

DevTV

Introduction to Revit programming

Meet the presenter

Augusto Gonçalves
Developer Technical Services
Brazil
Autodesk



Augusto is member of DevTech team since 2008 based at São Paulo office. He is specialist in AutoCAD, Civil3D and Revit APIs.

Before join Autodesk, Augusto worked on CAD-related developments for civil engineering and web commerce. He is graduated in Civil Engineering and holds a Master in Computer Engineer..

Getting started

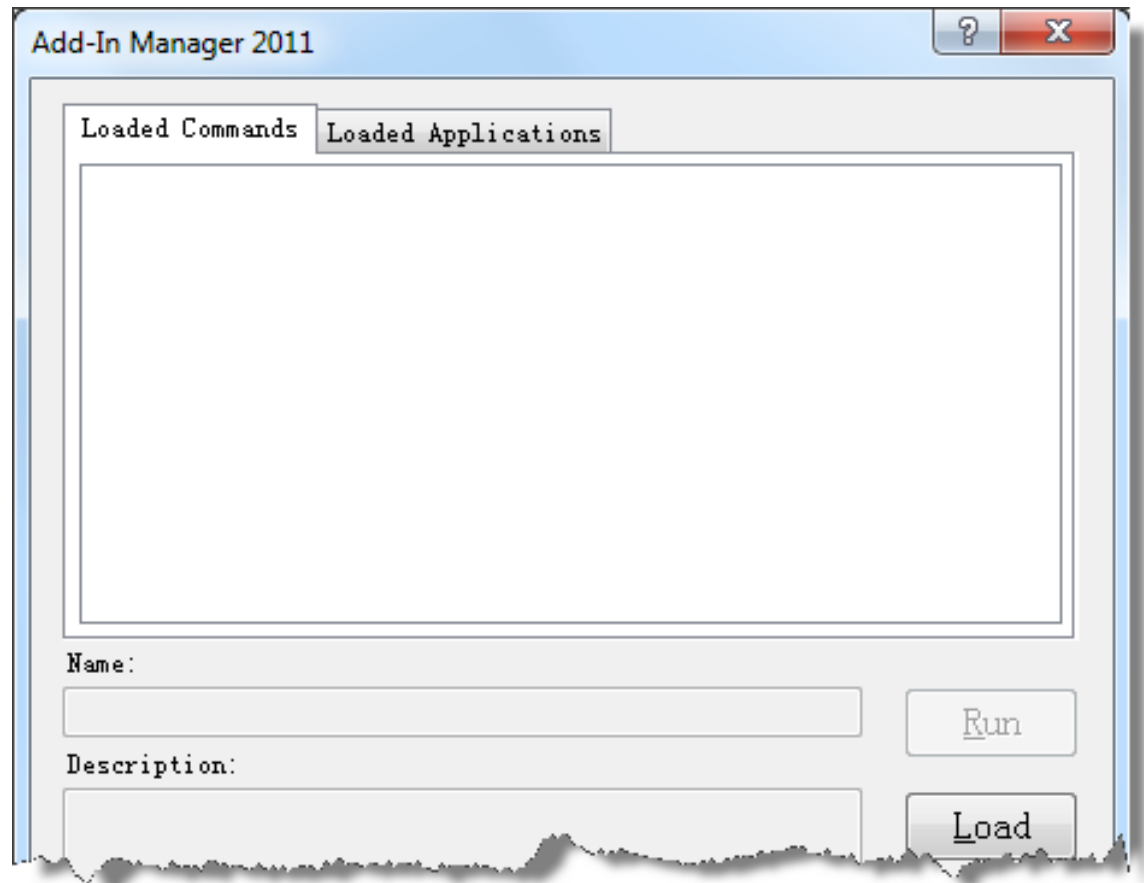
- **The product:** Revit Architecture, Structure and MEP
- **The API:** common set of features for all three versions, plus specific classes for each version.
- **The SDK:** contains a lot of information about the API
 - Developers Guide ([*Revit 2011 API Developer Guide.pdf*](#))
 - API Reference ([*RevitAPI.chm*](#))
 - Recommended Samples: RevitCommands, Fire Rating, Viewers/ElementViewer, CreateBeamsColumnsBraces

Quick start: “Hello World” sample

- Configure the environment
- Create the .addin registration file
- Setup the Transaction and Regeneration modes

Add-In Manager for Autodesk Revit

- Available at <SDK Folder>\Add-In Manager
- Benefits:
 - Run external commands without need to create an addin file



External command parameters

- **commandData**: access to information needed by the command, such as Application, Document, Selection, etc.
- **message**: this will be displayed only if the command return *Failed*.
- **elements**: the set of elements displayed with the failure message

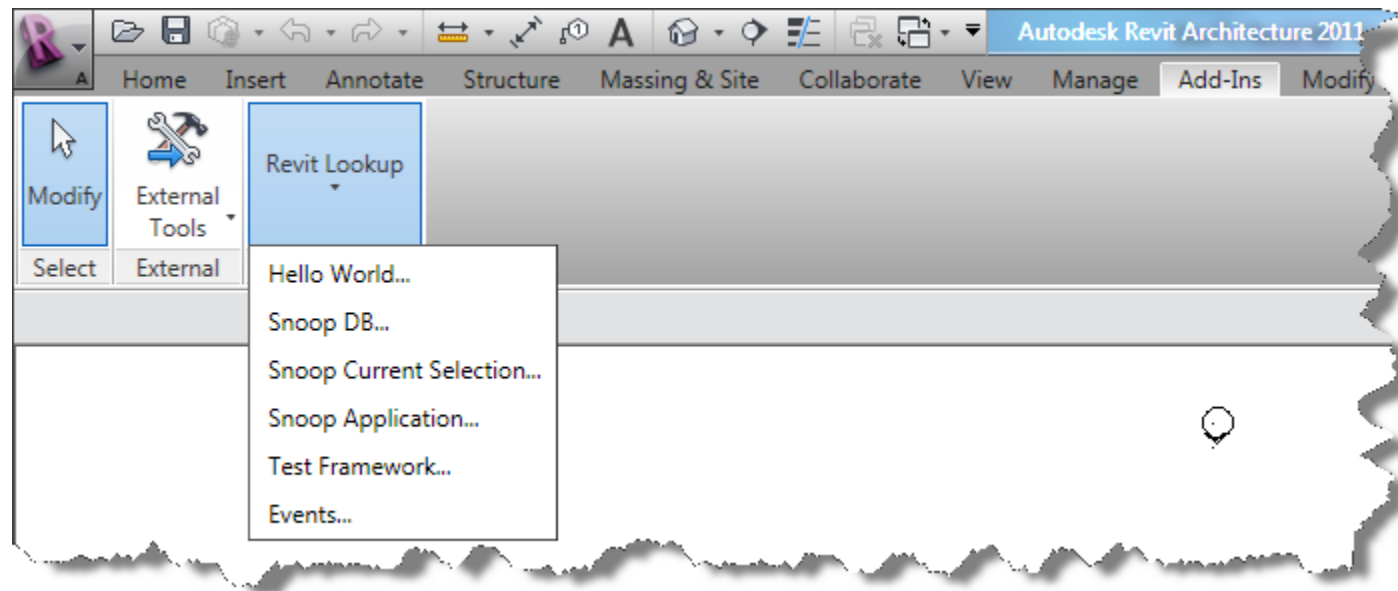
```
public class Commands : IExternalCommand
{
    public Result Execute(
        ExternalCommandData commandData,
        ref string message,
        Autodesk.Revit.DB.ElementSet elements)
    {
```

Current selection

- Let's make a demonstration on how get the current selection
- Show each element's name and category.

Inspector tool: Revit Lookup (formerly RvtMgdDbg)

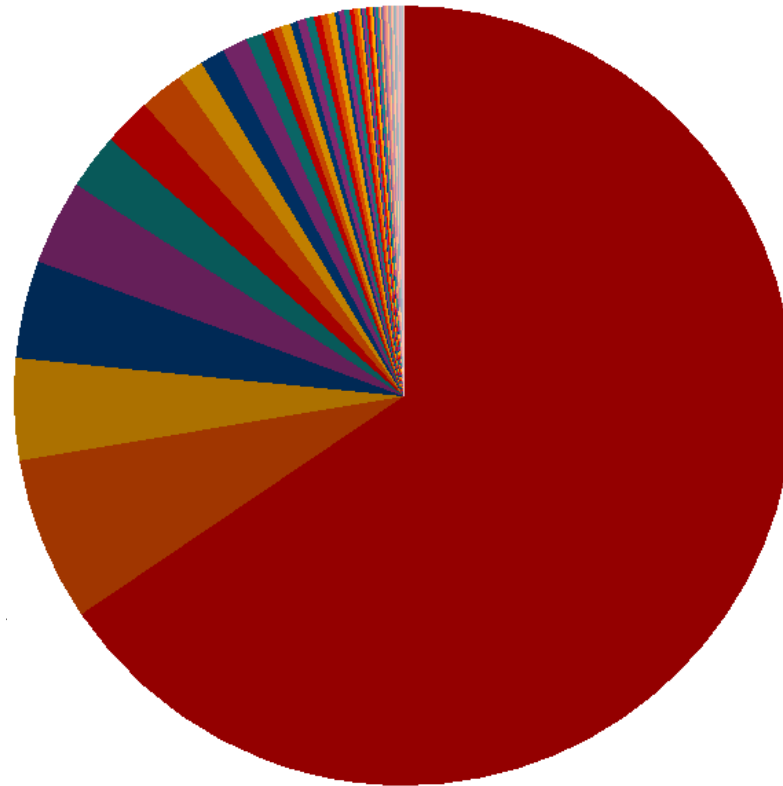
- Snoop the database, current selection and application
- Run tests
 - Contains many interesting samples
- Demonstration >>



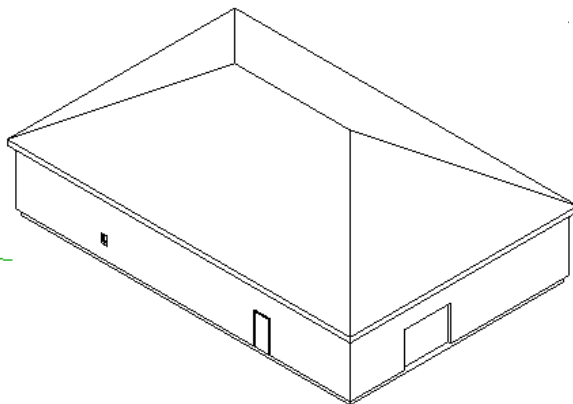
Composition of Revit database

- From a simple test drawing of 20~30 elements

| | | |
|---------------|------|--------|
| Element | 1586 | 65.48% |
| ElementType | 165 | 6.81% |
| MaterialOther | 102 | 4.21% |
| FamilySymbol | 98 | 4.05% |
| Dimension | 85 | 3.51% |
| Family | 56 | 2.31% |
| FillPattern | 48 | 1.98% |
| ModelLine | 44 | 1.82% |
| WallType | 26 | 1.07% |
| DetailLine | 23 | 0.93% |



- Element
- ElementType
- MaterialOther
- FamilySymbol
- Dimension
- Family
- FillPattern
- ModelLine
- WallType
- DetailLine
- SketchPlane
- LinePattern
- FamilyInstance
- MullionType
- Sketch
- DimensionType
- SpotDimensionType
- View



Iterate through database

- Using **FilteredElementCollector** to get elements
- Can filter by criteria such as:
 - OfCategory
 - OfCategoryId
 - OfClass
- Require an **Document** object

Parameters

- Built-in parameters can be accessed and edit for each element
- Each parameter has its own properties, such as name, storage type, value
- Shared parameters are external text files containing parameter specifications and are useful to store additional information on elements
- For more information about shared parameters, check the Chapter 19 of Developers Guide and Fire Rating SDK sample

Learning more – next steps

- The Building Code
 - <http://thebuildingcoder.typepad.com>
- Developer Center
 - www.autodesk.com/developervit
- Discussion groups
 - <http://discussion.autodesk.com/forums/forum.jspa?forumID=160>
- API Training
 - www.autodesk.com/apitraining
- Autodesk Developer Network
 - www.autodesk.com/joinadn

API Revit .NET

