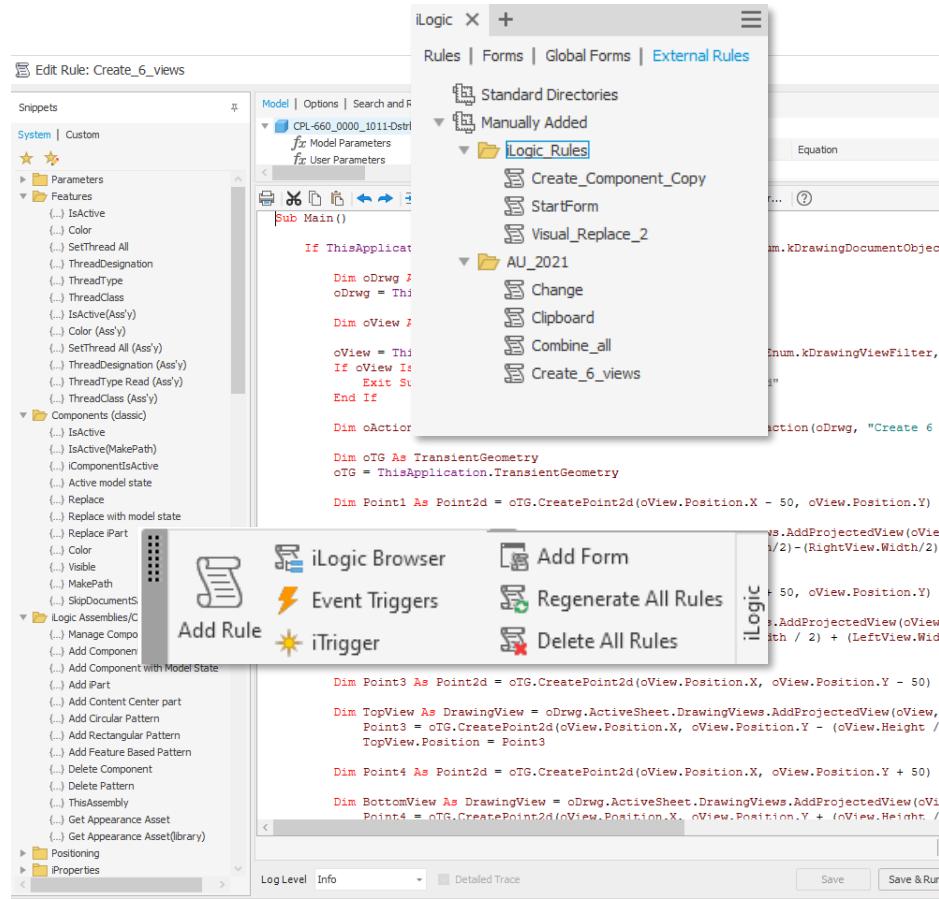
Programming in iLogic needs Inventor,
the iLogic Browser and an idea.

Snippets helps a lot.

Please do not forget to check the options!

You can find a lot of information in the web
which is very helpful for your iLogic work.

Additional search terms: vb.net or API



Example

Combine all solids in a part with one click

```
If ThisDoc.Document.DocumentType = DocumentTypeEnum.kPartDocumentObject Then  
    Dim Teil As PartDocument = ThisDoc.Document  
  
    If Teil.ComponentDefinition.SurfaceBodies.Count > 1 Then  
  
        Dim Aktion As Transaction = ThisApplication.TransactionManager.StartTransaction(Teil, "Combine All")  
  
        Dim oToolBodies As ObjectCollection = ThisApplication.TransientObjects.CreateObjectCollection  
  
        Dim i As Integer = 1  
  
        For Each Element As SurfaceBody In Teil.ComponentDefinition.SurfaceBodies  
            If i > 1 Then  
                oToolBodies.Add(Element)  
            End If  
            i = i+1  
        Next  
  
        Dim oCombine As CombineFeature  
        oCombine = Teil.ComponentDefinition.Features.CombineFeatures.Add(Teil.ComponentDefinition.SurfaceBodies.Item(1), _  
            oToolBodies, PartFeatureOperationEnum.kJoinOperation, False)  
  
        Aktion.End  
  
    End If  
  
End If
```

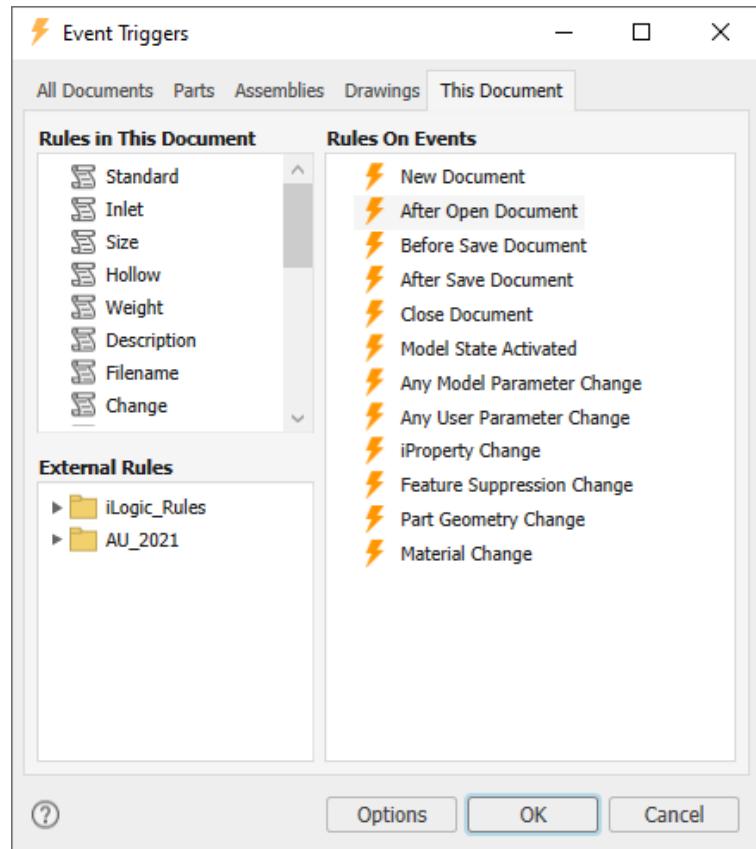
iLogic Events

When should your program start to work?

e.g. in case the document will be saved.

Or in case of parameter changes.

```
dim_PDM_T = dim_PDM_T
```



iLogic Forms

User Interface

You can create simple User forms to control iLogic rules.

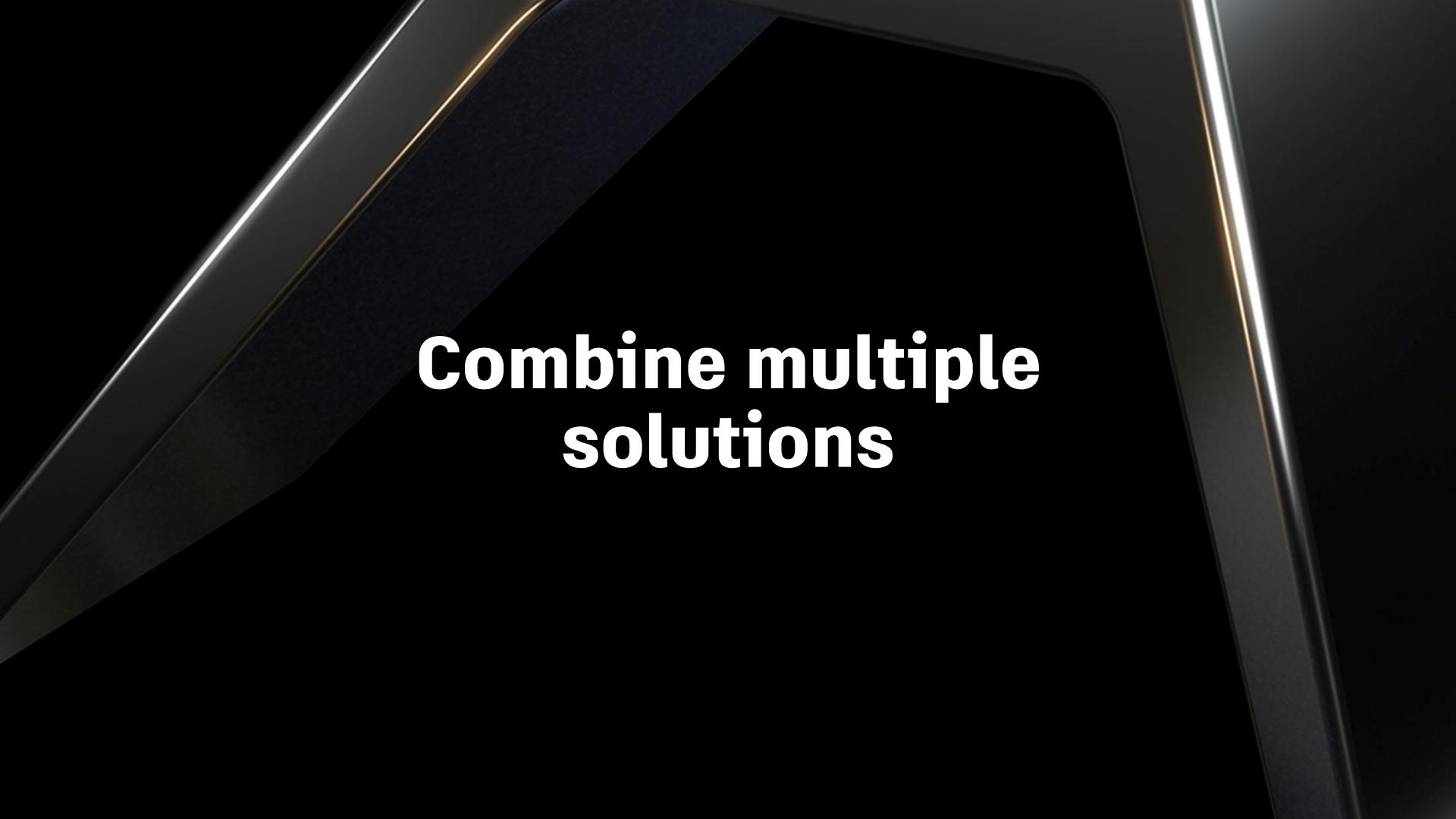
e.g. on the right side, you can see a simple control.



The screenshot shows a 'Control' dialog box with the following settings:

Parameter	Value
Platte	False
Schublade	False
Sockel	True
Breite	300 mm
Höhe	376 mm
Tiefe	560 mm
Griff	Oben

At the bottom right of the dialog is a 'Done' button.



**Combine multiple
solutions**

Combine iLogic rules with each other

You can combine iLogic rules

Reardless if it's another rule inside the same model, or a rule inside of an assembly component, you can combine/call rules.

It is possible to control the sequence of the program in the way you want.

```
iLogicVb.RunRule("Size")
iLogicVb.RunRule("Inlet")
iLogicVb.RunRule("Cases")
iLogicVb.RunRule("Reinigungsdeckel")
iLogicVb.RunRule("Outlet_No")
iLogicVb.RunRule("Outlet_Sizes")
iLogicVb.RunRule("Outlet_Inclination")
iLogicVb.RunRule("Cases_Delta")
iLogicVb.RunRule("Outlets")
iLogicVb.Ru Parameter("iL_Pos10", "dim_W") = dim_W
iLogicVb.Ru Parameter("iL_Pos10", "dim_H") = dim_H
iLogicVb.Ru
iLogicVb.Ru Parameter("iL_Pos20", "dim_W") = dim_W
iLogicVb.Ru Parameter("iL_Pos20", "dim_Angle") = dim_Ang
iLogicVb.Ru Parameter("iL_Pos30", "dim_W") = dim_W
iLogicVb.Ru Parameter("iL_Pos30", "dim_H") = dim_H
iLogicVb.Up Parameter("iL_Pos30", "dim_Angle") = dim_Ang
iLogicVb.Ru
Parameter("iL_Pos40", "dim_W") = dim_W
Parameter("iL_Pos40", "dim_H") = dim_H
Parameter("iL_Pos40", "dim_Angle") = dim_Ang
iLogicVb.UpdateWhenDone = True
```

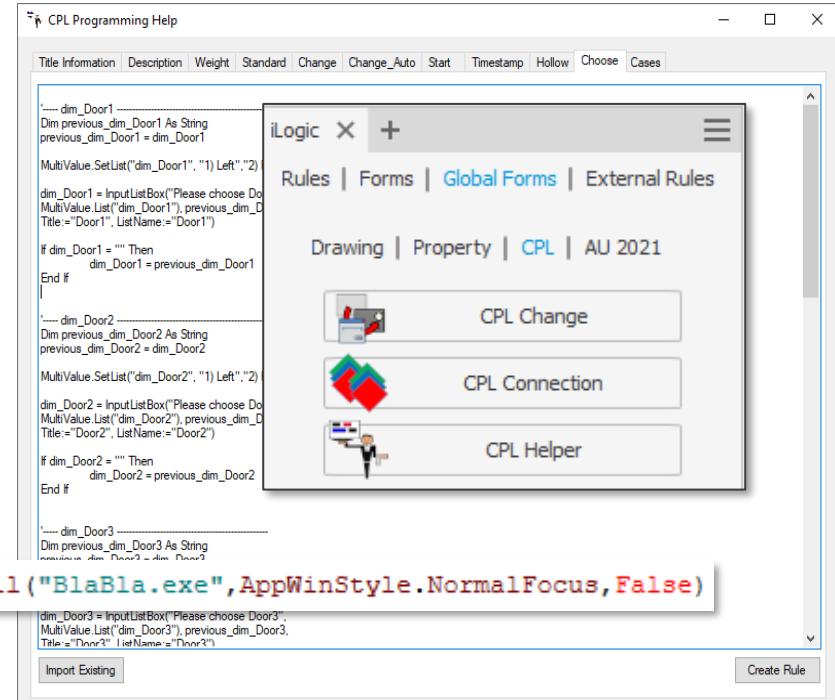
Combine iLogic Tech with other Technology

e.g. iLogic and .net

We're using external C# and vb.net tools by calling them out of the iLogic rules.

The iLogic rule only contain the call, the rest of the job is done by the external program.

For example, we also create iLogic Rules through an external tool.



Conclusion

**iLogic is a very
flexible tool**



AUTODESK UNIVERSITY

Autodesk und das Autodesk-Logo sind in den USA und/oder anderen Ländern eingetragene Marken oder Marken von Autodesk, Inc. und/oder seiner Tochterunternehmen und/oder verbundenen Unternehmen. Alle anderen Marken, Produktnamen und Kennzeichen gehören ihren jeweiligen Inhabern. Autodesk behält sich vor, Produkt- und Service-Angebote sowie Spezifikationen und Preise jederzeit ohne Vorankündigung zu ändern. Alle Angaben ohne Gewähr.