



OnProject

Automação de Criação de Caixas para Furos em Autodesk Inventor

Manual de Instalação e Execução da Solução

Conteúdo

Introdução à solução	2
Material Fornecido	2
Instalação	2
Exportação para o Excel	4
Opção “Hirarquia”	4
Opção “Só os componentes”	5
Cuidados a ter em conta	6
Script Desenvolvido	6
Regra “export excel”	6

Introdução à solução

A solução apresentada consiste num script desenvolvido pela TeamOn para colmatar necessidades específicas da ESI, pelo que este script é propriedade da ESI e não poderá ser replicado, vendido, doado ou partilhado pelas partes envolvidas no seu desenvolvimento sem o consentimento da ESI. A solução é um script desenvolvido no *Inventor Ilogic* denominado “resolver_caixas”. Este script é identificado pelo Inventor como regra (rule).

A regra “resolver_caixas” permite verificar se o diâmetro do parafuso coincide com o diâmetro da caixa onde este é inserido e se o comprimento do parafuso é maior que o comprimento da caixa onde está inserido. Caso os diâmetros não coincidam ou o parafuso apresente um comprimento maior que a caixa este será destacado e de seguida o diâmetro ou o comprimento da caixa será ajustado automaticamente.

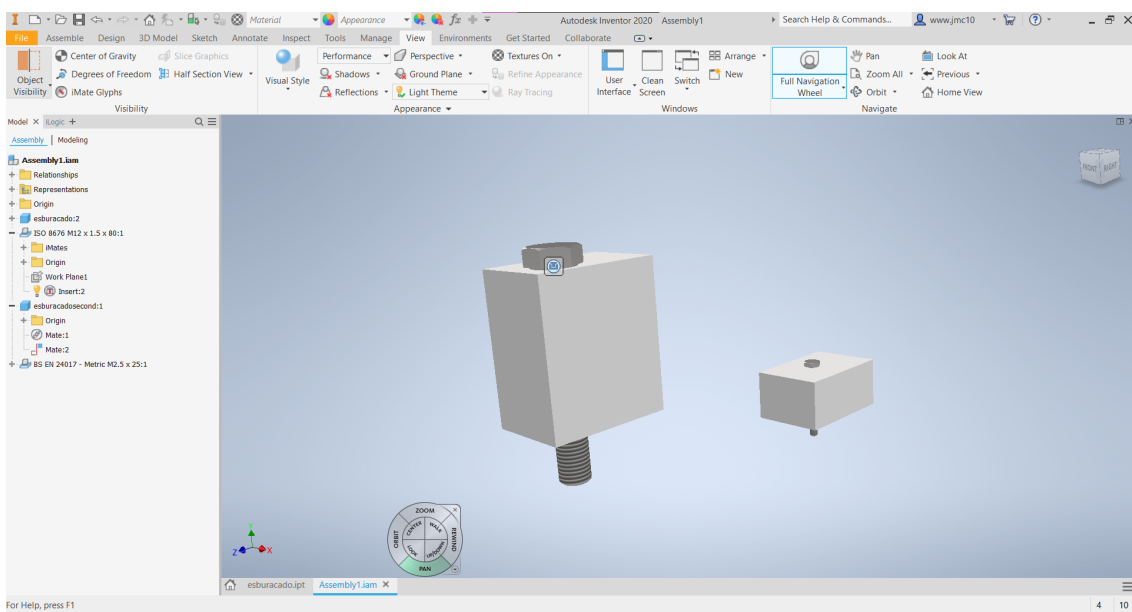
Material Fornecido

- Ficheiro “.txt” com a regra “resolver_caixas”
- Manual de instalação e execução da solução
- Ficheiro “.rar” contendo peça modelo

Instalação

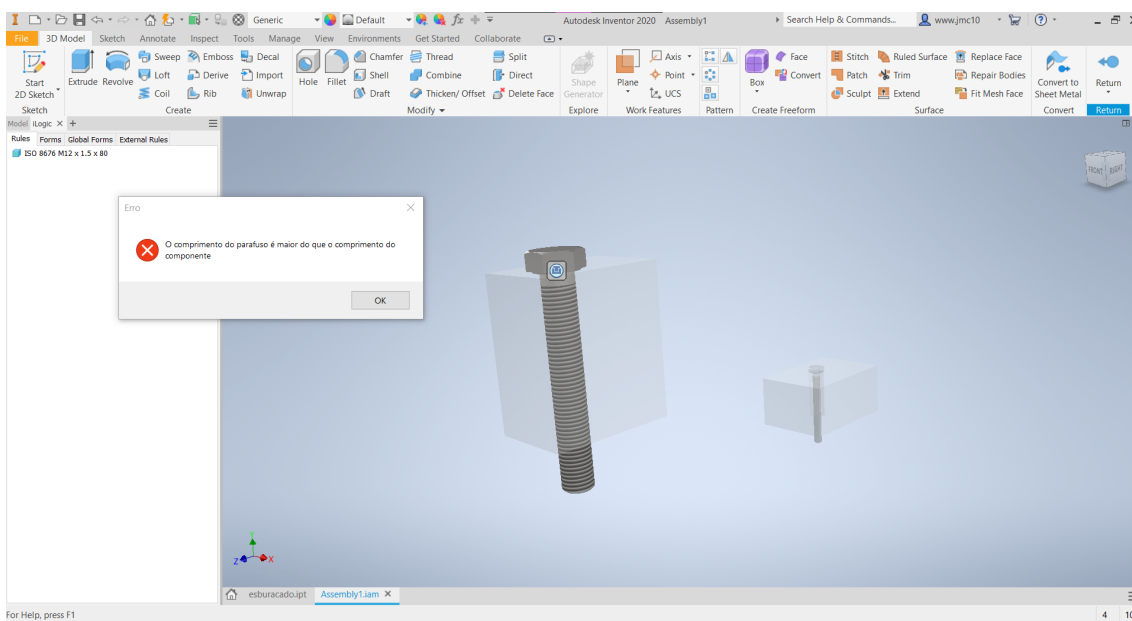
1. Abrir o ficheiro no qual se deseja aplicar a regra “resolver_caixas”.
2. Abrir a tab *Ilogic* e criar uma nova regra, nomeando-a como quiser.
3. Copiar os conteúdos do ficheiro “.txt” fornecido e colá-los na regra que acabou de ser criada.
4. Clicar no botão “Save” para guardar a regra ou no botão “Save & Run” para guardá-la e executá-la.

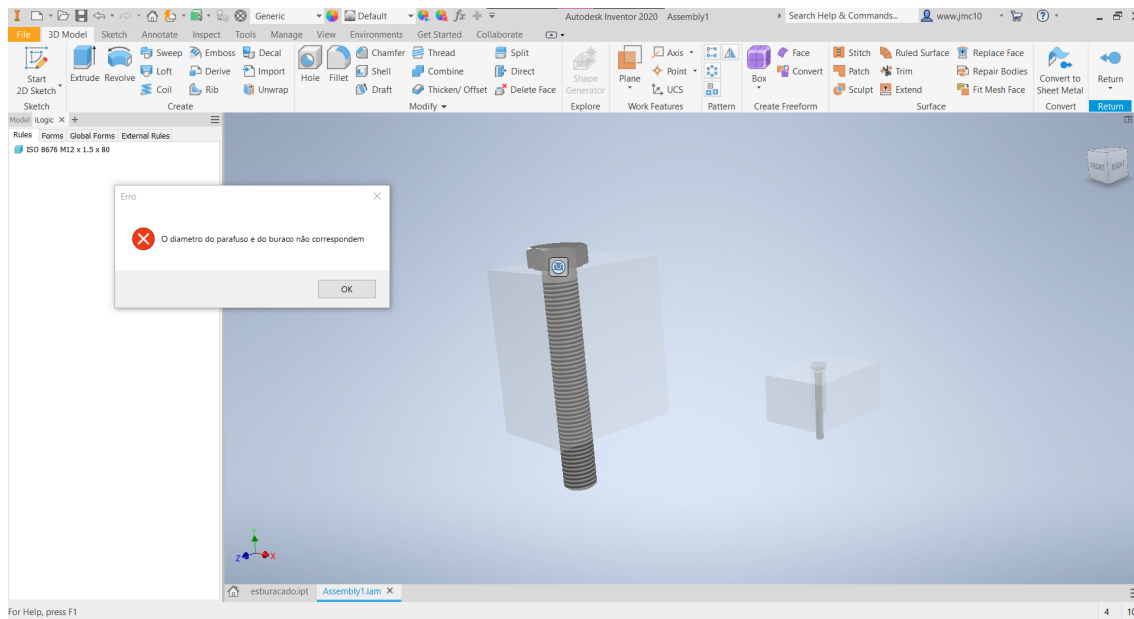
Verificação de dimensões de parafusos e caixas



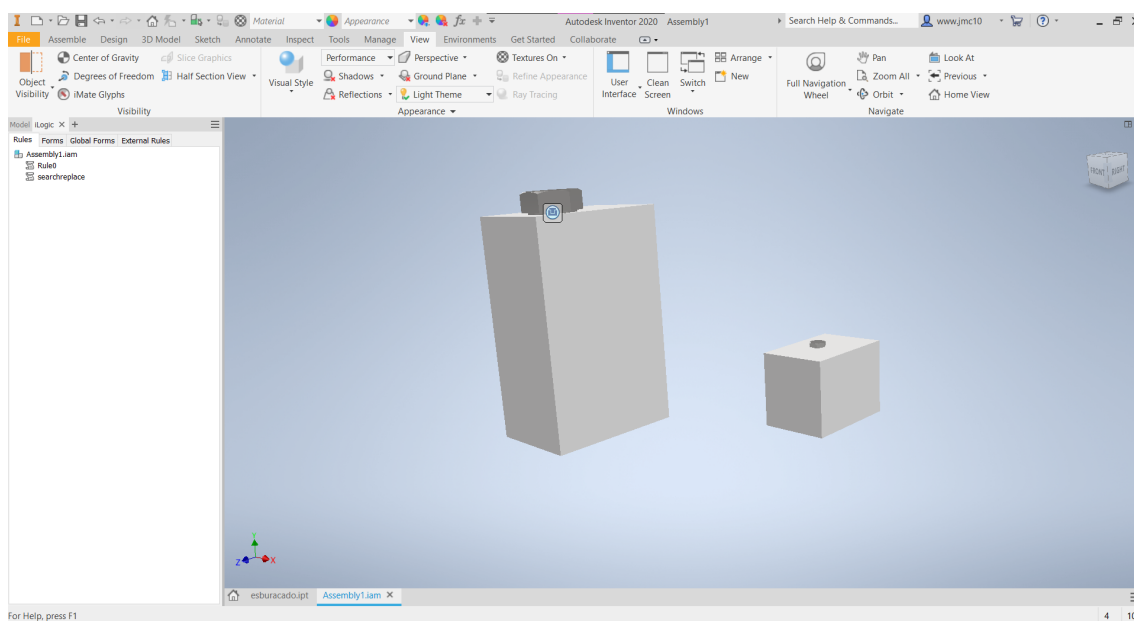
De modo a demonstrar a solução desenvolvida foram criados dois exemplos como apresentado na imagem. Em ambos os casos os diâmetros e comprimentos da caixa não correspondem ao diâmetro e comprimento do parafuso.

1. Executar a regra.
2. A regra irá encontrar o primeiro “erro” e destacar o mesmo. Irá reconhecer que o comprimento do parafuso é maior que o comprimento da caixa e que os diâmetros não correspondem.





3. De seguida a solução desenvolvida irá proceder ao ajuste das dimensões da caixa e seguir automaticamente para a correção do segundo parafuso.



Parameters							
Parameter Name	Consumed by	Unit/Type	Equation	Nominal Value	Tol.	Model Value	Key
Model Parameters							
DIA	Sketch1	mm	12 mm	12.000000		12.000000	
LEN	Extrusion1	mm	80 mm	80.000000		80.000000	
Q4	Extrusion1	deg	0.00 deg	0.000000		0.000000	
User Parameters							
<div> Add Numeric Update Purge Unused Reset Tolerance Link Immediate Update Done </div>							

Cuidados a ter em conta

A peça que contém as caixas dos furos tem de ter os parâmetros nomeados como DIA para o diâmetro do furo e LEN para o comprimento.

Só funciona com peças com um único furo pois não conseguimos distinguir os furos logo, para já, o script não consegue saber com qual furo deverá comparar o diâmetro do parafuso, caso a peça tenha vários furos, isto é, não conseguimos identificar a que furo corresponde cada parafuso.

Se existirem duas instâncias da mesma peça no mesmo assembly, ambas serão modificadas de acordo com o último parafuso.

Possíveis mudanças:

- o Otimizar esta regra para encontrar os erros e indicar os mesmos numa lista.
- o Criar uma nova regra que irá ajustar os casos errados de acordo com a lista.

Possíveis melhorias:

- o Criar um log file com todos os erros encontrados.
- o Tentar mudar o parafuso ao invés da caixa.
- o Verificar vários diâmetros de um só componente.
- o Escolher para cada parafuso “errado” o que fazer, via menus

Script Desenvolvido

Regra “export excel”

```
Dim assemblyDef As AssemblyComponentDefinition = ThisDoc.Document.ComponentDefinition
Dim oco As AssemblyConstraints = assemblyDef.Constraints
Dim occ As Inventor.ComponentOccurrence
For Each occ In assemblyDef.Occurrences.AllLeafOccurrences
    Dim refDoc As PartDocument = occ.Definition.Document
    'i = MessageBox.Show(occ.Name, "My iLogic Dialog", MessageBoxButtons.OK,
    MessageBoxIcon.Hand, MessageBoxDefaultButton.Button1)
    If iProperties.Value(occ.Name, "Project", "Description").Contains("bolt") Or
    iProperties.Value(occ.Name, "Project", "Description").Contains("Bolt") Or
    iProperties.Value(occ.Name, "Project", "Description").Contains("screw") Or
    iProperties.Value(occ.Name, "Project", "Description").Contains("Screw") Then
```

```
'par = Parameter(occ.Name, "d0")  
'i = MessageBox.Show(par, "My iLogic Dialog", MessageBoxButtons.OK,  
MessageBoxIcon.Hand, MessageBoxDefaultButton.Button1)
```

For Each myconstraint As AssemblyConstraint In occ.Constraints

```
        boltlength = Parameter(myconstraint.OccurrenceOne.Name,  
"NLG")  
        partlength = Parameter(myconstraint.OccurrenceTwo.Name,  
"LEN")  
  
        boltdiameter = Parameter(myconstraint.OccurrenceOne.Name,  
"NND")  
        partdiameter = Parameter(myconstraint.OccurrenceTwo.Name,  
"DIA")
```

```
        If boltlength > partlength  
            myconstraint.OccurrenceOne.Edit  
            i = MessageBox.Show("O comprimento do parafuso é  
maior do que o comprimento do componente", "Erro", MessageBoxButtons.OK,  
MessageBoxIcon.Hand, MessageBoxDefaultButton.Button1)  
            Parameter(myconstraint.OccurrenceTwo.Name, "LEN") =  
boltlength
```

End If

```
        If Not boltdiameter = partdiameter  
            myconstraint.OccurrenceOne.Edit  
            i = MessageBox.Show("O diametro do parafuso e do  
buraco não correspondem", "Erro", MessageBoxButtons.OK, MessageBoxIcon.Hand,  
MessageBoxDefaultButton.Button1)  
            Parameter(myconstraint.OccurrenceTwo.Name, "DIA") =  
boltdiameter
```

```
            'myconstraint.OccurrenceOne.Edit
```

End If

```
        'i = MessageBox.Show(myconstraint.Name, "My iLogic Dialog",  
MessageBoxButtons.OK, MessageBoxIcon.Hand, MessageBoxDefaultButton.Button1)  
        'i = MessageBox.Show(myconstraint.OccurrenceOne.Name, "My  
iLogic Dialog", MessageBoxButtons.OK, MessageBoxIcon.Hand,  
MessageBoxDefaultButton.Button1)
```

```
        'i = MessageBox.Show(boltlength, "My iLogic Dialog",  
MessageBoxButtons.OK, MessageBoxIcon.Hand, MessageBoxDefaultButton.Button1)
```

OnProject - Automação de Lista de Peças

Manual de Instalação e Execução da Solução



```
'i = MessageBox.Show(myconstraint.OccurrenceTwo.Name, "My
iLogic Dialog", MessageBoxButtons.OK, MessageBoxIcon.Hand,
MessageBoxDefaultButton.Button1)
'i = MessageBox.Show(partlength, "My iLogic Dialog",
MessageBoxButtons.OK, MessageBoxIcon.Hand, MessageBoxDefaultButton.Button1)
'i = MessageBox.Show(Parameter(OccurrenceTwo,"d3"), "My
iLogic Dialog", MessageBoxButtons.OK, MessageBoxIcon.Hand,
MessageBoxDefaultButton.Button1)
Next
occ.Visible = True
'i = MessageBox.Show(occ.Name, "My iLogic Dialog",
MessageBoxButtons.OK, MessageBoxIcon.Hand, MessageBoxDefaultButton.Button1)

Else
occ.Visible = True
End If

'If iProperties.Value(occ.Name,"Project", "Part Number").Contains("ISO") Then
'occ.Visible = True
'i = MessageBox.Show("The Value you entered was incorrect", "My
iLogic Dialog", MessageBoxButtons.OK, MessageBoxIcon.Hand,
MessageBoxDefaultButton.Button1)

' Else
'i = MessageBox.Show("The Value you entered was incorrect", "My iLogic
Dialog", MessageBoxButtons.OK, MessageBoxIcon.Hand,
MessageBoxDefaultButton.Button1)

'occ.Visible = True
'End If
Next
```