Moldflow Summit 2018: Automation with API

Matt Jaworski
Sr. Subject Matter Expert, Moldflow

Ana Maria Marin
TZERO® Consultant/Trainer - RJG, Inc.
RJG Global Offices

Servicing the Plastics Industry since 1989

RJG USA
Traverse City, MI
(RJG, Inc. Headquarters)

Woodstock, GA
(Regional Training Center)

Gibsonville, NC
(Regional Training Center)

RJG Mexico
Chihuahua, Mexico

RJG France
Arinthod, France

RJG Germany
Karlstein, Germany

RJG Ireland / UK
Co Tipperary, Ireland

RJG Italy
Next Innovation Srl
Milano, Italy

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

RJG China
Chengdu, China

RJG Korea
CAEPro
Seoul, Korea

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

Image courtesy of RJG, Inc.
Half the Battle is Knowing What is Happening in the Mold

Process Control Systems and Cavity Pressure Sensing Technology

Image courtesy of RJG, Inc.
What is TZERO®?

Validation Between Engineering and Manufacturing: A Systematic Process

GLOBAL

End-to End Services

- Consulting
- Simulation
- Training

Single Source Solutions Provider for Plastic Injection Molding

Image courtesy of RJG, Inc.
What is the Autodesk Moldflow Insight API?

- API = Application Programming Interface
- 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 through Autodesk Moldflow Insight (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

    ```vbnet
    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
    ```

  Must be first line to avoid this message

- 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
Class Member

Here is a list of all documented class members with links to the class documentation for each member:

- AbortJob(): JobMgr
- AbortJobByID(): JobMgr
- AbortMyJobsOnServer(): JobMgr
- AbortMyJobsOnServerByAddress(): JobMgr
- AbortMyJobsOnServerByOwner(): JobMgr
- ActivateLayer(): LayerManager
- ActivateLCS(): Modeler
- Active(): LayerManager
- AddBookmark(): ViewMgr
- AddDefaultPlots(): PlotMgr
- AddDouble(): DoubleArray
- AddFile(): StudyDoc
- AddInt(): IntElement
- AddObjectToFolder(): FolderManager
- AddProbePlane(): Plot
- AddProbePoint(): ProbePoint
- Address(): Server
- AddScalarData(): UserData
- AddString(): SizingArray
- AddTensorData(): UserData
- AddXYPlotCurve(): Plot
- AddXYPlotData(): UserPlot
- AddXYZ(): VectorArray
**TCode Reference**

- **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.
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
Notepad ++

- Great tool for writing, debugging and editing scripts
- Free version available
- Does have nagging timer to get you to buy
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>
Set the Melt Temperature to 240°C

```xml
<?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>
```
StudyRlt

- 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
NAME:
StudyRlt - Result Extraction Utility

SYNOPSIS:
studyrlt <study> -message <sequence> <message ID> <occurrence> <item> [-unit SI|Metric|English]
<study> -exportoutput [<sequence>] [-output <filename>] [-unit SI|Metric|English]
<study> -exportwarp <result ID> -actual|opposite -scale <x> -output <filename> [-unit SI|Metric|English]
<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]