Revit Family API DevTV
编程创建族库

叶雄进 Joe Ye
Developer Consultant
Autodesk Developer Network

Autodesk

关于讲师

- 叶雄进 Joe Ye　　Joe.Ye@autodesk.com
  - Developer Consultant
  - Autodesk Developer Network

- 3年软件开发咨询
- 5年工程软件研发
- 支持APIs
  - Revit Architecture/Structure/MEP
  - AutoCAD Architecture
  - AutoCAD

Revit Family API 简介

- 背景
  - 用命令方式创建族本身就是一个高度定制功能
  - 理解用Revit 命令的方式创建族对学习编程创建族很重要
  - 两种阵容的族创建者
    - 对界面命令非常了解，但是不熟悉Revit二次开发
    - 熟悉编程，但是对界面命令不熟悉

- 本讲座目的
  - 按照界面操作的步骤学习编程创建族的过程
  - 学习主要Family API

内容提要

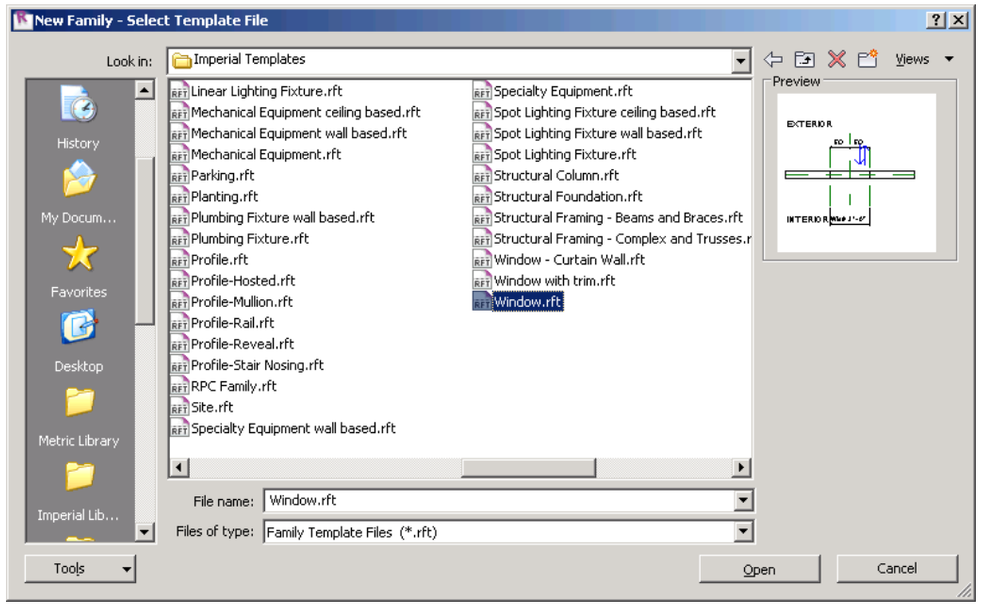
- 用户界面操作的角度来看族
  - 族是什么?
  - 手工创建一个简单的族
  - 从哪里开始、族分类、编辑器、族能实现的功能
  - 最佳实践
- 用Family API来创建族
  - 按最佳实践的方法学习编程步骤
  - 代码示例：创建L形的柱子
  - 学习资料介绍

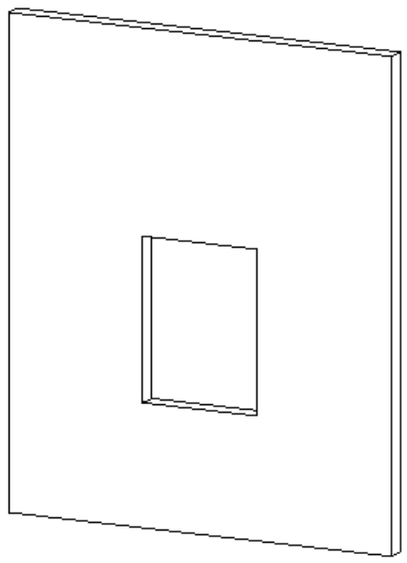
什么是Revit的族？

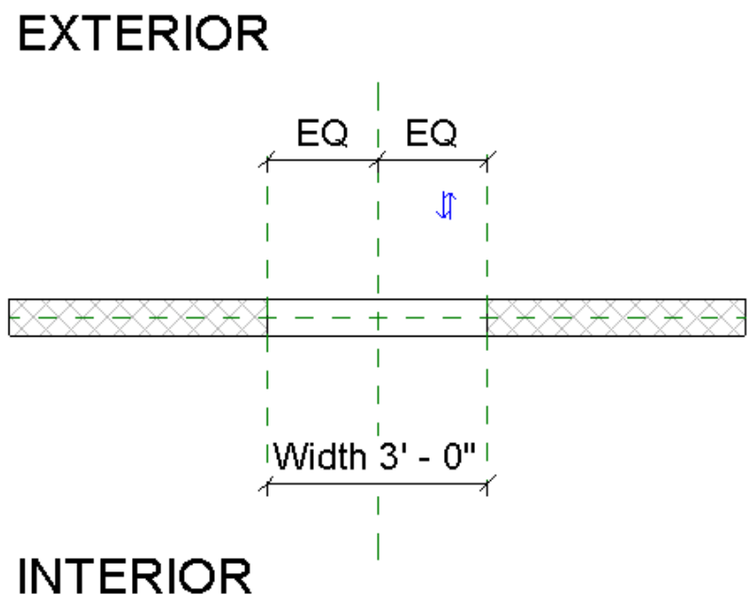
- 具有图形表示的建筑对象或符号
  - 有3D 或 2D 图形
  - 有建筑对象的数据
- 一般有三种类型
  - **系统族（System Family）** – 保存在工程模板文件中
    - 如墙，屋顶，楼板，天花板…
  - **标准族 （Standard Family）**– 独立 “.rfa” 文件
    - 窗户，门，家具，梁，照明灯具…
    - Revit 2010 中有API 来创建
  - **在位族（In-Place Families）** – “one of kind objects”

Revit Families – 从何处开始?

- 从哪里开始?
  - 从族模板开始
  - 修改一个已有族
- 使用哪个模板?
  - 2D 或 3D, 模型或详图?
  - 有附着主体的还是没有附着体， 如: Wall, Ceiling, Face based…
  - 族类别 （Category）
  - 布置类型：单点或两点
  - 特殊: 桁架, 钢筋…







Revit Family 编辑器

- 提供了6种基本的族编辑器
  - 3D 模型, 标准, 详图符号, 钢筋, 桁架 和新的概念体量.

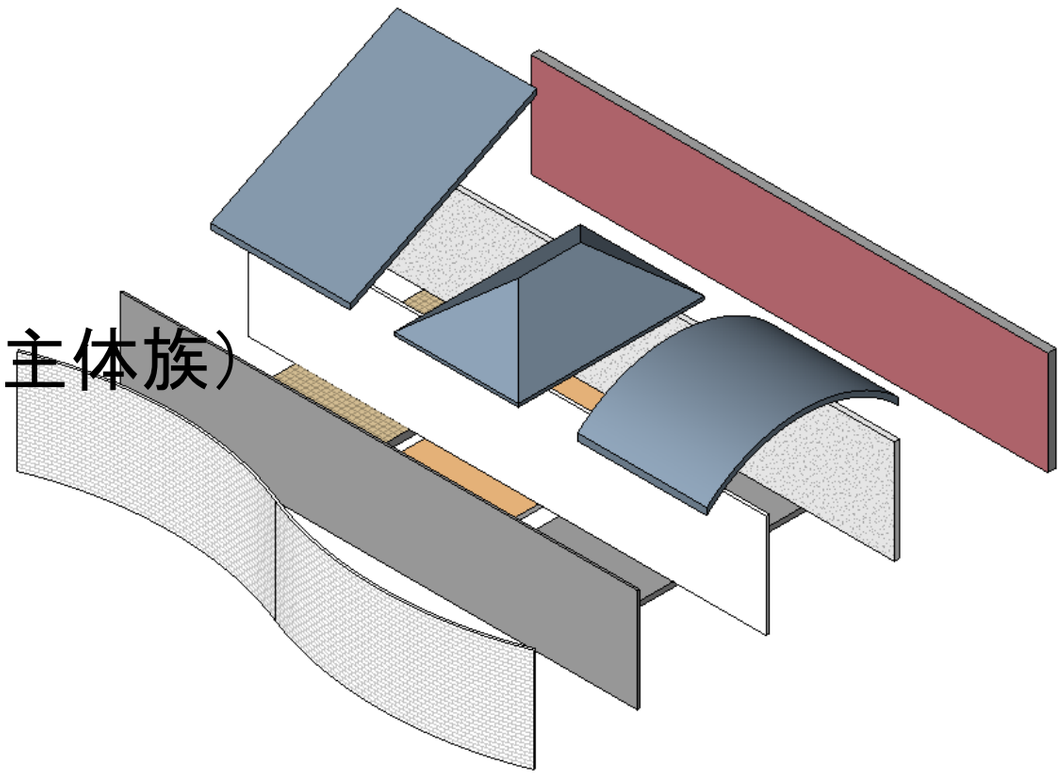
- 每一个族编辑器提供了一组匹配的命令，编辑器的样式取决于所选用的族模板
  - 几何 – extrusions, blends, sweeps, revolves, swept blends
  - 画线 – 模型线，符号线，详图线
  - 基础编辑工具 – 拷贝, 镜像, 喷绘, 拼接/断开拼接, 剪切几何/不剪切
  - 参照工具 – 参考面, 参考线
  - 标注工具 – 标注与族参数绑定
  - 高级工具 – 公式，嵌套，阵列，类型目录
  - MEP 工具 – 添加 connectors

创建Revit族的最佳实践

了解创建族的步骤是学习创建族最重要的一项内容

创建步骤: (有些步骤次序可调换）

1. 规划 (插入点，参数原点）
2. 布局参照平面 (搭架子)
3. 添加参数
4. 添加多个有厚度的主体类型(用于测试主体族)
5. 添加两个以上的类型
6. 灵活的类型和主体，用于测试
7. 添加单一级别的几何体
8. 重复6 & 7 直到满意结果
9. 在工程环境中测试

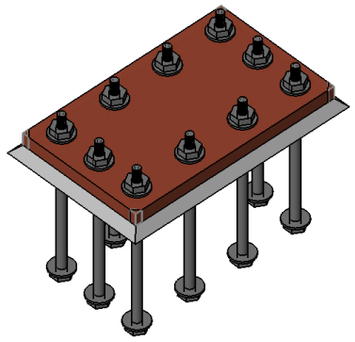


Steven Campbell, Revit Content Project Manager

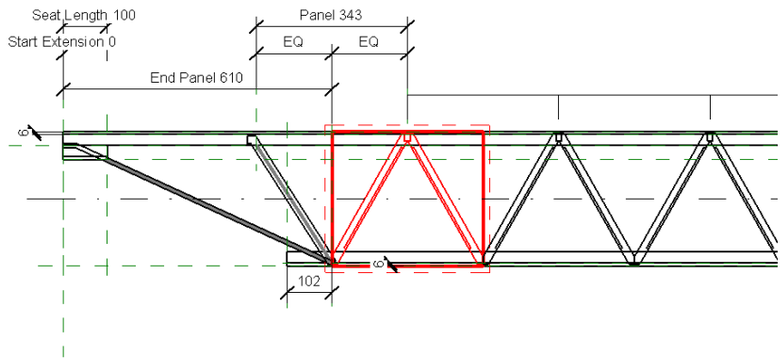
新建族

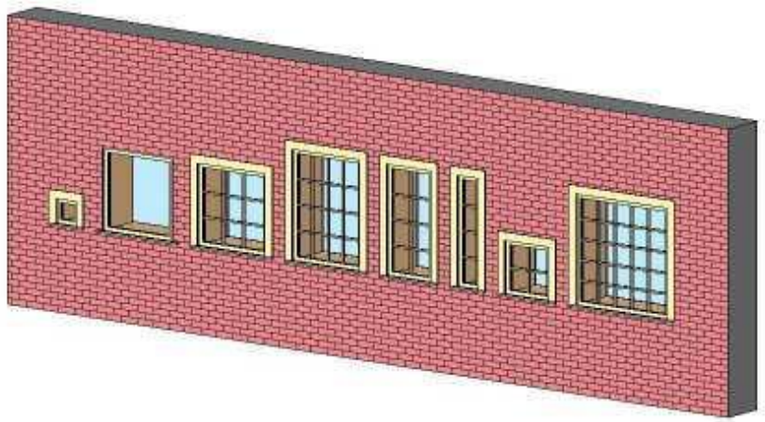
- 演示用命令创建L形柱族

Revit Families – 能做什么？

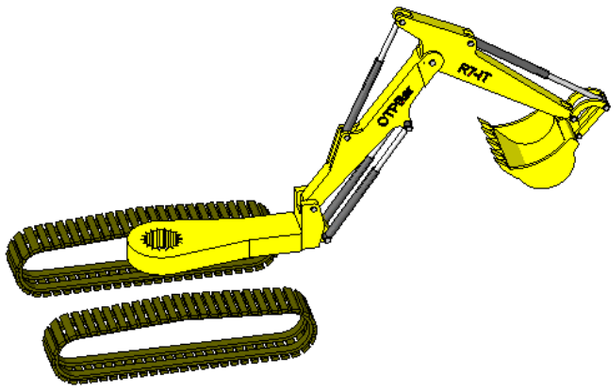


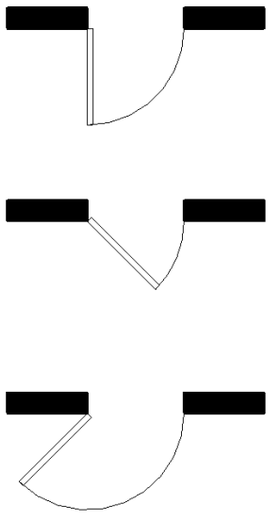
- 公式 – 用于控制行为，可见性和阵列对象
- 阵列和嵌套





- 高级嵌套 – 子对象可以交换
- 参照线 –可动角度





Family 资料

- Revit Families Guide
  - 2010& 2011 release

- 论坛
  - discussion.autodesk.com
  - AUGI - http://forums.augi.com
- 书籍 & 光盘
  - Mastering Autodesk Revit Building – Paul F. Aubin
  - Mastering Family Editor Series - 5 DVD’s – Paul F. Aubin

Family API

沿着最佳实践学习基础

Family API 概览

族是什么?

- Revit API 运行在族编辑器上下文环境
- 能自动创建族库，后台实时生成族
- 提取信息，并修改
- 模拟界面功能
- 需要意识到它有一些区别和局限

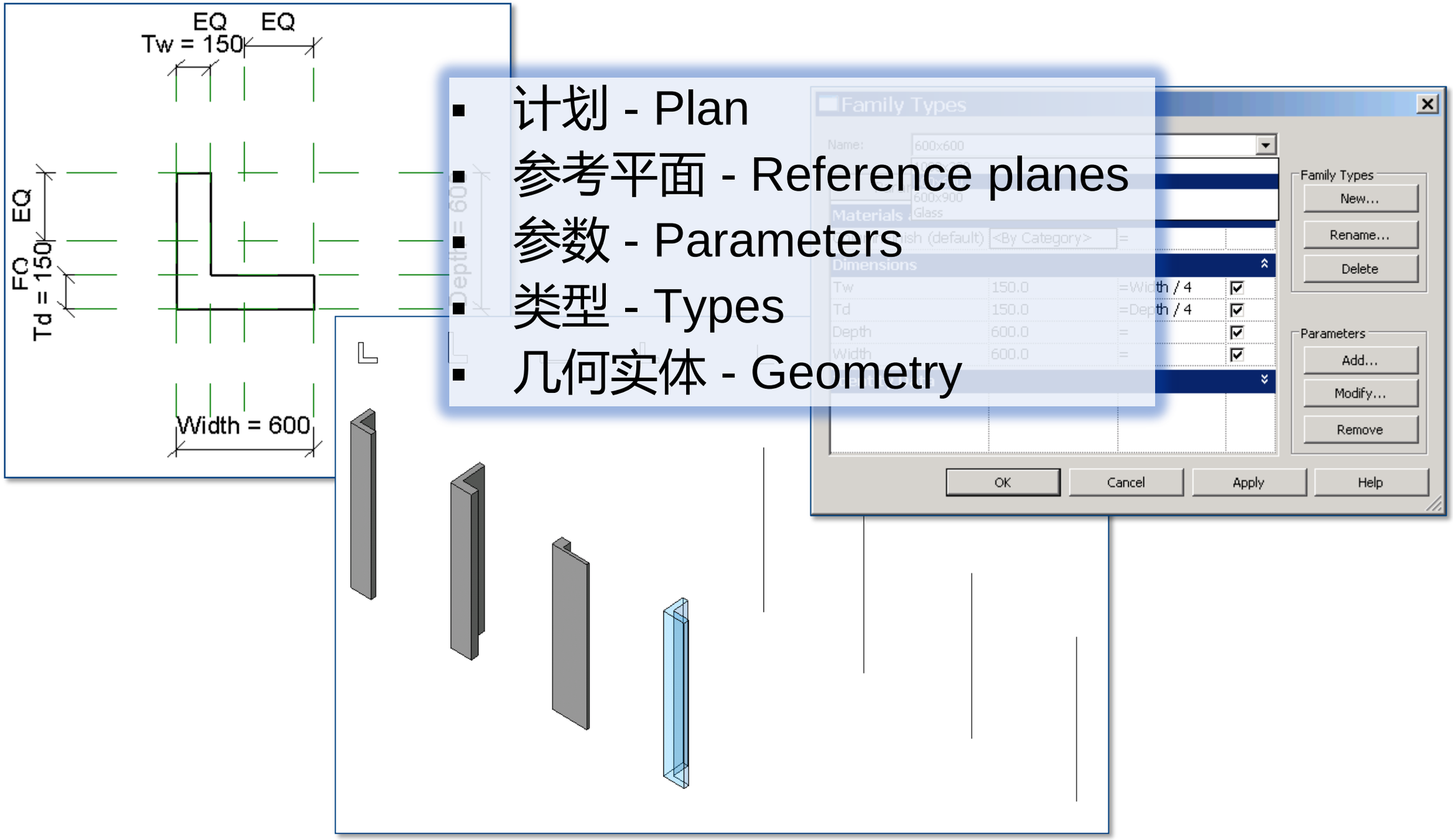
Family API 概览

族的特定类和方法

- FamilyManager 类:
  - 功能：增删改族类型，增删改族参数, 设置族参数值和设置公式
- Document 类中与族相关的方法:
  - FamilyManager – 返回FamilyManager对象，用来访问族的类型和参数
  - FamilyCreate – 返回FamilyItemFactory对象，用于在族文档中创建对象. 类似于在模型中几何实体对象
  - EditFamily – 用来打开一个已经加载到项目中的族
  - IsFamilyDocument – 用来判断Document是否是族文档
  - OwnerFamily – 返回当前族文档（Family Document）所代表的族Family

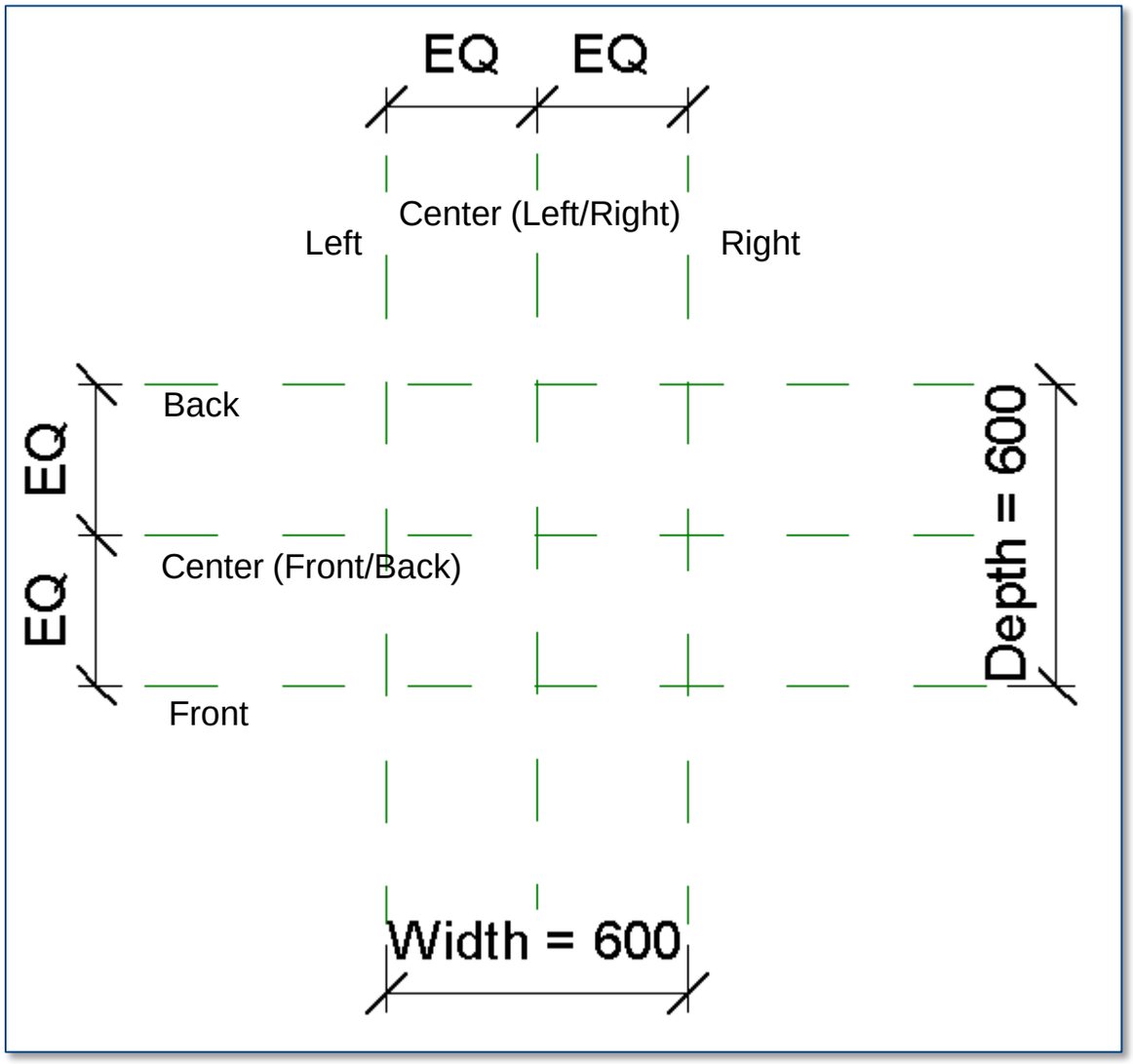
沿最佳实践步骤学习Family API

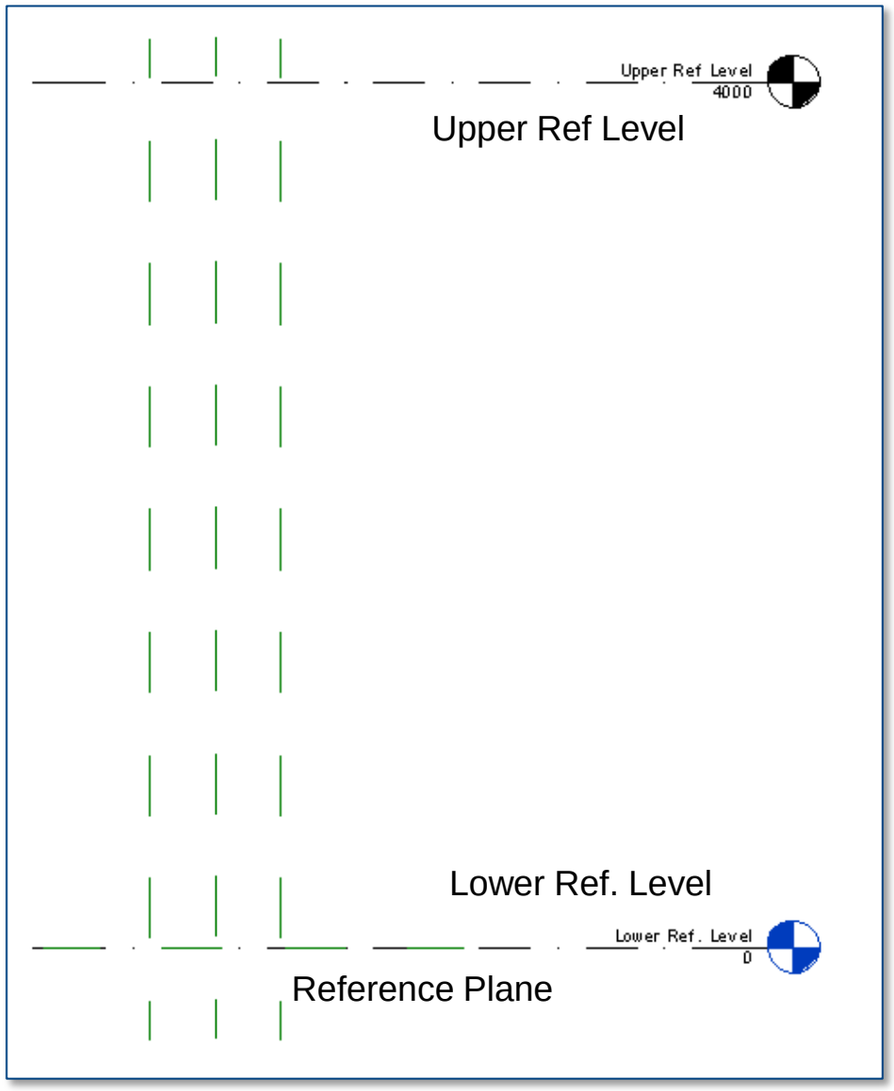
实例: 简单的L 型柱子

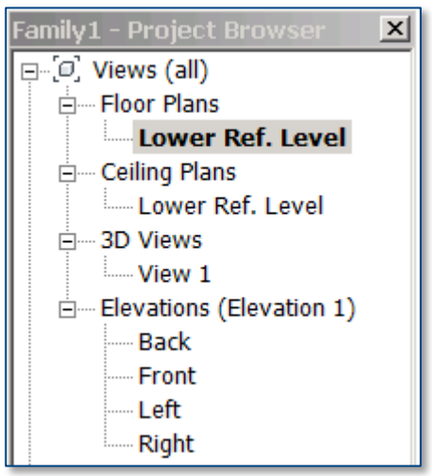


1. 规划

- 选择正确的族模板 (e.g., “Metric Column.rft”)
- 理解模板







1. 规划

确认文档是否是柱模板生成

```
Function ValidateDocument(ByVal rvtDoc As Document) As Boolean
    ''  our command works in the context of family editor only
    If Not rvtDoc.IsFamilyDocument Then
        TaskDialog.Show("Family API", "This works only in the family editor.")
        Return False
    End If
    ''  check if we have a right template
    Dim ownerFamily As Family = rvtDoc.OwnerFamily
    If ownerFamily Is Nothing Then
        TaskDialog.Show("Family API", "This document does not have Owner Family.")
        Return False
    End If
    ''  check the family category of this document
    Dim catColumn As Category = _
        rvtDoc.Settings.Categories.Item(BuiltInCategory.OST_Columns)
        If Not ownerFamily.FamilyCategory.Id.Equals(catColumn.Id) Then
            TaskDialog.Show("Family API", "Please open Metric Column.rft")
        End If
    Return True
End Function
```

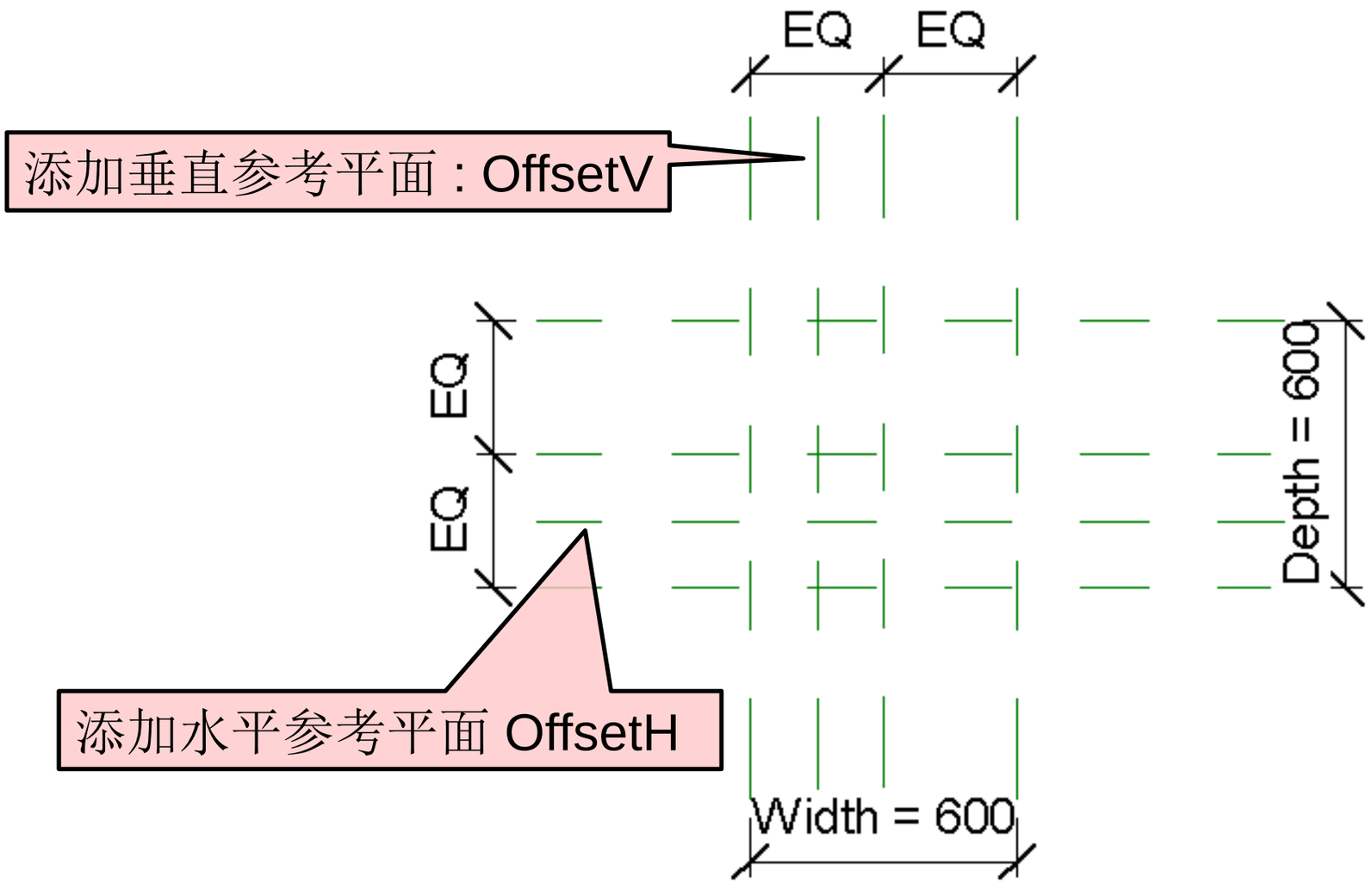
检查当前文档是否是族文档

获得当前文档所表达的 Family 对象

检查当前文档的族类别 是否是柱子

2. 布置参考平面

添加参考平面



2. 布置垂直参考平面

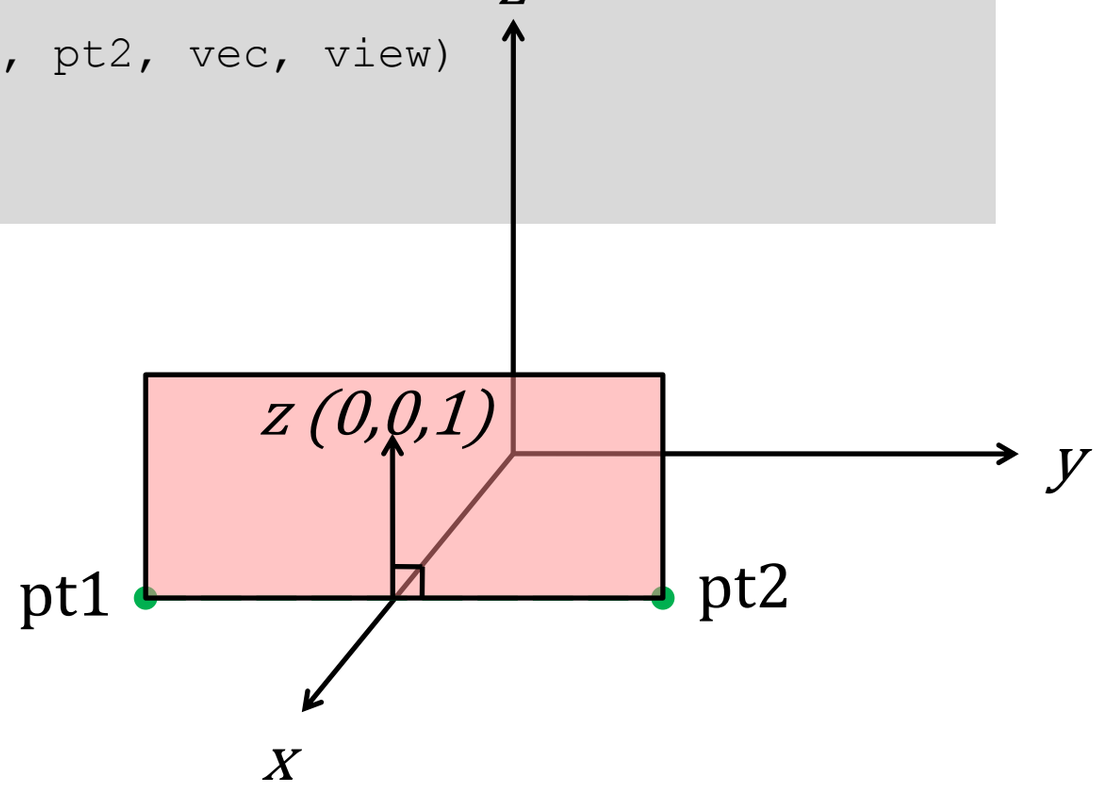
代码实例：垂直偏移

```
Sub AddReferencePlane_VerticalOffset()
    ''  create a reference plan, using NewReferencePlane
    Dim pt1 As New XYZ(-0.5, -2.0, 0.0) '' one end
    Dim pt2 As New XYZ(-0.5, 2.0, 0.0)  '' the other end
    Dim vec As XYZ = XYZ.BasisZ         '' perpendicular to the first line.
    Dim view As View = _
      Utils.FindElement(rvtDoc, GetType(ViewPlan), "Lower Ref. Level")

    Dim refPlane As ReferencePlane =
      m_rvtDoc.FamilyCreate.NewReferencePlane(pt1, pt2, vec, view)
    refPlane.Name = "OffsetV"
End Sub
```

2. 布局垂直参考平面

实例: 垂直偏移

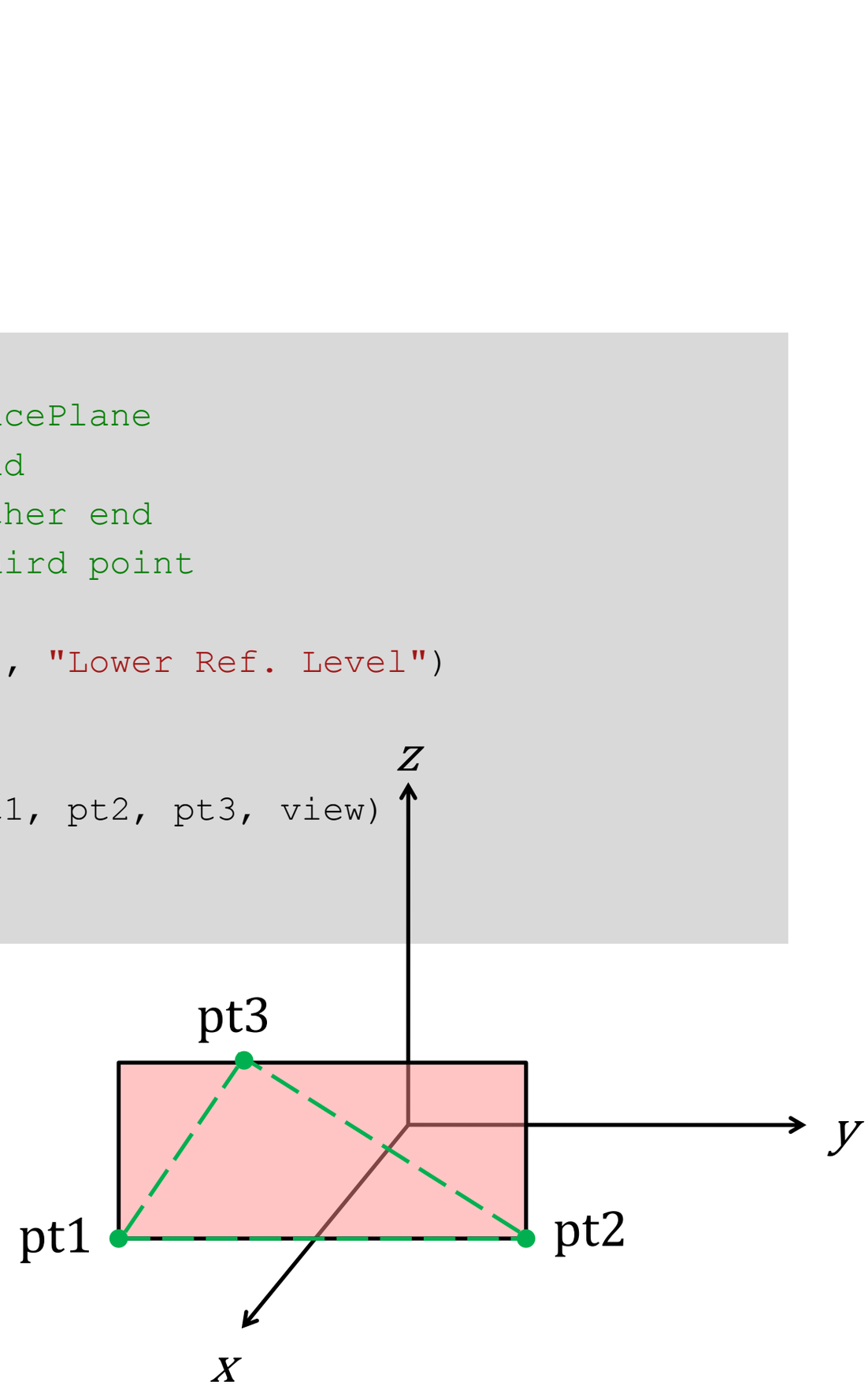
```
Sub AddReferencePlane_VerticalOffset()
    ''  create a reference plan, using NewReferencePlane
    Dim pt1 As New XYZ(-0.5, -2.0, 0.0) '' one end
    Dim pt2 As New XYZ(-0.5, 2.0, 0.0)  '' the other end
    Dim vec As XYZ = XYZ.BasisZ         '' perpendicular to the first line.
    Dim view As View = _
      Utils.FindElement(rvtDoc, GetType(ViewPlan), "Lower Ref. Level")

    Dim refPlane As ReferencePlane =
      m_rvtDoc.FamilyCreate.NewReferencePlane(pt1, pt2, vec, view)
    refPlane.Name = "OffsetV"
End Sub
```



2. 布置参照平面

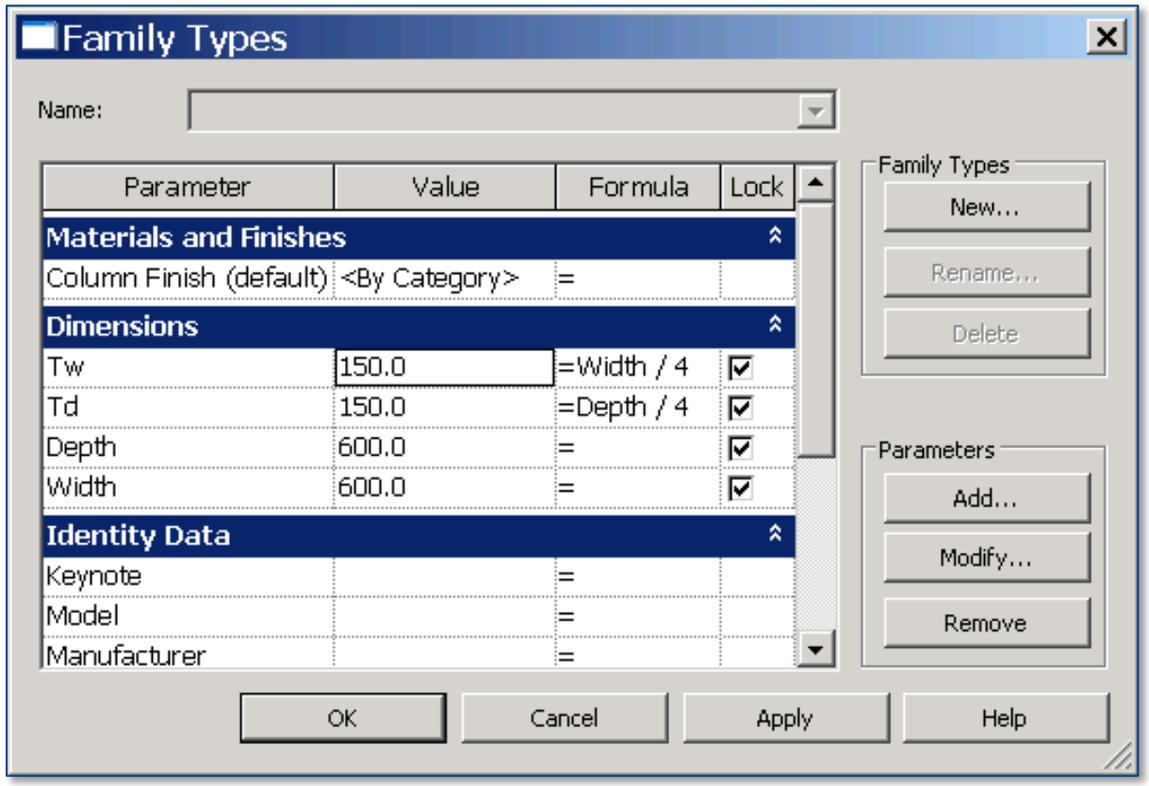
实例: NewReferencePlane2()

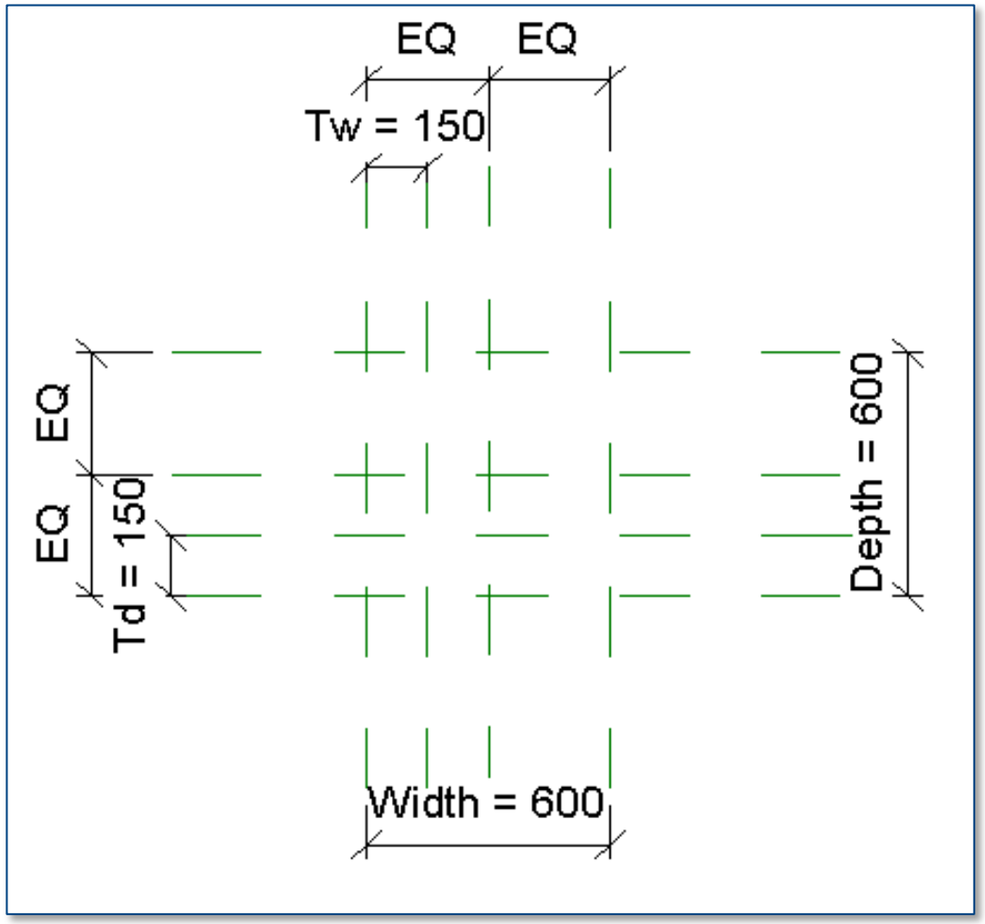
```
Sub AddReferencePlane_VerticalOffset2()
    ''  create a reference plan, using NewReferencePlane
    Dim pt1 As New XYZ(-0.5, -2.0, 0.0) '' one end
    Dim pt2 As New XYZ(-0.5, 2.0, 0.0)  '' the other end
    Dim pt3 As New XYZ(-0.5, -1.0, 1.0) '' the third point
    Dim view As View = _
      Utils.FindElement(rvtDoc, GetType(ViewPlan), "Lower Ref. Level")

    Dim refPlane As ReferencePlane =
      m_rvtDoc.FamilyCreate.NewReferencePlane2(pt1, pt2, pt3, view)
    refPlane.Name = "OffsetV"
End Sub
```



3. 新建参数

- 参数 - Parameters
- 尺寸标注- Dimensions

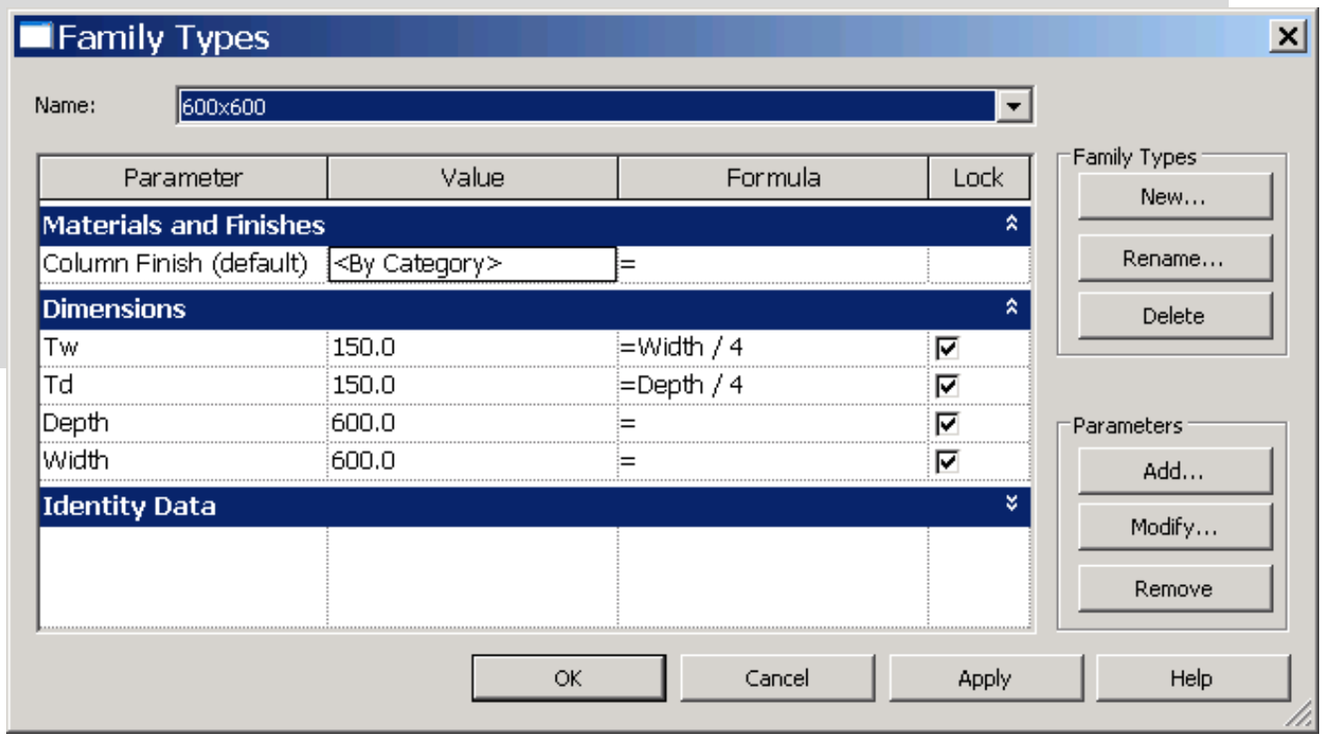




3a. 添加参数

Example: Tw

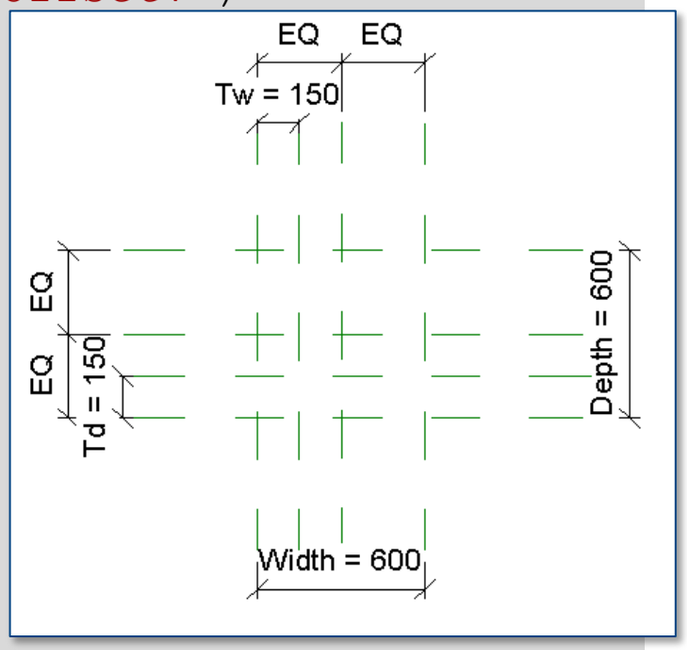
```
Dim m_familyMgr As FamilyManager = m_rvtDoc.FamilyManager

Sub AddParameter_Tw()
    ''  add a parameter "Tw"
    Dim isInstance As Boolean = False
    Dim paramTw As FamilyParameter = _ m_familyMgr.AddParameter( _
        "Tw", BuiltInParameterGroup.PG_GEOMETRY, ParameterType.Length, isInstance)
    ''  give initial values.
    Dim tw As Double = Utils.mmToFeet(150.0) '' in metric
    'Dim tw As Double = 0.5  '' in feet
    m_familyMgr.Set(paramTw, tw)
    ''  add a formula (optional)
    m_familyMgr.SetFormula( _
        paramTw, "Width / 4.0")
End Sub
```



3b 添加尺寸标注

Example: Tw

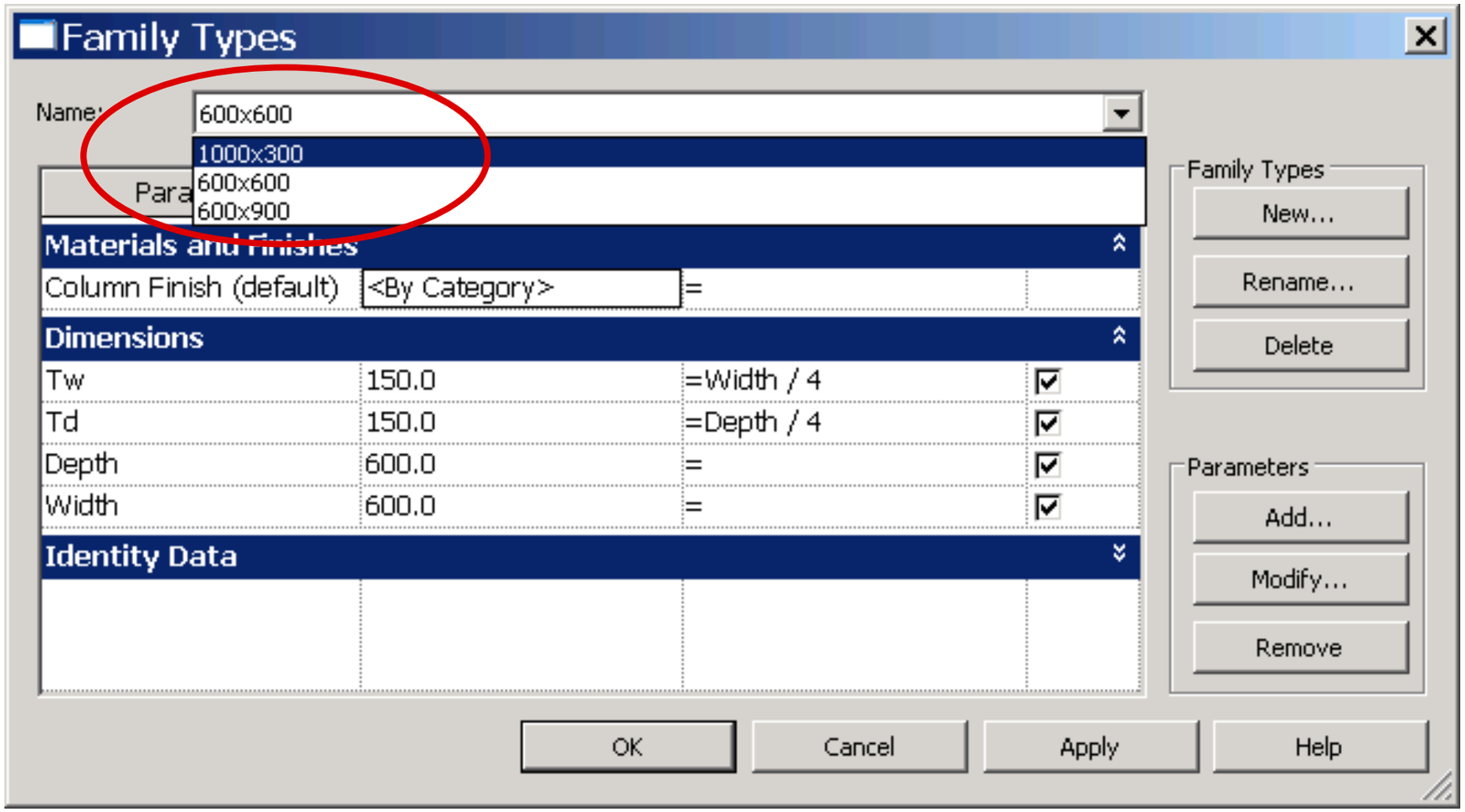
```
Sub AddDimention_Tw()
    ''  find the plan view that we want to place a dimension
    Dim pViewPlan As View = _
        Utils.FindElement(m_rvtDoc, GetType(ViewPlan), "Lower Ref. Level")
    ''  find two reference planes which we want to add a dimension between
    Dim ref1 As ReferencePlane = _
        Utils.FindElement(m_rvtDoc, GetType(ReferencePlane), "Left")
    Dim ref2 As ReferencePlane = _
        Utils.FindElement(m_rvtDoc, GetType(ReferencePlane), "OffsetV")
    ''  make an array of references
    Dim pRefArray As New ReferenceArray
    pRefArray.Append(ref1.Reference)
    pRefArray.Append(ref2.Reference)
    ''  define a dimension line
    Dim p0 As XYZ = ref1.FreeEnd
    Dim p1 As XYZ = ref2.FreeEnd
    Dim pLine As Line = m_rvtApp.Create.NewLineBound(p0, p1)
    ''  create a dimension
    Dim pDimTw As Dimension = _
        m_rvtDoc.FamilyCreate.NewDimension(pViewPlan, pLine, pRefArray)
    ''  add label to the dimension
    pDimTw.Label = m_familyMgr.Parameter("Tw")
End Sub
```



4. 添加多个主体类型

- 为了测试族
- 本例子中没有主体

5. 添加两个或更多类型

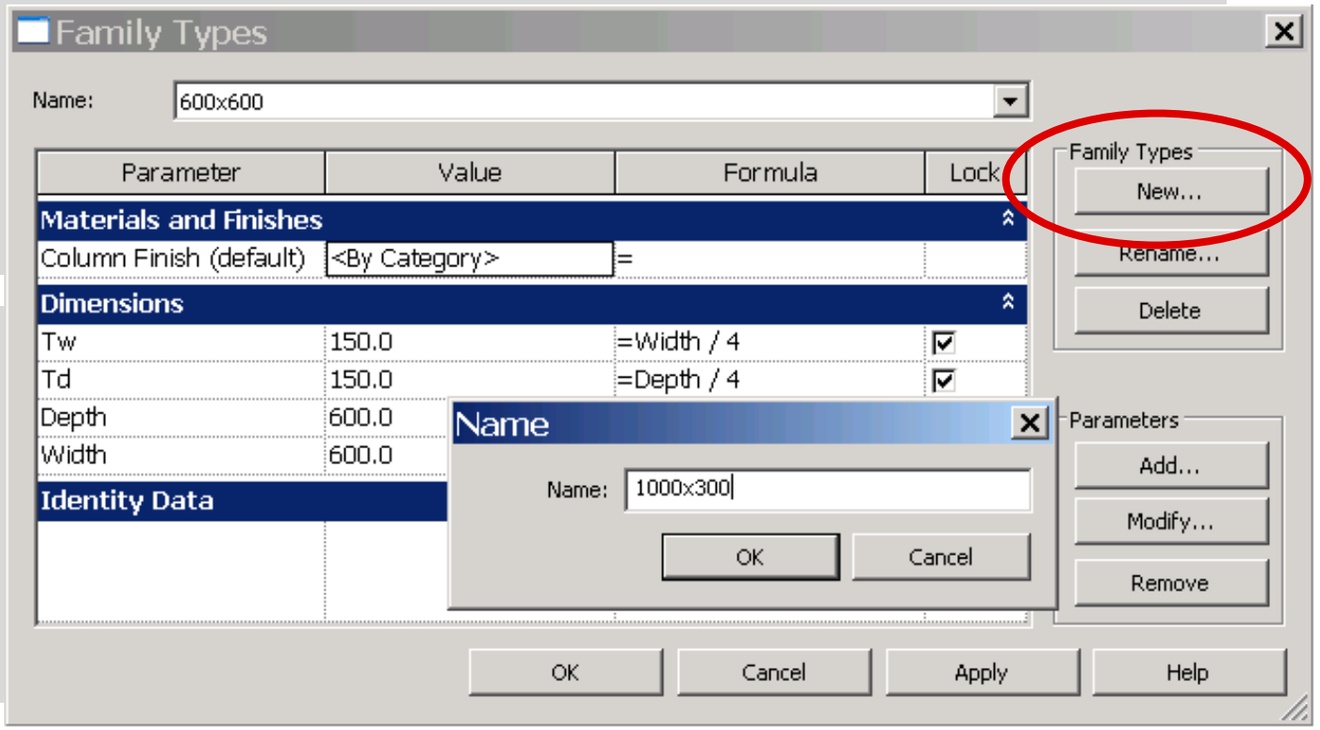

Family Types
Name:
600x600
1000x300
600x600
600x900
Para
Materials and Finishes
Column Finish (default)
<By Category>
=
Dimensions
Tw
150.0
=Width / 4
Td
150.0
=Depth / 4
Depth
600.0
=
Width
600.0
=
Identity Data
Family Types
New...
Rename...
Delete
Parameters
Add...
Modify...
Remove
OK
Cancel
Apply
Help

5. 添加两个或更多类型

Example: Width x Depth

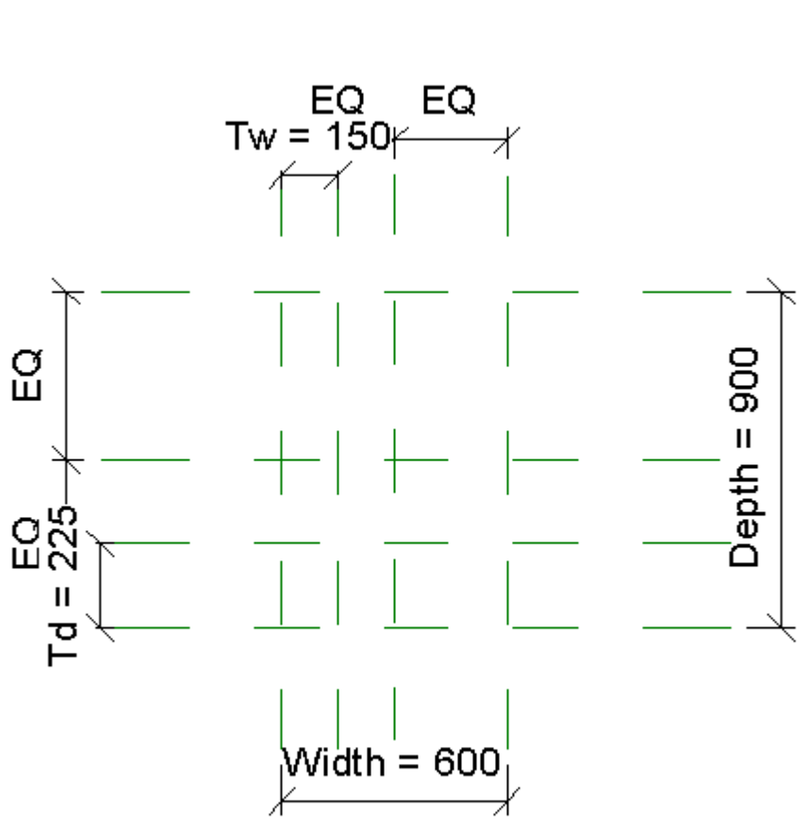
```
Sub AddType(ByVal name As String, ByVal w As Double, ByVal d As Double)
    ''  add new types with the given name.
    Dim type1 As FamilyType = m_familyMgr.NewType(name)
    ''  look for 'Width' and 'Depth' parameters and set them with the given values
    Dim paramW As FamilyParameter = m_familyMgr.Parameter("Width")
    Dim valW As Double = Utils.mmToFeet(w)
    If paramW IsNot Nothing Then
        m_familyMgr.Set(paramW, valW)
    End If
    Dim paramD As FamilyParameter = m_familyMgr.Parameter("Depth")
    Dim valD As Double = Utils.mmToFeet(d)
    If paramD IsNot Nothing Then
        m_familyMgr.Set(paramD, valD)
    End If
End Sub
```

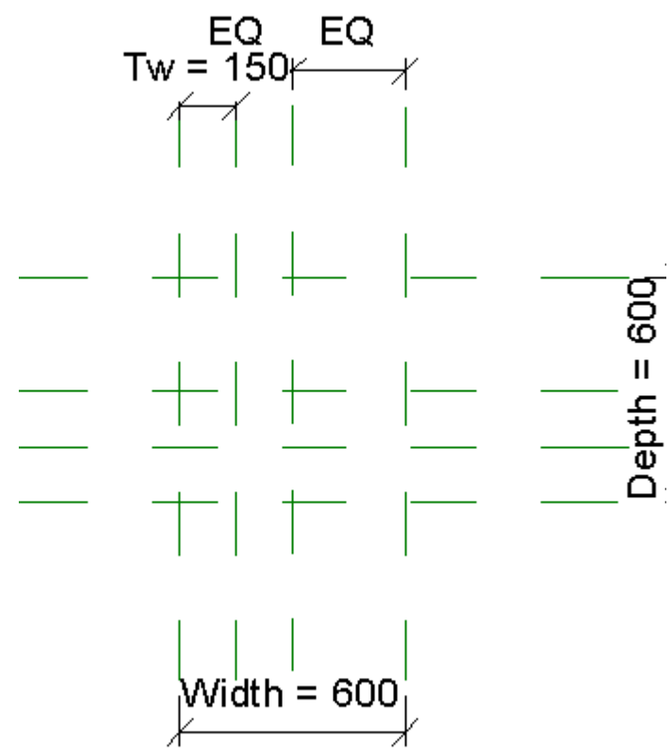
```
Sub AddTypes()
      '' AddType(name,Width,Depth)
      AddType("600x900", 600.0, 900.0)
      AddType("1000x300", 1000.0, 300.0)
      AddType("600x600", 600.0, 600.0)
End Sub
```

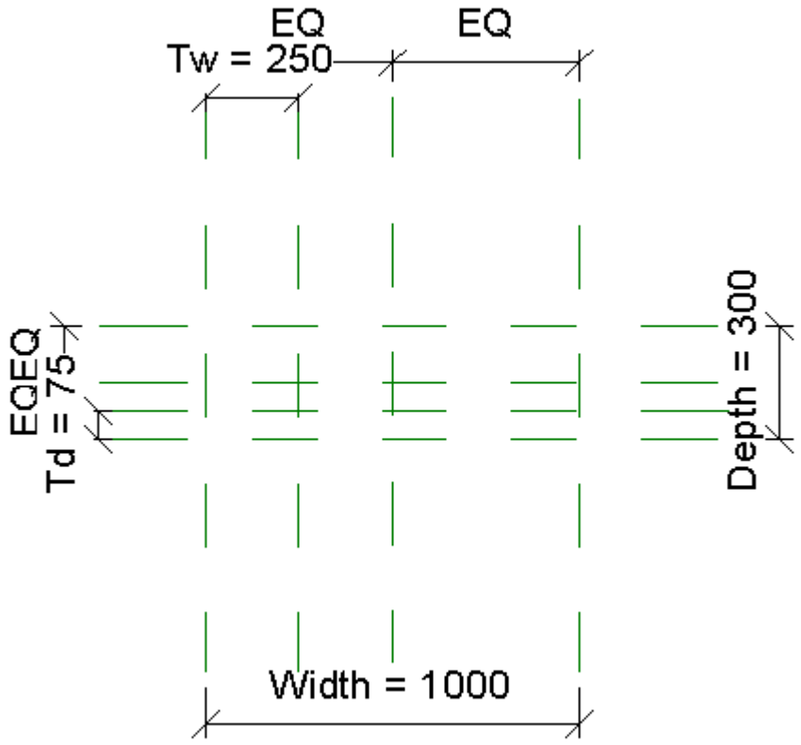


6. 测试类型

- 检查类型是否生效。
- 切换不同的类型结果Testing Procedure

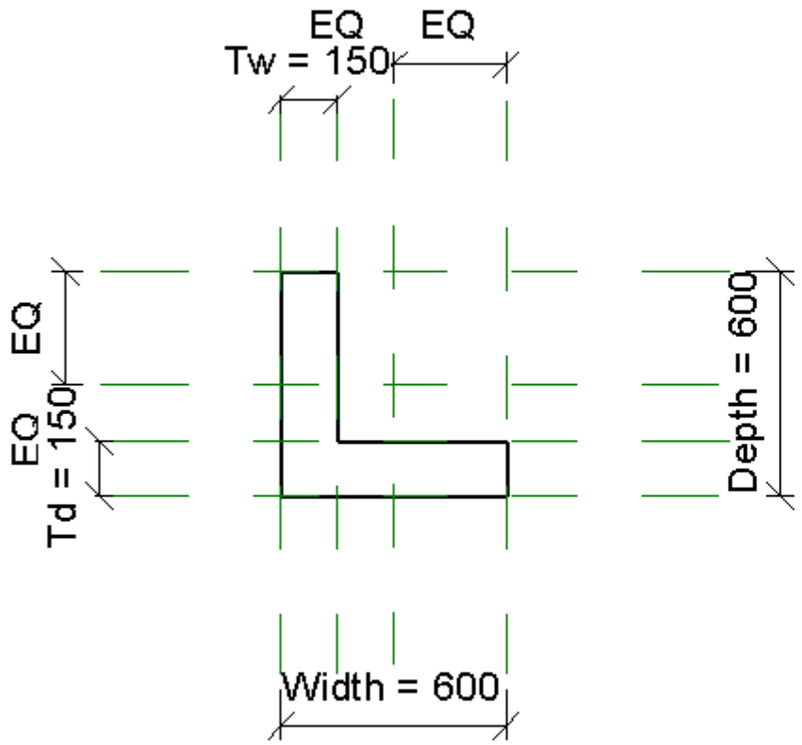


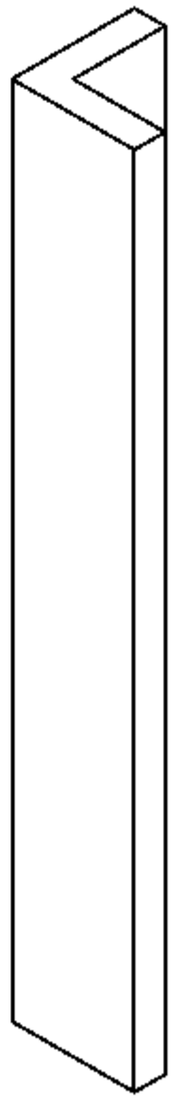




7. 添加几何体

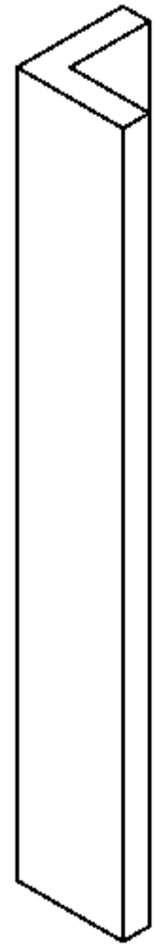
- 添加实体
- 添加对齐





7a. 添加几何体

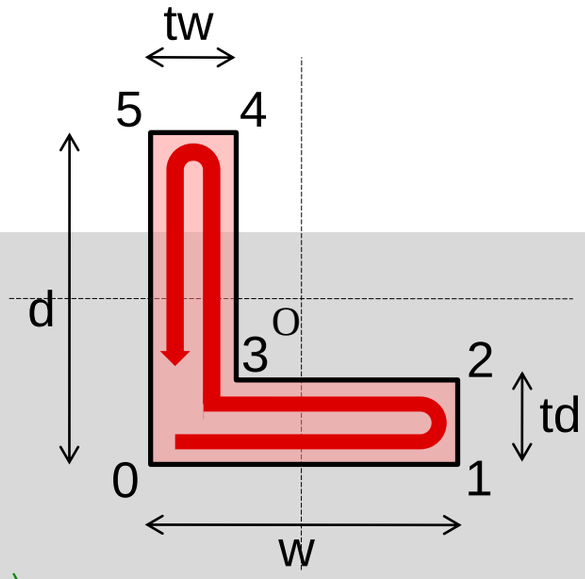
Example: 拉伸



```
Function CreateSolid() As Extrusion
    ''  (1) define a simple L-shape profile
    Dim pProfile As CurveArrArray = CreateProfileLShape()

    ''  (2) create a sketch plane
    Dim pRefPlane As ReferencePlane = _
        Utils.FindElement(m_rvtDoc, GetType(ReferencePlane), "Reference Plane")
    Dim pSketchPlane As SketchPlane = _
        m_rvtDoc.FamilyCreate.NewSketchPlane(pRefPlane.Plane)

    ''  (3) height of the extrusion. distance between Lower and Upper Ref Level
    Dim dHeight As Double = Utils.mmToFeet(4000)

    ''  (4) create an extrusion here. at this point.
    Dim bIsSolid As Boolean = True '' as oppose to void.
    Dim pSolid As Extrusion = _
        m_rvtDoc.FamilyCreate.NewExtrusion(bIsSolid, pProfile, pSketchPlane, dHeight)
    Return pSolid

End Function
```

7a.添加几何体

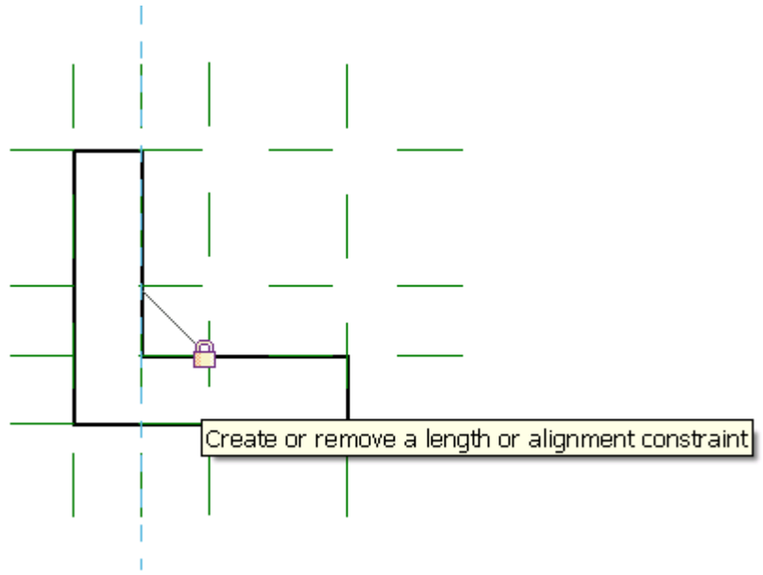
Example: L型的截面




```
Function CreateProfileLShape() As CurveArrArray
    Dim w As Double = Utils.mmToFeet(600)
    Dim d As Double = Utils.mmToFeet(600)
    Dim tw As Double = Utils.mmToFeet(150)
    Dim td As Double = Utils.mmToFeet(150)
    ''  define vertices (the last one is to make the loop simple)
    Const nVerts As Integer = 6 '' the number of vertices
    Dim pts() As XYZ = {New XYZ(-w / 2, -d / 2, 0), New XYZ(w / 2, -d / 2, 0), _
        New XYZ(w / 2, -d / 2 + td, 0), New XYZ(-w / 2 + tw, -d / 2 + td, 0), _
        New XYZ(-w / 2 + tw, d / 2, 0), New XYZ(-w / 2, d / 2, 0),
        New XYZ(-w / 2, -d / 2, 0)}
    ''  define a loop. define individual edges and put them in a curveArray
    Dim pLoop As CurveArray = m_rvtApp.Create.NewCurveArray
    Dim lines(nVerts - 1) As Line
    For i As Integer = 0 To nVerts - 1
        lines(i) = m_rvtApp.Create.NewLineBound(pts(i), pts(i + 1))
        pLoop.Append(lines(i))
    Next
    ''  then, put the loop in the curveArrArray as a profile
    Dim pProfile As CurveArrArray = m_rvtApp.Create.NewCurveArrArray
    pProfile.Append(pLoop)
    Return pProfile
End Function
```

7b. 创建对齐

Example: 把一个面对齐到 参考面：“OffsetV”




```
Sub AddAlignment_ReferencePlane(ByVal pSolid As Extrusion, _
    ByVal normal As XYZ, ByVal nameRefPlane As String)

    ''  get the plan view
    Dim pViewPlan As View = _
        Utils.FindElement(m_rvtDoc, GetType(ViewPlan), "Lower Ref. Level")

    ''  find reference planes
    Dim refPlane As ReferencePlane =
        Utils.FindElement(m_rvtDoc, GetType(ReferencePlane), nameRefPlane)

    ''  find the face of the solid
    Dim pFace As PlanarFace = Utils.FindFace(pSolid, normal, refPlane)

    '' create a locked alignment
    m_rvtDoc.FamilyCreate.NewAlignment( _
        pViewPlan, refPlane.Reference, pFace.Reference)

End Sub
```

7b. 创建对齐

Example: 与楼层对齐

```
Sub AddAlignment_Level( _
    ByVal pSolid As Extrusion, ByVal normal As XYZ, ByVal nameLevel As String)

    ''  which direction are we looking at?
    Dim pView As View = Utils.FindElement(m_rvtDoc, GetType(View), "Front")

    ''  find the upper ref level. FindElement() is a helper function.
    Dim pLevel As Level = Utils.FindElement(m_rvtDoc, GetType(Level), nameLevel)

    ''  find the face of the box. FindFace() is a helper function.
    Dim pFace As PlanarFace = Utils.FindFace(pSolid, normal)

    ''  create alignments
    m_rvtDoc.FamilyCreate.NewAlignment(pView, pLevel.PlaneReference, pFace.Reference)

End Sub
```

7b. 新建对齐

Example: L-shape solid alignments

```
Sub AddAlignments(ByVal pSolid As Extrusion)

    AddAlignment_Level(pSolid, New XYZ(0.0, 0.0, 1.0), "Upper Ref Level")
    AddAlignment_Level(pSolid, New XYZ(0.0, 0.0, -1.0), "Lower Ref. Level")
    AddAlignment_ReferencePlane(pSolid, New XYZ(1.0, 0.0, 0.0), "Right")
    AddAlignment_ReferencePlane(pSolid, New XYZ(-1.0, 0.0, 0.0), "Left")
    AddAlignment_ReferencePlane(pSolid, New XYZ(0.0, -1.0, 0.0), "Front")
    AddAlignment_ReferencePlane(pSolid, New XYZ(0.0, 1.0, 0.0), "Back")
    AddAlignment_ReferencePlane(pSolid, New XYZ(1.0, 0.0, 0.0), "OffsetV")
    AddAlignment_ReferencePlane(pSolid, New XYZ(0.0, 1.0, 0.0), "OffsetH")

End Sub
```

8. 重复步骤6、7直到满意

- 6. Flex Types and Host (Testing Procedure)
- 7. Add geometry

9. 测试族

- 创建项目来测试族

其它的类和方法

实体的可见性

```
Sub SetVisibility(ByVal pSolid As Extrusion)
    ''  set the visibility of the model not to shown in coarse.
    Dim pVis As FamilyElementVisibility = _
        New FamilyElementVisibility(FamilyElementVisibilityType.Model)
    pVis.IsShownInCoarse = False

    pSolid.SetVisibility(pVis)
End Sub
```

其它类和方法

关联族的参数和实体的参数

```
Sub addMaterials(ByVal pSolid As Extrusion)
    ''  get the material id that we are intersted in (e.g., "Glass")
    Dim pMat As Material = Utils.FindElement(m_rvtDoc, GetType(Material), "Glass")
    Dim idMat As ElementId = pMat.Id
    ''  add a parameter for material finish
    Dim paramFamilyMaterial As FamilyParameter = _
        m_familyMgr.Parameter("Column Finish")

    ''  associate material parameter to the family parameter we just added
    Dim paramSolidMaterial As Parameter = pSolid.Parameter("Material")
    m_familyMgr.AssociateElementParameterToFamilyParameter( _
        paramSolidMaterial, paramFamilyMaterial)

    ''  let's add another type with Glass finish
    AddType("Glass", 600.0, 600.0)
    m_familyMgr.Set(paramFamilyMaterial, idMat)
End Sub
```

Family API Labs 练习

学习资料

步进式学习创建柱族

- Lab1 – 创建矩形柱族
- Lab2 – 创建L形柱族
- Lab3 – 创建公式和设定材质
- Lab4 – 可视化控制

包含步进式学习指导

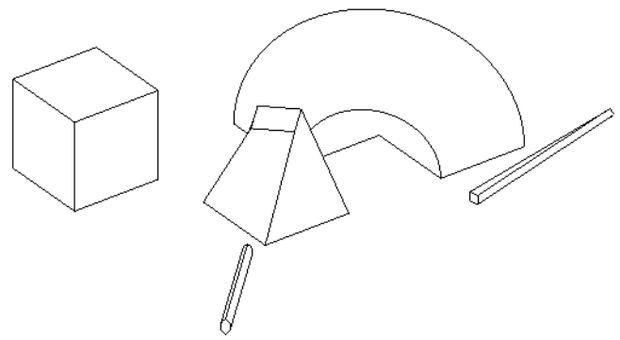
提供 VB.NET 和C# 代码

Revit SDK 中的例子介绍

Learning resources

在 <SDK folder>\Samples\FamilyCreation 目录中

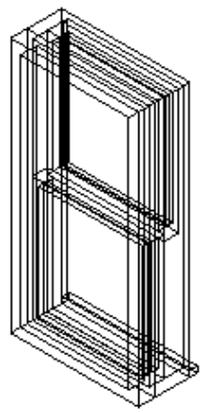
- AutoJoin
- AutoParameter
- DWGFamilyCreation
- GenericModelCreation

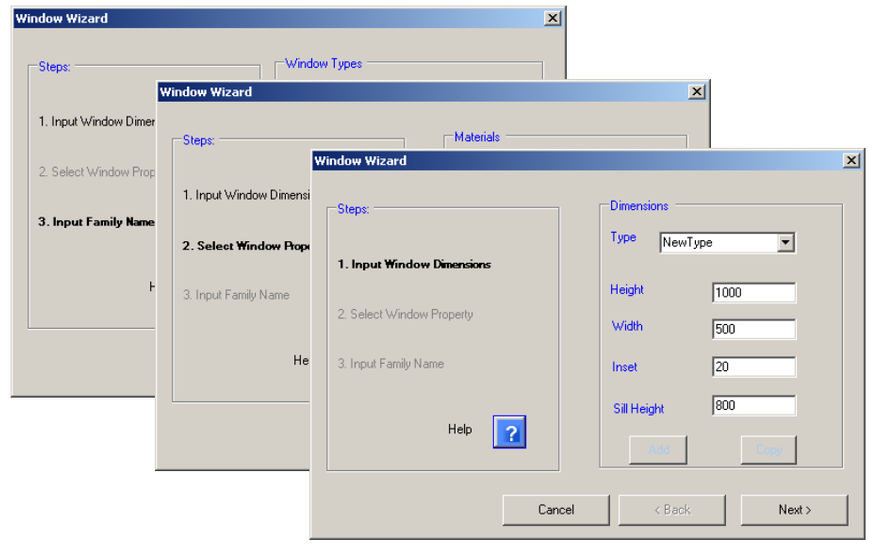


Revit SDK 中的例子介绍(续..)

Learning resources

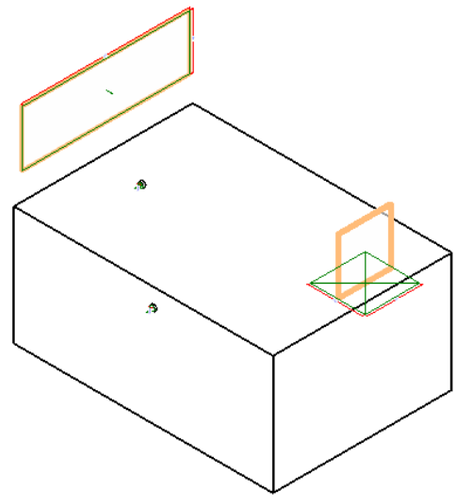
- TypeRegeneration
- ValidateParameters
- WindowWizard





Revit SDK 中的例子介绍(续..)

Learning resources



- CreateAirHandler – RME
- CreateTruss – RST

更多 ...

学习资料

- Revit 2011 API Developer Guide
  - Section 13 Family Creation (pp. 147 ~ pp. 158)
- RevitAPI.chm
  - Reference guide. Include code snippet
- Recordings
  - Revit Family API Webcast (July 2009)
  - AU 2009 virtual class AU2009 CP9118-1 :
    "The New Autodesk® Revit® Family API: Everything is Relative" by Jeremy Tammik
  - In Revit 2010, but mostly still valid for Family API
  - Based on the exercise labs
- Discussion Groups >> Revit Architecture >> Revit API
- API Training Classes
- The Building Coder, Jeremy Tammik's Revit API Blog
- Autodesk Developer Network
- DevHelp Online for ADN members

总结

- 界面命令创建族
  - 什么是族?
  - 如何开始，族的分类，编辑器，能实现的功能
  - 最佳实践
- 用API编程创建族
  - 沿着最佳实践学习Learning along best practice
  - 例子： 创建L形柱子
  - 深入学习资料

Thank you very much!

Autodesk®