#### Moldflow Summit 2018: Automation with API

Matt Jaworski Sr. Subject Matter Expert, Moldflow

Ana Maria Marin TZERO<sup>®</sup> Consultant/Trainer - RJG, Inc.

AUTODESK.

#### **RJG Global Offices**

Servicing the Plastics Industry since 1989

RJG USA Traverse City, MI (RJG, Inc. Headquarters) <u>RJG Italy</u> Next Innovation Srl Milano, Italy

Woodstock, GA (Regional Training Center)

RJG (S.E.A.) PTE LTD Republic of Singapore

**Gibsonville, NC** (Regional Training Center)

RJG Mexico Chihuahua, Mexico

RJG France Arinthod, France RJG Korea CAEPro Seoul, Korea

RJG China Chengdu, China

RJG Germany Karlstein, Germany

**RJG Ireland / UK** Co Tipperary, Ireland



Successful Molding is Based on an Understanding of the Entire Process "Global"



# Half the Battle is Knowing What is Happening in the Mold

**Process Control Systems and Cavity Pressure Sensing Technology** 



## What is TZERO<sup>®</sup>?

Validation Between Engineering and Manufacturing: A Systematic Process

#### GLOBAL

End-to End Services

- Consulting
- Simulation
- Training

Single Source Solutions Provider for Plastic Injection Molding





# **API Background**

## What is the Autodesk Moldflow Insight API?

- API = <u>Application Programming Interface</u>
- An Object Linking and Embedding (OLE) programming interface that allows AMI functionality to be automated
- Manipulation of AMI is done through scripts or third party software
- Functionality available since MPI 4.0 (2002)
- Additional functionality added in each major Insight release





#### What Does the API Do?

- Increases user productivity for repetitive tasks
- Customizes the UI, result plots & solvers
- Supports 3rd party add-on products
- Enhances integration to other applications
  - ERP, CAD/CAE, MS Office Excel/Word/PPT
- Supports university research programs
- Standardizes corporate protocols and best practices
- Supports industry data formats





# Object Linking and Embedding (OLE) Introduction

- An OLE automation client is needed to control the API though <u>Autodesk Moldflow Insight</u> (AMI)
  - Visual Basic Script or vbScript (VBS)
    - AMI records all scripts in Visual Basic Script
    - The only OLE that Autodesk offers help/support for
- Other Potential Interfaces
  - JScript and other programming languages
  - Visual Basic for Applications (VBA)
  - Visual Basic (VB)
  - Perl
  - Python
  - ActiveX scripts in Internet Explorer



### Object Linking and Embedding (OLE) Introduction

 The OLE automation interface will automatically use the version of Synergy (AMI's GUI) that was most recently executed on your computer

 If you last opened an early version of AMI/MPI and try to run a macro or script that uses features that are not supported, your macro or script will generate an error

### **Running Multiple Synergy Instances**

 Prior to the 2016 Release it was only possible to run 1 API instance across all Synergy Instances

- API now works with multiple Synergy instances
  - You can launch a macro for a specific instance of Synergy on the local machine, using the InstanceID that you can find in the Help > About box



# **Running Multiple Synergy Instances**

- Edit Macros recorded/written prior to the 2016
  - Add the following code segment

'%RunPerInstance

Must be first line to avoid this message

```
Automation Unavailable
Automation Unavailable
Macro recording and playback, the command line and DLE automation
are not available in this Synargy session since there is another instance of
the application already active with these features enabled
```

Dim SynergyGetter, Synergy

On Error Resume Next

Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%")) On Error GoTo 0

If (Not IsEmpty(SynergyGetter)) Then

```
Set Synergy = SynergyGetter.GetSASynergy
```

Else

```
Set Synergy = CreateObject("Synergy.Synergy")
End If
```

Code to Ensure Correct Instance is executed

## **API On-line Help**

- Valuable resource for the beginner or expert user
- Accessed via
  - Help > Automation > Synergy Application Programming Interface (API)



## Example Help VB Scripts

- Nice starting scripts located in Examples section
- Good variety of common uses
- Accessed via
  - Help > Automation > Examples

#### AUTODESK<sup>®</sup> MOLDFLOW<sup>®</sup> INSIGHT 2018

#### 🕒 Get Started

- Model Preparation
- Types of Analysis
- Materials and Databases
- Processing Conditions
- 🕒 Simulation Job Manager
- Optimization analyses
- Automation
  - + Command line control
  - Synergy Application Programming Interface (API)
    - The OLE Automation Interface
  - 🕂 Macros
  - Limitations in API functionality
  - + Command line and Visual Basic scripts
  - + Examples

#### Examples

The following are examples of API scripts that allow you to automate this product.

#### Topics in this section

- API example: The first lines of a script
- API example: Customized aspect ratio plot
- API example: Showing thicknesses within a range
- API example: Reading pressure data
- API example: Looping through entities
- $\cdot\,$  API example: The minimum, maximum, average of an entity list
- API example: Writing nodal data to a file
- API example: Creating multiple drops
- API example: Remove gas core
- API example: Extract information from screen out file

Parent topic: Synergy Application Programming Interface (API)

#### Synergy API Reference Manual



#### **Class Member**



#### **TCode Reference**

- A TCode corresponds to a single feature in the process settings, geometry or solver parameters
- When programming with the Moldflow API, you need to know the numeric ID corresponding to a particular solver or modeling feature

(→) ② C:\Program Files\Autodesk\Moldflow Synergy 2018.2\help\tcode-reference\index.html	- C Search
A Moldflow Insight 2018 Help: E 🧭 Moldflow tcode reference 🛛 🗙	
× Select	× Google
Convert      Select     toode reference     100 Number of laminae across thickness     180 Write filling phase regular results at     181 Write filling phase regular results at     182 Write packing phase regular results at     184 Dynamically update results at     184 Dynamically update results at     198 Filling phase profiled results at     198 Filling phase     200 Number of regular results     201 Number of profiled results     202 Number of regular results     203 Number of profiled results     204 Melt menterature convergence tolerance     305 Fiber orientation convergence tolerance     305 Mold-melt Heat Transfer Coefficient (HTC) option     309 Mold-melt Heat Transfer Coefficient     311 Flow to produe interface for Process Optimization     311 Stold/melt Heat Transfer Coefficient (HTC) varial     311 Stold/melt Heat Transfer Coefficient (HTC) varial     311 Flow to produe interface for Process Optimization     311 Stold/melt Heat Transfer Coefficient (HTC) varial     311 Flow to produe interface for Process Optimization     311 Stold/melt Heat Transfer Coefficient (HTC) varial     311 Flow to produe interface for Process Optimization     311 Stold/melt Heat Transfer Coefficient (HTC) varial     311 Flow to produe interface for Process Optim	Coogle     Coogle
313 Mold temperature convergence tolerance	Referenced by tcodeset 10074 Reactive injection-compression molding solver parameters (Midplane)
<ul> <li>514 transfert mold temperature convergence tolerance for each time step</li> </ul>	1

#### **TCodeset Reference**

- A **TCodeset** comprises the TCodes which together control one aspect of a solver
- When programming with the Moldflow API, you need to know the numeric ID corresponding to a particular solver or modeling feature



#### Vbscript Editors/Debuggers Summary

Notepad++

Editor Only

vbsedit

Editor/Debugger

Microsoft Visual Studio

Editor/Debugger

#### Notepad ++



- Great tool for writing and editing scripts
- Really an editor only
- Nice features compared to regular Notepad

Free

File Gait Search View Encoding Language Setting: Macro Run Plugins Window ? X	C:\AU2015\API Training 1\Sample Scripts\ExportSpecifiedToMFR.vbs - Notepad++	- C X
Image: Section of the section of th	File Edit Search View Encoding Language Settings Macro Run Plugins Window ?	Х
Energy Id         EpotSpecifiedToMFR.vbs         Id           1         **KunPerInstance         **           2         **         **           3         ** DESCRIPTION         **           4         **         Loop through all study files. Mark specified result in these studies           5         **         **           6         **         StyTEX           7         **         EXportSpecifiedToMFR           8         **         **           9         **         PARAMETERS           10         **         **           12         **         DEFENDENCIES/LIMITATIONS           13         **         Script only has basic Error Handling           14         **         **           15         **         History           16         **         Created DRA and SK 05/11/2015           17         **         **           18         Option Explicit         **           19         Dim SynergyGetter, Synergy         **           21         Set Synergy Synergy         **           22         Dim SynergyGetter, GetSobject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance*"))           24<	[ ] 🖶 🖶 🖕 [ ] ] ↓ 🛍 [ ] ] → ⊄   # 🍇   🤏 🦂   💁 ¶ [ ] ] 🖉 🖉 🔊   1 [ ] ] [ ] ] ↓ ] ● 🗉 🕑 📓	
1       *BunPerInstance       *         2       *         3       *B DESCRIPTION         4       * Loop through all study files. Mark specified result in these studies         5       *         6       SYNTAX         7       *E ExportSpecifiedToMFR         *       *         9       *PARAMETES         10       *         11       *         12       * DEPENDENCIES/LIMITATIONS         13       * Script only has basic Error Handling         14       *         15       *         16       * Created DRA and SK 05/11/2015         17       *         18       Option Explicit         19       *         20       Dim FS         21       Set FS = CreateObject("Scripting.FileSystemObject")         22       Dim SynergyGetter, Synergy         23       Dim SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance*"))         26       On Error GoTo 0         27       *         28       Set Synergy = SynergyGetter.GetSASynergy         29       Else         33       *Loop throup all studies <t< td=""><td>change.log 🗵 🖶 ExportSpecifiedToMFR.vbs 🗵</td><td></td></t<>	change.log 🗵 🖶 ExportSpecifiedToMFR.vbs 🗵	
2       '6         3       '6         4       '6         1       1000 through all study files. Mark specified result in these studies         5       '6         6       '8 SYNTAX         7       '6         7       '6         9       '7         9       '7         10       '6         11       '6         12       '6         14       '6         15       '8         16       '8         17       '8         7       '8         7       '8         16       '8         17       '8         18       Option Explicit         19       '9         20       Dim F5         21       Set FS = CreateObject("Scripting.FileSystemObject")         22       Dim SynergyGetter, Synergy         23       Oin Error Resume Next         24       Set Synergy = SynergyGetter.) Then         25       Set Synergy = SynergyGetter.) Then         26       Set Synergy = CreateObject("SynergyGetter.) Synergy")         21       End If         33       'Lo	1 '\$RunPerInstance	A
3       *0       DESCRIPTION         4       *1       Coop through all study files. Mark specified result in these studies         4       *0       SWIRX         7       *0       EXportSpecificdToMFR         8       *0         9       *1       PERAMETERS         10       *0       E         11       *0         12       *0       DEPROENCIES/LIMITATIONS         13       *0       Script only has basic Error Handling         4       *0       E         10       *0       E         14       *0       E         15       *1       Script only has basic Error Handling         4       *0       E         14       *0       E         15       *1       Script only has basic Error Handling         16       *1       Created DRA and SK 05/11/2015         17       *16       *1         20       Dim FS       E         21       Set FS = CreateObject("Scripting.FileSystemObject")         22       Dim SynergyGetter, Synergy         23       Dim SynergyGetter, Synergy         24       Set Synergy = SynergyGetter.) Then <t< td=""><td>2 '0</td><td></td></t<>	2 '0	
<pre>4 'E Loop through all study files. Mark specified result in these studies 5 'e 6 '8 SYNTAX 7 'e EXportSpecificdToMFR 9 'P PARAMETERS 10 'e none 11 'e 12 'e DEPENDENCIES/LIMITATIONS 13 'e Script only has basic Error Handling 14 'e 15 'e History 16 'e Created DRA and SK 05/11/2015 7 'e 18 Option Explicit 19 20 Dim FS 21 Set FS = CreateObject("Scripting.FileSystemObject") 22 23 Dim SynergyGetter, Synergy 24 On Error Resume Next 25 Set SynergyGetter, Synergy 24 On Error GoTo 0 7 Elf (Not 1sEmpt(SynergyGetter)) Then 28 Set Synergy = CreateObject("Synergy.Synergy") 29 Else 30 Set Synergy = CreateObject("Synergy.Synergy") 31 End If 32 33 4 'Loop throup all studies 35 Dim Project 4 Visual Basic file length:1873 line: 79 Ln:1 Col:1 Sel:010 DosWindows UTF-8 INS // 35 Set Synergy 4 36 Set Synergy 4 37 Set Synergy 4 38 Set Synergy 4 39 Else 30 Set Synergy 5 30 Set Synergy 5 30 Set Synergy 5 31 End If 32 33 34 'Loop throup all studies 35 Dim Project 4 4 4 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5</pre>	3 '@ DESCRIPTION	
5       '9         6       '9         6       '9         7       '9         7       '9         10       '9         11       '9         12       '9         13       '9         14       '9         15       '9         16       '9         17       '9         18       Option Explicit         20       Dim FS         21       Set FS = CreateObject("Scripting.FileSystemObject")         22       Dim SynergyGetter, Synergy         24       On Error Resume Next         25       Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance*"))         26       On Error GoTo 0         27       Tf (Mot IsEmpty(SynergyGetter)) Then         28       Set Synergy = SynergyGetter.GetSASynergy         29       Else         30       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         32       'Loop throup all studies         33       'Loop throup all studies         33       'Loop throup all studies         34       'Loop throup all studies	4 'G Loop through all study files. Mark specified result in these studies	
<pre>v SINIAA v Standard Stand</pre>		
<pre>v = PARAMETERS * 9 * 9 PARAMETERS 10 'e none 11 'e 12 'e DEFENDENCIES/LIMITATIONS 13 'e Script only has basic Error Handling 14 'e 15 'e History 16 'e Created DRA and SK 05/11/2015 17 'e 18 Option Explicit 19 Dim FS 20 Dim FS 21 Set FS = CreateObject("Scripting.FileSystemObject") 22 Dim SynergyGetter, Synergy 24 On Error Resume Next 25 Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%")) 26 On Error Golo 27 Elf (Not IsEmpty(SynergyGetter)) Then 28 Set Synergy = CreateObject("Synergy.Synergy") 29 Ellee 30 Set Synergy = CreateObject("Synergy.Synergy") 31 End If 32 33 * Loop throup all studies 35 Dim Project * ***********************************</pre>	1 G EVENTSTOOT FLOOT FLO	
<pre>9 *@ PARAMETERS 10 *@ none 11 *@ 12 *@ DEFENDENCIES/LIMITATIONS 13 *@ Script only has basic Error Handling 14 *@ 15 *@ History 16 *@ Created DRA and SK 05/11/2015 17 *@@ 20 Dim FS 21 Set FS = CreateObject("Scripting.FileSystemObject") 22 23 Dim SynergyGetter, Synergy 24 On Error Resume Next 25 Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%")) 26 On Error GoTO 0 27 Elf (Not IsEmpty(SynergyGetter)) Then 28 Set Synergy = CreateObject("Synergy.Synergy") 29 EElse 30 Set Synergy = CreateObject("Synergy.Synergy") 31 End If 32 33 'Loop throup all studies 35 Dim Project * Visual Basic file length:1873 lines:79 Ln:1 Col:1 Sel:010 Dos\Windows UTF-8 INS in </pre>	e 18	
<pre></pre>	0 10 DADAMFTEDS	=
<pre>11 '% 12 '% DEPENDENCIES/LIMITATIONS 13 '% Script only has basic Error Handling 14 '% 15 '% History 16 '% Created DRA and SK 05/11/2015 17 '%% 18 Option Explicit 19 20 Dim FS 21 Set FS = CreateObject("Scripting.FileSystemObject") 22 23 Dim SynergyGetter, Synergy 24 On Error Resume Next 25 Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%")) 26 On Error GoTo 0 27 ⊟If (Not IsEmpty(SynergyGetter)) Then 28 L Synergy = SynergyGetter.GetSASynergy 29 ⊟Else 30 L Set Synergy = CreateObject("Synergy.Synergy") 31 End If 32 33 4 'Loop throup all studies 35 Dim Project VisualBasic file length:1873 lines:79 Ln:1 Col:1 Sel:010 DosWindows UTF-8 INS minimation </pre>		
12       '0 DEPENDENCIES/LIMITATIONS         13       '0 Script only has basic Error Handling         14       '0         15       '0 History         16       '0 Created DRA and SK 05/11/2015         17       '00         20       Dim FS         21       Set FS = CreateObject("Scripting.FileSystemObject")         22       Dim SynergyGetter, Synergy         24       On Error Resume Next         25       Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%"))         26       On Error GoTo 0         27       □1 f(Not IsEmpty(SynergyGetter)) Then         28       Set Synergy = SynergyGetter.GetSASynergy         29       □Else         30       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         32          33       'Loop throup all studies         35       Dim Project         36       Loop throup all studies         35       Dim Project		
13       '0 Script only has basic Error Handling         14       '0         15       '0 History         16       '0 Created DRA and SK 05/11/2015         17       '00         18       '0 Created DRA and SK 05/11/2015         17       '00         18       '0 Created DRA and SK 05/11/2015         19       '0         20       Dim FS         21       Set FS = CreateObject("Scripting.FileSystemObject")         22       Dim SynergyGetter, Synergy         23       Dim SynergyGetter, Synergy         24       On Error GoTo 0         27       EIf (Not IsEmpty(SynergyGetter)) Then         28       Set Synergy = SynergyGetter.GetSASynergy         29       ELse         30       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         32       'Loop throup all studies         33       'Loop throup all studies         33       Dim Project         34       'Loop throup all studies         35       Dim Project	12 '@ DEPENDENCIES/LIMITATIONS	
14       '%         15       '%         16       '%         16       '%         17       '%         18       Option Explicit         19         20       Dim FS         21       Set FS = CreateObject("Scripting.FileSystemObject")         22         23       Dim SynergyGetter, Synergy         24       On Error Resume Next         25       Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%"))         26       On Error GoTo 0         27       □If (Not IsEmpty(SynergyGetter)) Then         28       Set Synergy = SynergyGetter.GetSASynergy         29       □Else         30       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         32       Thoop throup all studies         33       'Loop throup all studies         35       Dim Project         26       Dim Kindows         27       Ursuel Basic file         28       In:1 Col:1 Sel:0 0         29       District Set Synergy	13 '9 Script only has basic Error Handling	
<pre>15 '@ History 16 '@ Created DRA and SK 05/11/2015 17 '@@ 18 Option Explicit 19 20 Dim FS 21 Set FS = CreateObject("Scripting.FileSystemObject") 22 23 Dim SynergyGetter, Synergy 24 On Error Resume Next 25 Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%")) 26 On Error GoTo 0 27 Elf(Not IsEmpty(SynergyGetter)) Then 28 Set Synergy = SynergyGetter.GetSASynergy 29 Else 30 Set Synergy = CreateObject("Synergy.Synergy") 31 End If 32 33 34 'Loop throup all studies 35 Dim Project 7 Visual Basic file length:1873 lines:79 Ln:1 Col:1 Sel:010 Dos/Windows UTF-8 INS mathematical studies 35 Set Synergy = Synergy Synerg Synergy Synergy Synergy Synerg Synergy Synergy Synerg Syner</pre>	14 '0	
16       '% Created DRA and SK 05/11/2015         17       '% 0         18       Option Explicit         19       Dim FS         21       Set FS = CreateObject("Scripting.FileSystemObject")         22       Dim SynergyGetter, Synergy         24       On Error Resume Next         25       Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%"))         26       On Error GoIo 0         27       FIf (Not IsEmpty(SynergyGetter.) Then         28       Set Synergy = SynergyGetter.GetSASynergy         29       Else         30       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         32       Set Supergot = CreateObject("Synergy.Synergy")         31       End If         32       Set Supergot = CreateObject ("Synergy.Synergy")         33       'Loop throup all studies         34       'Loop throup all studies         35       Dim Project         36       Image: Set Stripting Set	15 '@ History	
17       '88         18       Option Explicit         19       Dim FS         21       Set FS = CreateObject("Scripting.FileSystemObject")         22       Dim SynergyGetter, Synergy         24       On Error Resume Next         25       Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%"))         26       On Error GoTo 0         27       EIf (Not IsEmpty(SynergyGetter)) Then         28       Set Synergy = SynergyGetter.GetSASynergy         29       ELse         30       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         32       Dim Project         34       'Loop throup all studies         35       Dim Project         34       End If         35       Dim House         36       Dim Windows         37       UTF-8	16 '@ Created DRA and SK 05/11/2015	
18       Option Explicit         19       Dim FS         21       Set FS = CreateObject("Scripting.FileSystemObject")         22       Dim SynergyGetter, Synergy         23       Dim SynergyGetter, Synergy         24       On Error Resume Next         25       Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%"))         26       On Error GoTo 0         27       □ff (Not IsEmpty(SynergyGetter)) Then         28       Set Synergy = SynergyGetter.GetSASynergy         29       □Else         30	17 '00	
19         20       Dim FS         21       Set FS = CreateObject("Scripting.FileSystemObject")         23       Dim SynergyGetter, Synergy         24       On Error Resume Next         25       Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%"))         26       On Error GoTo 0         27       Fif (Not IsEmpty(SynergyGetter)) Then         28       Set Synergy = SynergyGetter.GetSASynergy         29       Else         30       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         33       'Loop throup all studies         35       Dim Project         34       'Loop throup all studies         35       Dim Project	18 Option Explicit	
20       Dim FS         21       Set FS = CreateObject("Scripting.FileSystemObject")         22       Dim SynergyGetter, Synergy         23       Dim SynergyGetter, Synergy         24       On Error Resume Next         25       Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%"))         26       On Error GoTo 0         27       EIf (Not IsEmpty(SynergyGetter)) Then         28       Synergy = SynergyGetter.GetSASynergy         29       ELse         30       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         32       Toop throup all studies         33       'Loop throup all studies         34       'Loop throup all studies         35       Dim Project         26       Point file         27       Visual Basic file         Visual Basic file       length:1873 lines:79         26       Dos/Windows       UTF-8	19	
21       Set FS = CreateObject ("Scripting.FileSystemObject")         22       Dim SynergyGetter, Synergy         23       Dim SynergyGetter, Synergy         24       On Error Resume Next         25       Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%"))         26       On Error GoTo 0         27       EIf (Not IsEmpty(SynergyGetter)) Then         28       Synergy = SynergyGetter.GetSASynergy         29       Else         30       L Set Synergy = CreateObject("Synergy.Synergy")         31       End If         32       Stop throup all studies         35       Dim Project         26       Dim String [hength:1873 lines:79         27       Visual Basic file	20 Dim FS	
22       Dim SynergyGetter, Synergy         23       Dim SynergyGetter, Synergy         24       On Error Resume Next         25       Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%"))         26       On Error GoTo 0         27       If (Not IsEmpty(SynergyGetter)) Then         28       Set Synergy = SynergyGetter.GetSASynergy         29       Else         30       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         33       'Loop throup all studies         34       'Loop throup all studies         35       Dim Project         Visual Basic file       length:1873 lines:79         Visual Basic file       length:1873 lines:79	<pre>21 Set FS = CreateObject("Scripting.FileSystemObject")</pre>	
24       On Error Resume Next         25       Set SynergyGetter = GetObject (CreateObject ("WScript.Shell").ExpandEnvironmentStrings ("%SAInstance%"))         26       On Error GoTo 0         27       Elf (Not IsEmpty(SynergyGetter)) Then         28       Set Synergy = SynergyGetter.GetSASynergy         29       Else         30       Set Synergy = CreateObject ("Synergy.Synergy")         31       End If         32		
25       Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentStrings("%SAInstance%"))         26       On Error GoTo 0         27       If (Not IsEmpty(SynergyGetter)) Then         28       Set Synergy = SynergyGetter.GetSASynergy         29       Else         30       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         32       Set Synergy = CreateObject("Synergy.Synergy")         34       'Loop throup all studies         35       Dim Project         26       Visual Basic file         Visual Basic file       length:1873 lines:79         Ln:1       Col:1         Set Solo       Dos\Windows	23 Dim SynergyGetter, Synergy	
26     On Error GoTo 0       27     □ If (Not IsEmpty(SynergyGetter)) Then       28	24 On Effor Resume Next	
27       [If (Not IsEmpty(SynergyGetter)) Then         28       Set Synergy = SynergyGetter.GetSASynergy         29       [Else         30       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         32	25 Set SynergyGetter = Getobject(CreateObject(-WSCript.Sherr)):ExpandEnvironmentStrings("SAThStances"))	
28       Set Synergy = SynergyGetter.GetSASynergy         29       Else         30       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         32       33         34       'Loop throup all studies         35       Dim Project         Visual Basic file       length:1873 lines:79         Ln:1       Col:1         Set:010       Dos\Windows         UTF-8       INS	27 OF FIGURE STORE (Supergradetter)) Then	
29       Else         30       Set Synergy = CreateObject ("Synergy.Synergy")         31       End If         32       33         34       'Loop throup all studies         35       Dim Project         Visual Basic file         length:1873       lines:79         Ln:1       Col:1         Sel:0       Dos/Windows         UTF-8       INS	28 Set Subarguy (synergy Getter, GetSASupergy	
30       L       Set Synergy = CreateObject("Synergy.Synergy")         31       End If         32       33         34       'Loop throup all studies         35       Dim Project         Yisual Basic file         length:1873 lines:79       Ln:1 Col:1 Sel:0 0         Dos\Windows       UTF-8	29 Else	
31       End If         32       33         33       'Loop throup all studies         35       Dim Project         Visual Basic file       length:1873 lines:79         Ln:1       Col:1         Sel:0       0         Dos/Windows       UTF-8	30 T Set Synergy = CreateObject ("Synergy.Synergy")	
32         33         34       'Loop throup all studies         35       Dim Project         Visual Basic file       length:1873 lines:79         Ln:1       Col:1         Sel:0       0         Dos/Windows       UTF-8	31 End If	
33       34     'Loop throup all studies       35     Dim Project       Visual Basic file     length:1873 lines:79       Ln:1     Col:1       Sel:0     Dos\Windows       UTF-8     INS	32	
34     'Loop throup all studies       35     Dim Project       Visual Basic file     length:1873 lines:79     Ln:1 Col:1 Sel:0 0     Dos\Windows     UTF-8     INS	33	
35         Dim Project         v           Visual Basic file         length:1873 lines:79         Ln:1 Col:1 Sel:0 0         Dos\Windows         UTF-8         INS	34 'Loop throup all studies	
Visual Basic file length : 1873 lines : 79 Ln : 1 Col : 1 Sel : 0   0 Dos\Windows UTF-8 INS	35 Dim Project	-
	Visual Basic file length : 1873 lines : 79 Ln : 1 Col : 1 Sel : 0   0 Dos\Windows UTF-8	INS

#### Notepad ++



- Great tool for writing, debugging and editing scripts
- Free version available
- Does have nagging timer to get you to buy

ReadMesh.vbs - Vbs8dit			
File Edit View Debug Snippets Samples Tools Help			
🗋 🗃 🖬 🕘 🖇 🖏 🖏 🕫 🕫 🦉 📷 📷 ன 🎯 🍇 🎕 😤 Help 💷 💭 🐺 🐺 🗐 😇 🛀	🗋 🚰 🖟 🖄 🖕 🖄 🔊 🗢 🔄 Shippets * 📓 🗐 🚔 🎕 🥵 😰 Help 📖 💭 👒 👰 🔅 接目 🛄 💭		
🕨 🕨 Start 🖩 🖩 🖗 🚰 📕 🌽 🌘 🍈 WScript 🧰 CScript 📺	-		
1 *&RunPerInstance	Object Browser		
2 '8	<search> 💌 🖌</search>		
3 '@ DESCRIPTION	E H- Mt Beam		
4 '% This script will read mesh data	GetTempFile		
9° C	m. 🕂 MeshClass		
7 '9 Beadweah	m. 42 Node		
8 • 9	SmartSplit		
9 '@ PARAMETERS	m Ar TETA		
10 '@ None	de TRIS		
11 '8	in the		
12 '@ DEPENDENCIES/LIMITATIONS	WsbRuntimel ibran/		
13 '8 none	Scripting		
	in fill whit Charge		
10 'ee Employie	(II) (III) WINE CLOSSES		
to option Expirit			
19 Dim SynergyGetter, Synergy			
20 On Error Resume Next			
21 Set SynergyGetter = GetObject(CreateObject("WScript.Shell").ExpandEnvironmentSTetngs("%SAInstance%"))			
22 On Error Goto 0			
23 If (Not IsEmpty(SynergyGetter)) Then			
24 Set Synergy = SynergyGetter.GetSASynergy			
25 Else			
<pre>2b Set Synergy = Create(b)ect("Synergy") 27 Ford If</pre>			
<pre>29 Set FS = CreateObject("Scrinting.FileSystemObject")</pre>			
30			
31			
32 Dim StudyDoc			
33 Set StudyDoc = Synergy.StudyDoc			
34			
35 ' read the mesh of the current study file			
36 Dim Mesh			
37 Set mesh - New Meshclass			
39 real-readdone cr3 ()			
40 Dim 1Str			
41 1Str = "Number of Nodes = " & Mesh.GetNumNode() & " Highest Node number = " & Mesh.GetHighNodeLabel() & vbCrLf & vbCrLf			
42 1Str = 1Str & "1DET Filter = " & Mesh.Get1DETFilter() & vbCrLf			
43 1Str = 1Str & "Number of 1DET = " & Mesh.GetNum1DET() & " Highest 1DET number = " & Mesh.GetHigh1DETLabel() & vbCrLf & vbCrLf			
44 1Str = 1Str & "IEI4 Filter = " & Mesh.GetTET4Filter() & vbCrLf			
<pre>45 lStr = lStr &amp; "Number of TET4 elements = " &amp; Mesh.GetNumTET4() &amp; " Highest TET4 element number = " &amp; Mesh.GetHighTET4Label() &amp; vbCrLf &amp; vbCrLf</pre>			
1 str = 1str & "Ikis ritter = " & Mesh.GetIRISFilter() & VbCrLf			
1 1961 - 1961 & Houmber of FRIS elements - " & REST.GetNUMIKIS() & " highest ikis element humber - " & REST.GetAl@hIKISLADEI() 40 Member 1857 - MScripters			
49	*		
	•		
Ready	Ln 1, Col 1 NUM		

## vbScript Programming Resources

#### Online

- Include the word vbscript in any searches
- Searches often returns results for Visual Basic for Applications(VBA)
  - VBA is compiled, vbscript is interpreted
- There are a number of user guides for vbscript
  - Search for vbscript Language Reference or vbscript User Guide

- Books
  - VBScript in a Nutshell



#### **StudyMod**

#### StudyMod - Modify an existing Study file

- Allows changes to
  - Boundary Conditions
  - Mesh
  - Processing Settings
  - Materials
- Not designed to create/add/modify geometry
- Modifier file is in XML format

NAME:

Studymod - Modify a study File

SYNOPSIS: studymod <InputStudy> <OutputStudy> <ModifierFile>

#### StudyMod Example Modifier File

Set the Melt Temperature to 240C

```
<?xml version="1.0" encoding="utf-8"?>
<StudyMod title="Autodesk StudyMod" ver="1.00">
  <UnitSystem>Metric</UnitSystem>
  <Property>
       <TSet>
           <!--Process controller-->
           <ID>30011</ID>
           <SubID>1</SubID>
           <!--Melt temperature-->
           <TCode>
                <ID>11002</ID>
                <Description>Melt
temperature</Description>
                <Value>240</Value>
           </TCode>
     </TSet>
  </Property>
</StudyMod>
```

#### **Study**Rlt

#### StudyRlt – Extract Results Data

- Data which can be extracted
  - Individual message output (-message)
  - Sequence screen output (-exportoutput)
  - Result in XML format (-xml)
  - Model in Patran format (-exportpatran)
  - Value from Result Data (-result)
    - By Region/Layer
    - Calculation: Min/Max/Average

#### StudyRlt Example From Command Line

NAME:

Studyrlt - Result Extraction Utility

SYNOPSIS:

<study> -xml <result ID>

<study> -exportpatran

<study> -result <result ID>

-min|-max|-average|-stddev|-count|-node <node number>|-element <element number>

[-layer <layer name>|-cavity|-gate|-runner|-sprue]

[-component <number> [-anchor <node1> <node2> <node3>]

[-unit SI|Metric|English]

# 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 and services 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.