# Dynamo for Dummies: An Intro to Dynamo and How It Interacts with Revit

Marcello Sgambelluri

BIM Director - John A. Martin Structural Engineers LA,CA

Twitter: @marcellosgamb

# Class summary

In this lab attendees will get very gradual doses of the Dynamo extension and its programming environment so they gain the skills to make the Dynamo extension work in their everyday workflows with Revit software. No programming experience or complex modeling experience needed to attend this lab. You only need a little Revit experience.

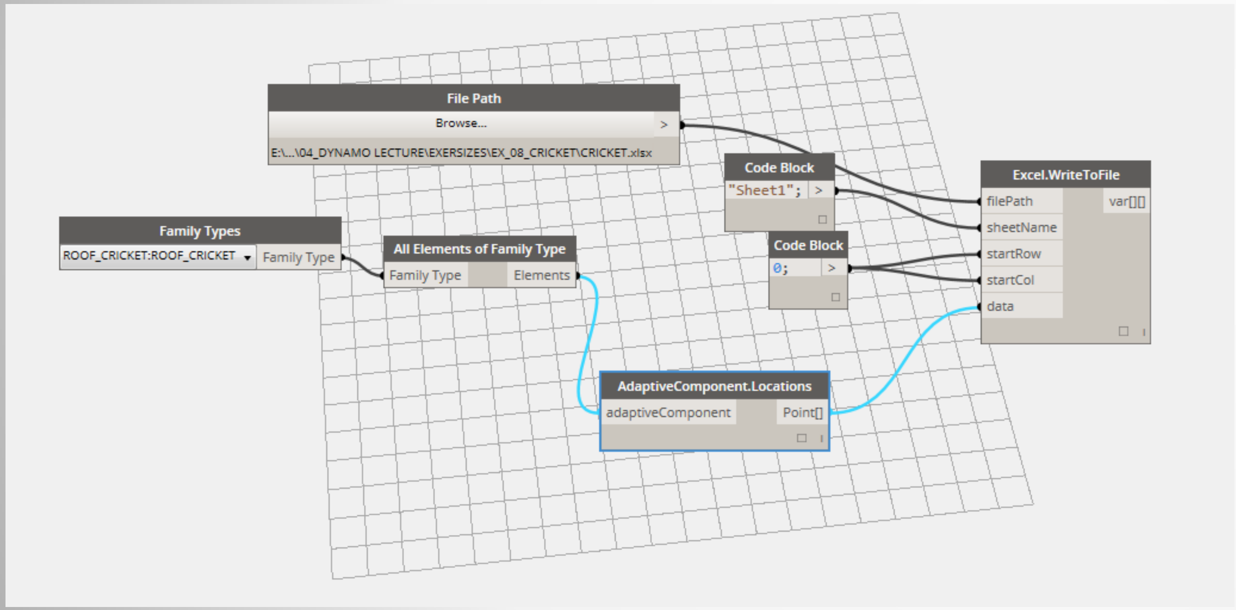
# Key learning objectives

At the end of this class, you will be able to:

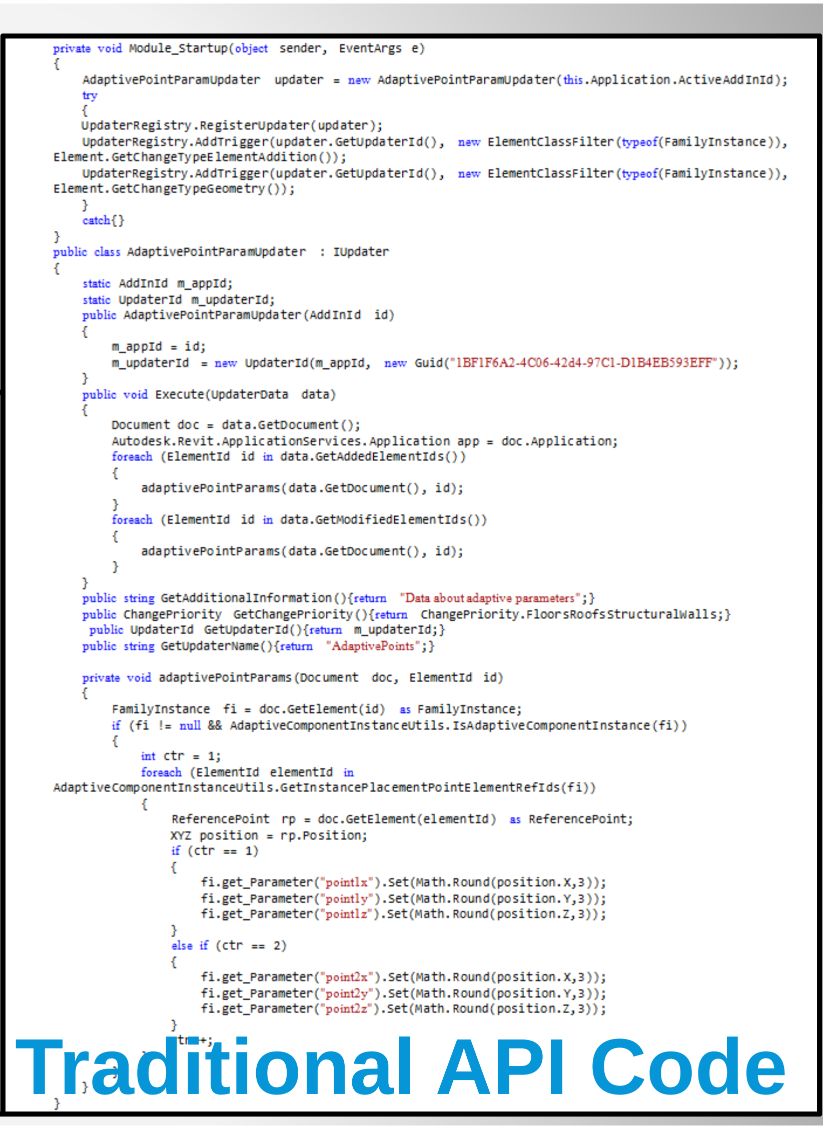
- Learn the answer to the question: What is the Dynamo extension?
- Learn how to program using visual programming
- Learn practical uses of dynamo
- Learn how to use the Dynamo extension in your everyday Revit software workflows

# Introduction

# DYNAMO IS A VISUAL PROGRAMING LANGUAGE



## Visual Programming Code



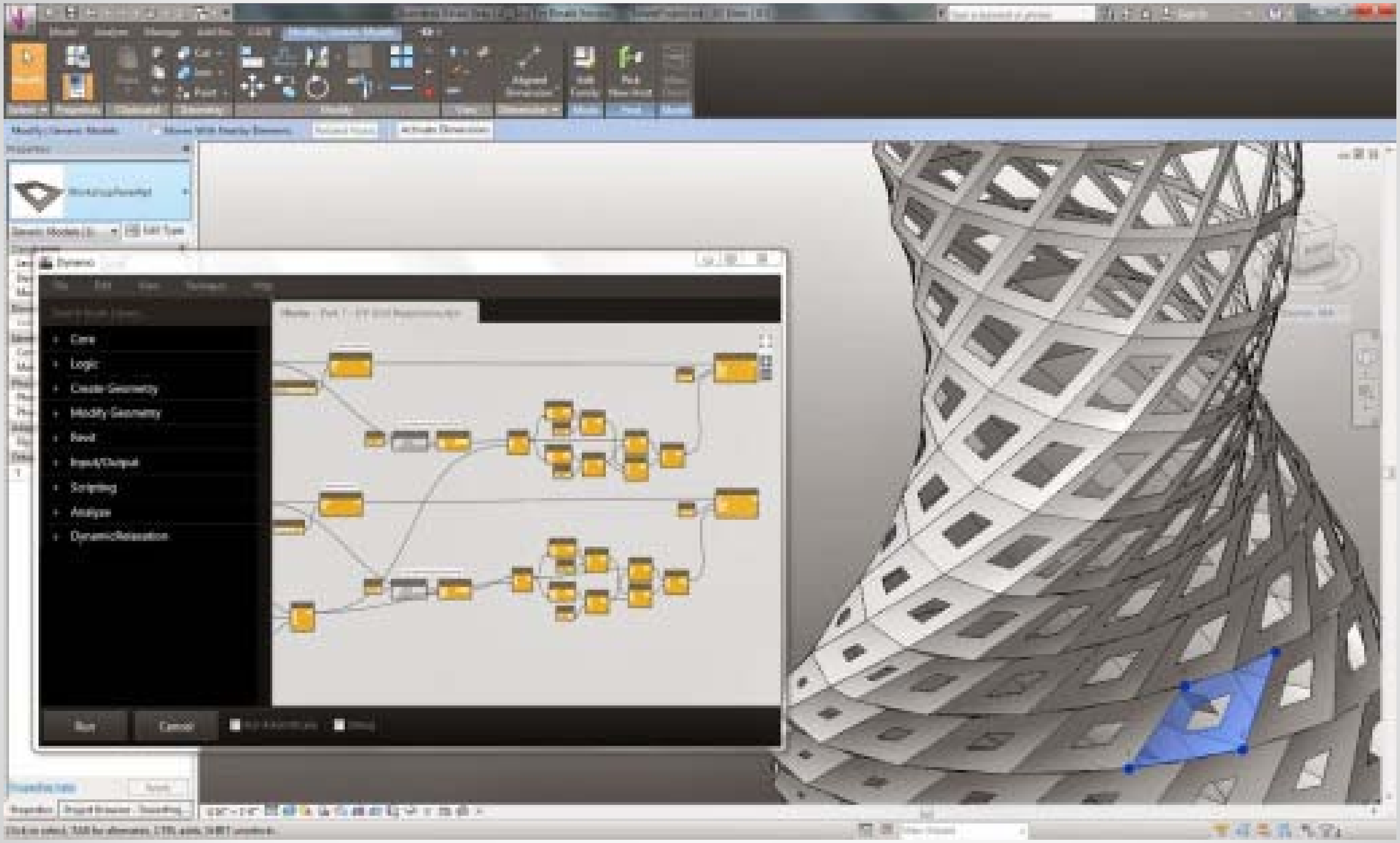
```
private void Module_Startup(object  sender,  EventArgs  e)
{
    AdaptivePointParamUpdater  updater  = new AdaptivePointParamUpdater(this.Application.ActiveAddInId);
    try
    {
    UpdaterRegistry.RegisterUpdater(updater);
    UpdaterRegistry.AddTrigger(updater.GetUpdaterId(),  new ElementClassFilter(typeof(FamilyInstance)),
Element.GetChangeTypeElementAddition());
    UpdaterRegistry.AddTrigger(updater.GetUpdaterId(),  new ElementClassFilter(typeof(FamilyInstance)),
Element.GetChangeTypeGeometry());
    }
    catch{}
}
public class AdaptivePointParamUpdater  : IUpdater
{
    static AddInId m_appId;
    static UpdaterId m_updaterId;
    public AdaptivePointParamUpdater(AddInId  id)
    {
        m_appId = id;
        m_updaterId  = new UpdaterId(m_appId,  new Guid("1BF1F6A2-4C06-42d4-97C1-D1B4EB593EFF"));
    }
    public void Execute(UpdaterData  data)
    {
        Document doc = data.GetDocument();
        Autodesk.Revit.ApplicationServices.Application app = doc.Application;
        foreach (ElementId  id in data.GetAddedElementIds())
        {
            adaptivePointParams(data.GetDocument(), id);
        }
        foreach (ElementId  id in data.GetModifiedElementIds())
        {
            adaptivePointParams(data.GetDocument(), id);
        }
    }
    public string GetAdditionalInformation(){return  "Data about adaptive parameters";}
    public ChangePriority  GetChangePriority(){return  ChangePriority.FloorsRoofsStructuralWalls;}
     public UpdaterId  GetUpdaterId(){return  m_updaterId;}
    public string GetUpdaterName(){return  "AdaptivePoints";}

    private void adaptivePointParams(Document  doc, ElementId  id)
    {
        FamilyInstance  fi = doc.GetElement(id)  as FamilyInstance;
        if (fi != null && AdaptiveComponentInstanceUtils.IsAdaptiveComponentInstance(fi))
        {
            int ctr = 1;
            foreach (ElementId  elementId  in
AdaptiveComponentInstanceUtils.GetInstancePlacementPointElementRefIds(fi))
            {
                ReferencePoint  rp = doc.GetElement(elementId)  as ReferencePoint;
                XYZ position = rp.Position;
                if (ctr == 1)
                {
                    fi.get_Parameter("point1x").Set(Math.Round(position.X,3));
                    fi.get_Parameter("point1y").Set(Math.Round(position.Y,3));
                    fi.get_Parameter("point1z").Set(Math.Round(position.Z,3));
                }
                else if (ctr == 2)
                {
                    fi.get_Parameter("point2x").Set(Math.Round(position.X,3));
                    fi.get_Parameter("point2y").Set(Math.Round(position.Y,3));
                    fi.get_Parameter("point2z").Set(Math.Round(position.Z,3));
                }
                ctr++;
            }
        }
    }
}
```

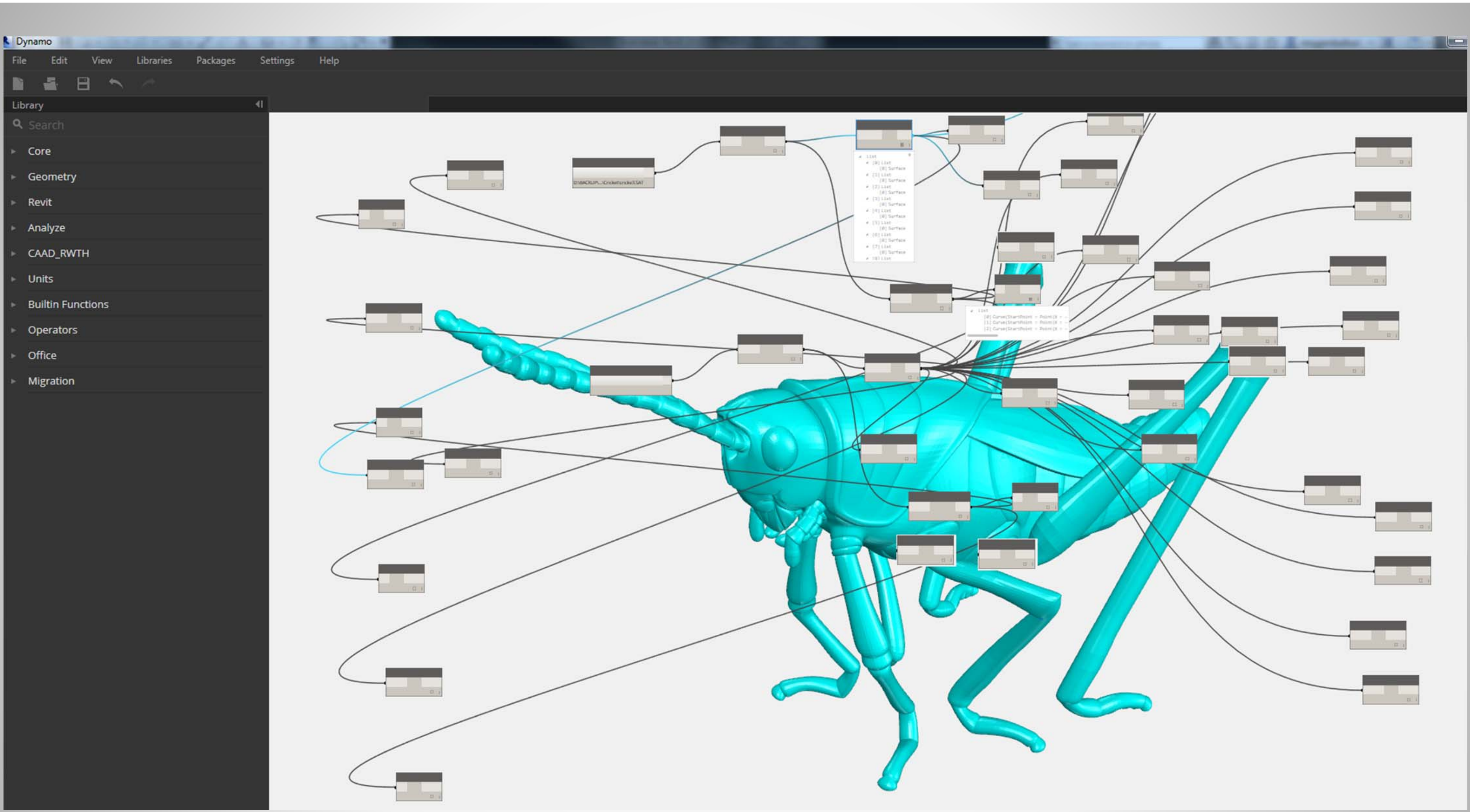
## Traditional API Code

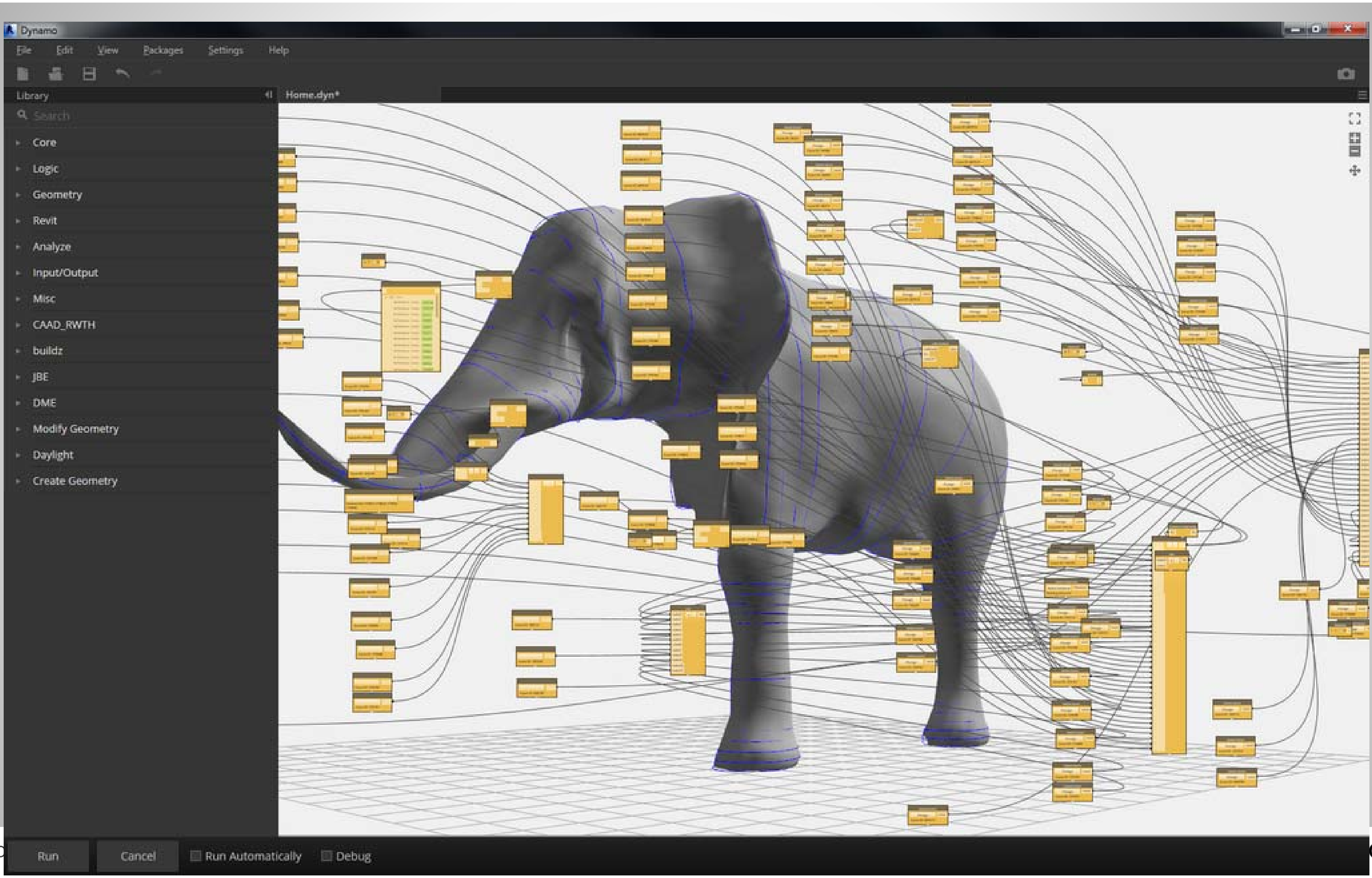
AUTODESK UNIVERSITY 2014

AUTODESK.



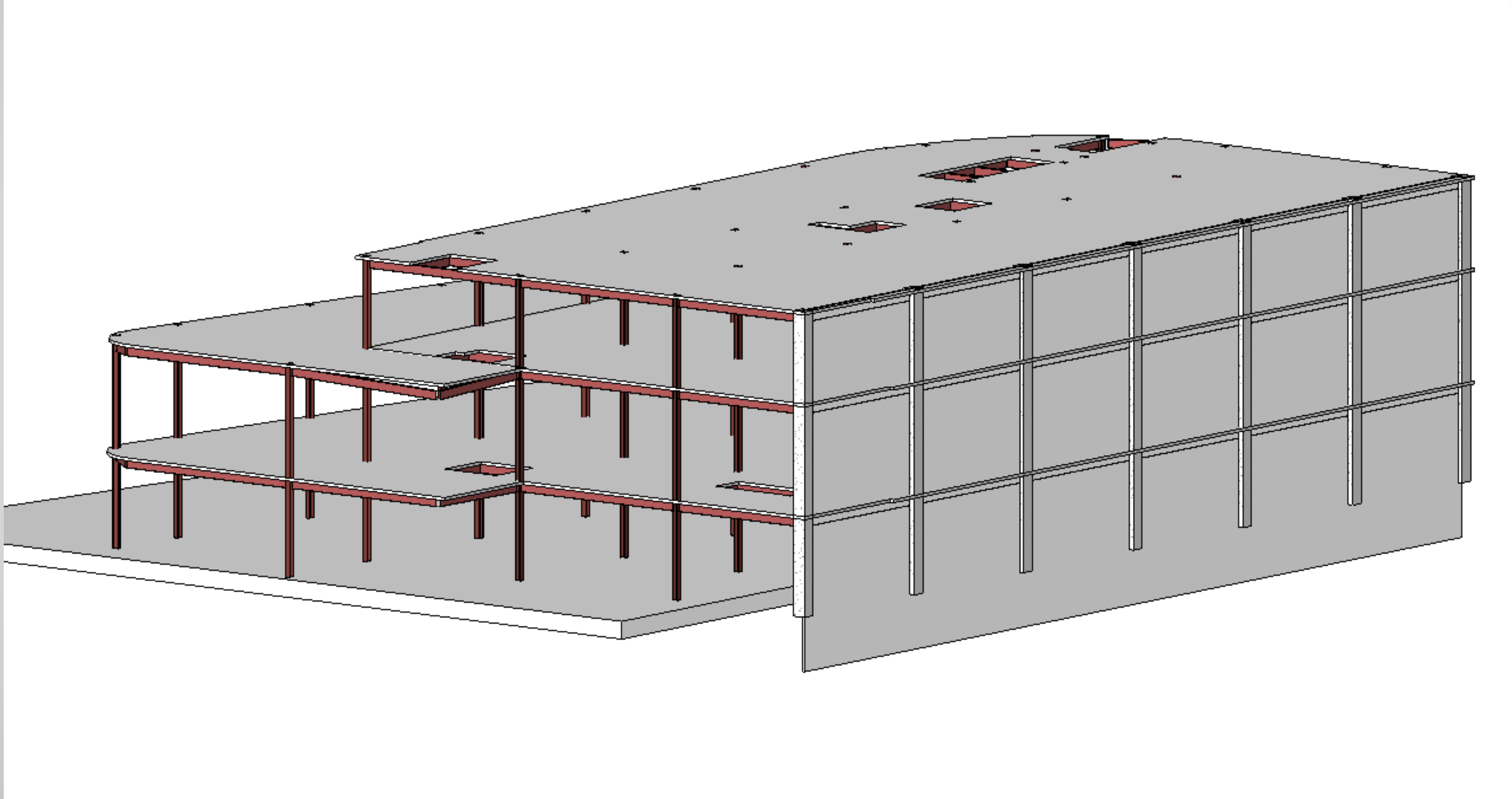
Source
http://85flukus.wordpress.com/2013/12/11/dynamo-for-revit/


Dynamo
File
Edit
View
Libraries
Packages
Settings
Help
Library
Search
Core
Geometry
Revit
Analyze
CAAD_RWTH
Units
Builtin Functions
Operators
Office
Migration

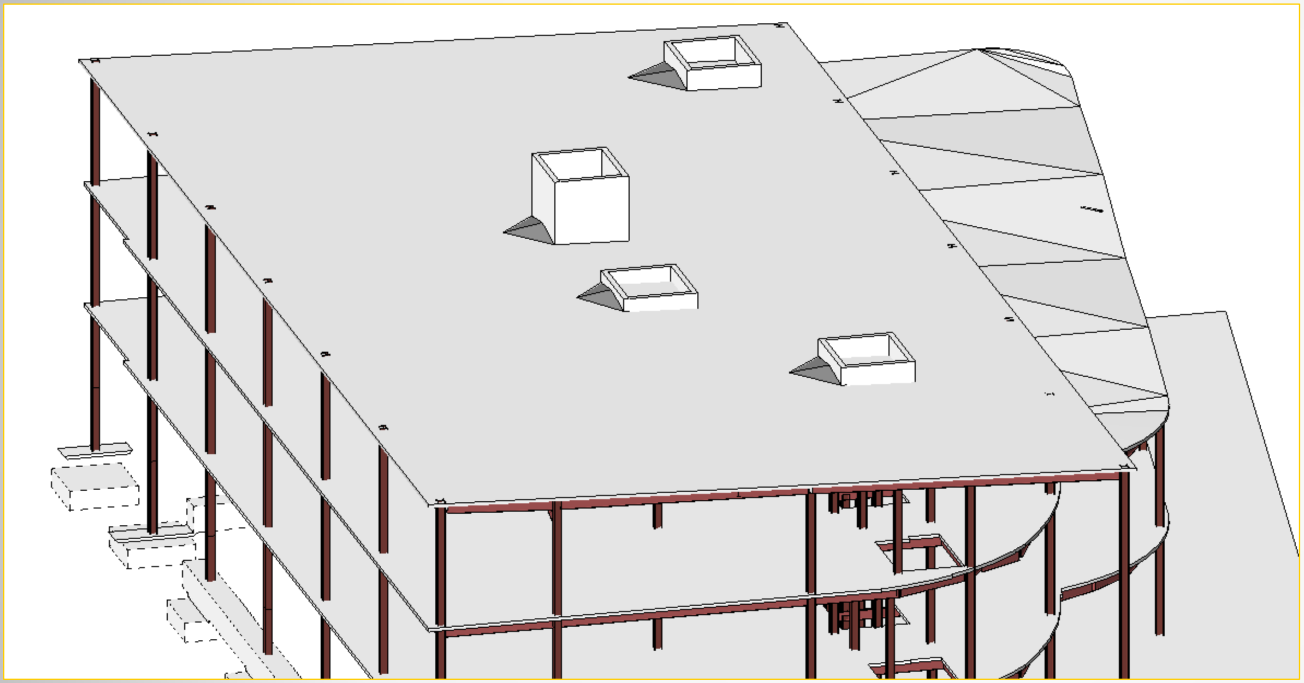

Dynamo
File
Edit
View
Packages
Settings
Help
Library
Search
Home.dyn*
Core
Logic
Geometry
Revit
Analyze
Input/Output
Misc
CAAD_RWTH
buildz
JBE
DME
Modify Geometry
Daylight
Create Geometry
Run
Cancel
□ Run Automatically
□ Debug

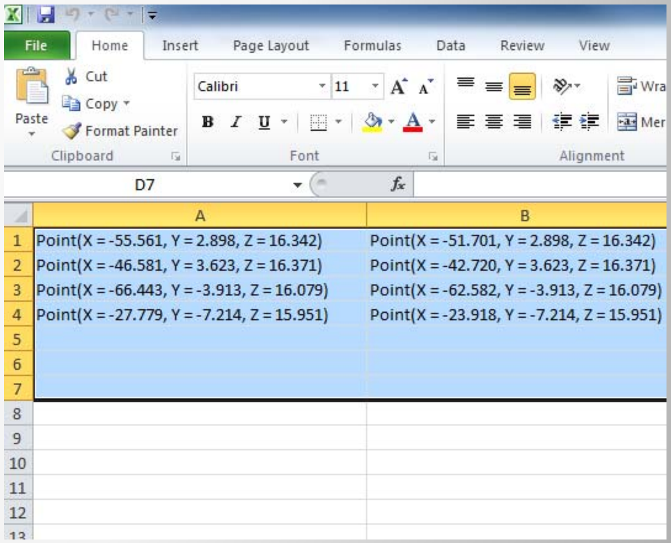
# Exercises

# Getting and Setting Parameters (Col/Wall Base)



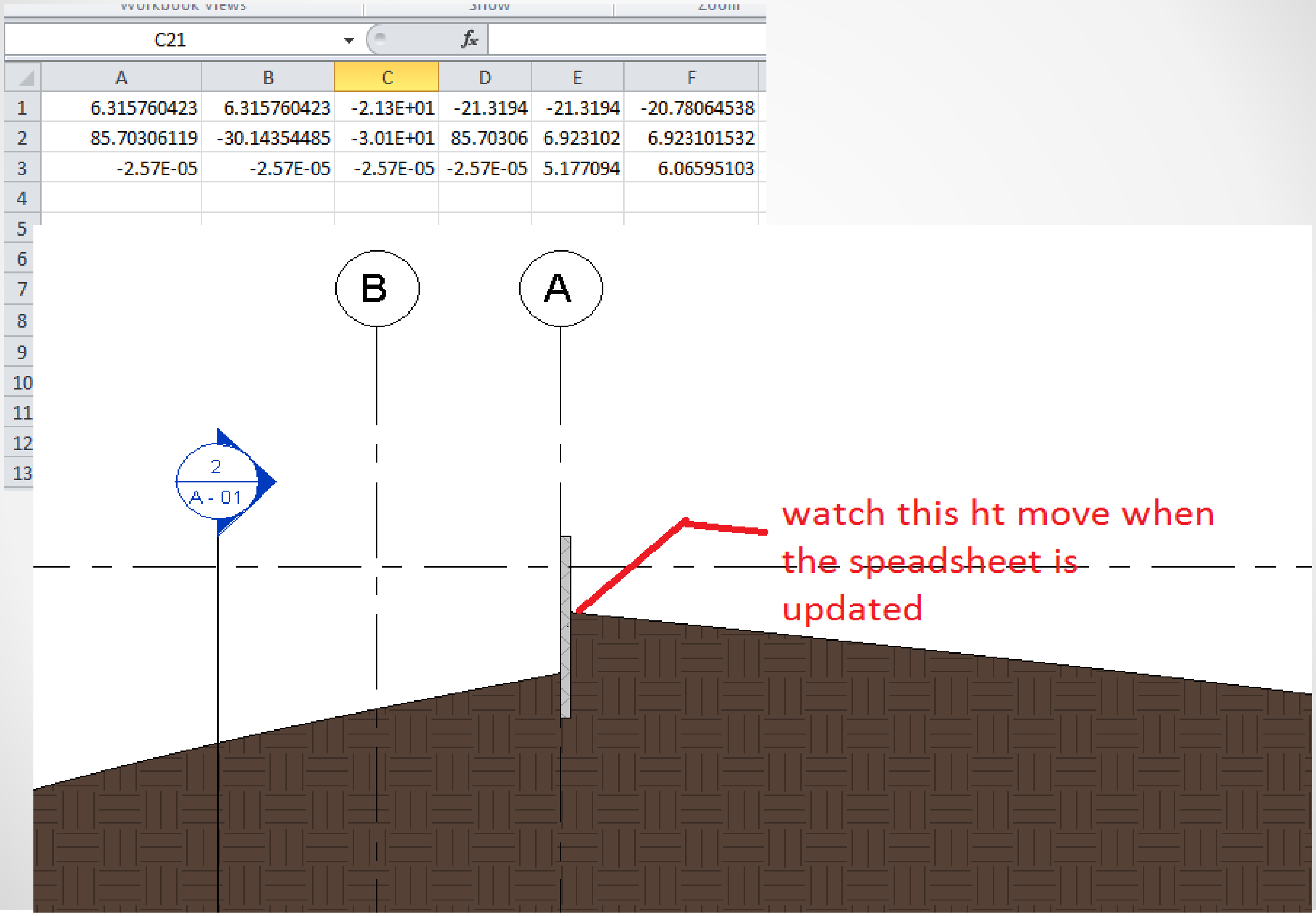
# Extracting AC Points and Writing to Excel





| | A | B |
|---|---|---|
| 1 | Point(X = -55.561, Y = 2.898, Z = 16.342) | Point(X = -51.701, Y = 2.898, Z = 16.342) |
| 2 | Point(X = -46.581, Y = 3.623, Z = 16.371) | Point(X = -42.720, Y = 3.623, Z = 16.371) |
| 3 | Point(X = -66.443, Y = -3.913, Z = 16.079) | Point(X = -62.582, Y = -3.913, Z = 16.079) |
| 4 | Point(X = -27.779, Y = -7.214, Z = 15.951) | Point(X = -23.918, Y = -7.214, Z = 15.951) |
| 5 | | |
| 6 | | |
| 7 | | |
| 8 | | |
| 9 | | |
| 10 | | |
| 11 | | |
| 12 | | |

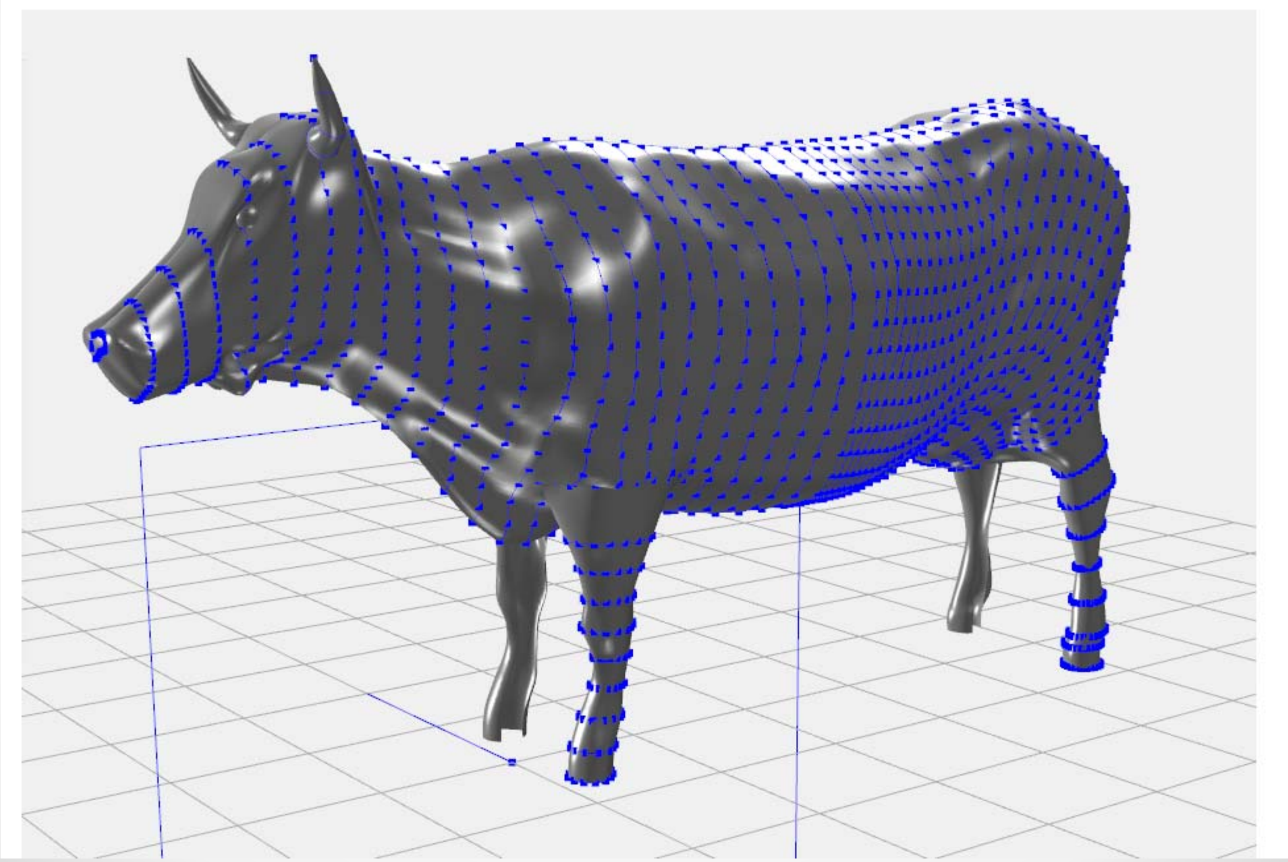
# Create Topo From Excel Points and Modify



|  | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | 6.315760423 | 6.315760423 | -2.13E+01 | -21.3194 | -21.3194 | -20.78064538 |
| 2 | 85.70306119 | -30.14354485 | -3.01E+01 | 85.70306 | 6.923102 | 6.923101532 |
| 3 | -2.57E-05 | -2.57E-05 | -2.57E-05 | -2.57E-05 | 5.177094 | 6.06595103 |

watch this ht move when the speadsheet is updated

# Design Scripting and the Revit Cow



# Lets Get Started!!!!!!!!!

# Session Feedback

- Via the Survey Stations, email or mobile device
- AU 2014 passes given out each day!
- Best to do it right after the session
- Instructors see results in real-time

AUTODESK®