esri China BEIJING

2013 Esri China Developer Summit

Esri CityEngine 规则开发

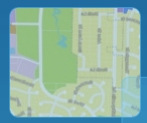
慕晓燕

内容

- CityEngine 快览
- CGA 基础
- CGA 建模示例
  - Demo：城市建筑物
  - Demo：郊区房屋
  - Demo：报表
- CityEngine Web Scene

2013 Esri China
Developer Summit

CityEngine 快览



esri China
BEIJING

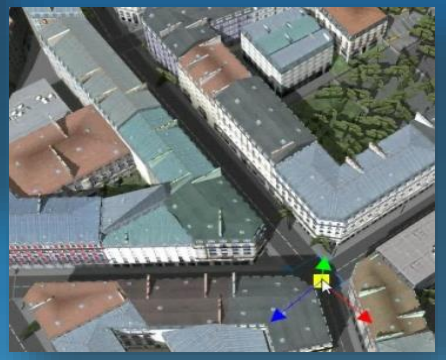
CityEngine

3D程序化建模与设计解决方案

– **3D 城市内容构建**
  - 数据 + 规则

几何 + 属性 + 规则

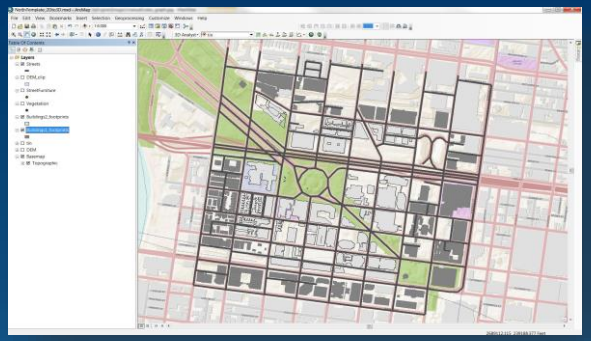
– **3D 城市设计**
  - 交互式
  - 规则驱动的3D设计



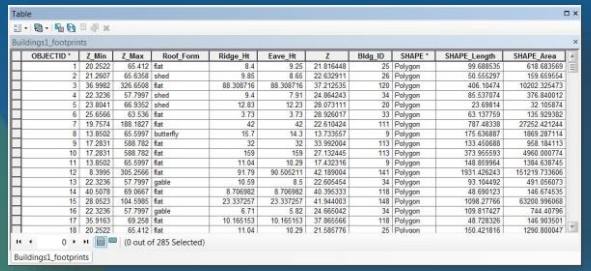
动态
参数化
编辑

3D 城市内容创建

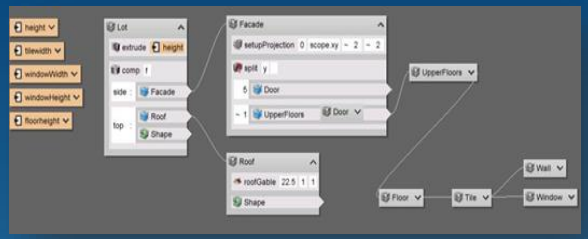
程序化的自动建模



几何



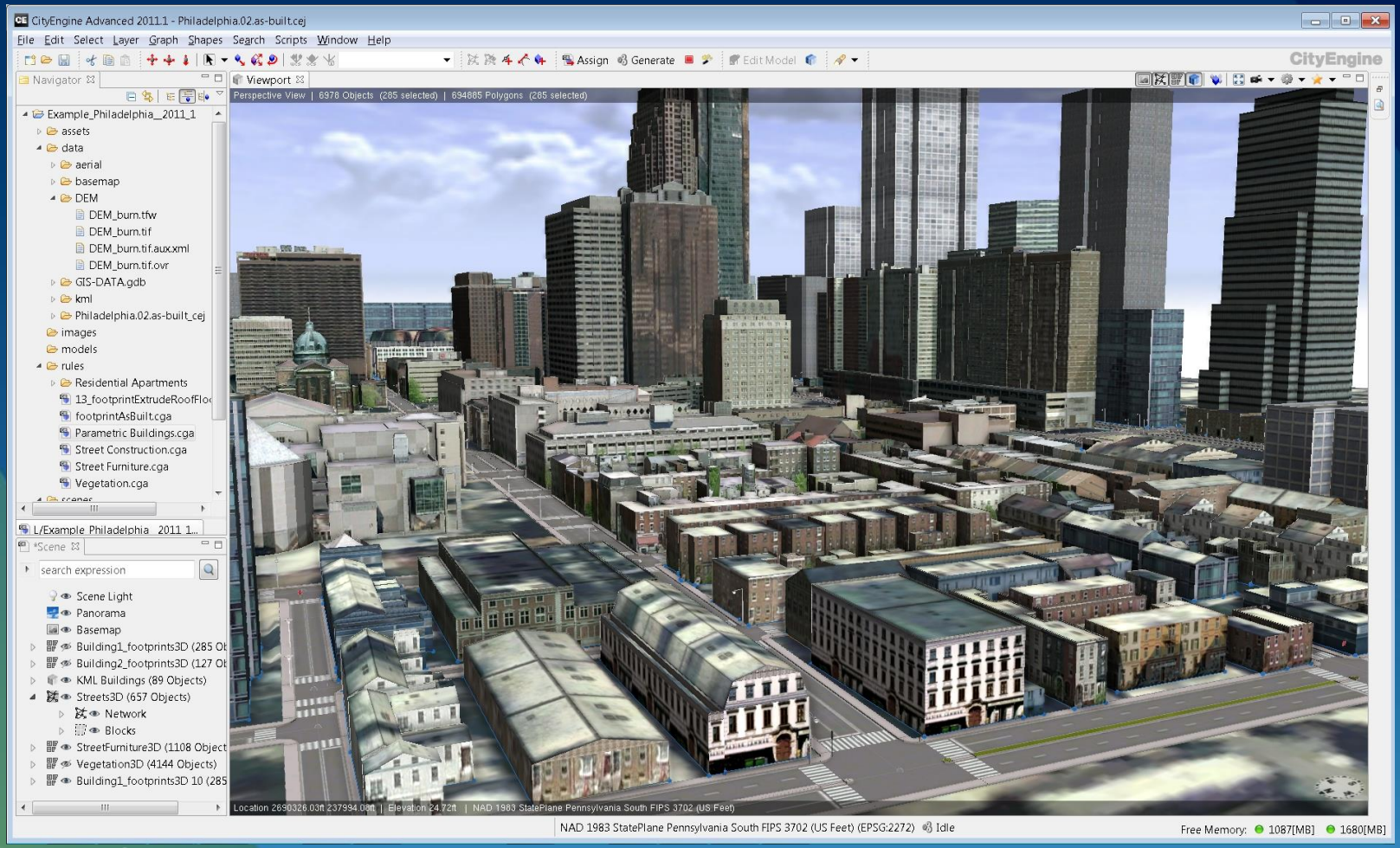
属性



规则

3D 城市内容创建

程序化的自动建模



2013 Esri China Developer Summit

esri China BEIJING

3D 城市设计

3D 程序化、参数化、动态设计

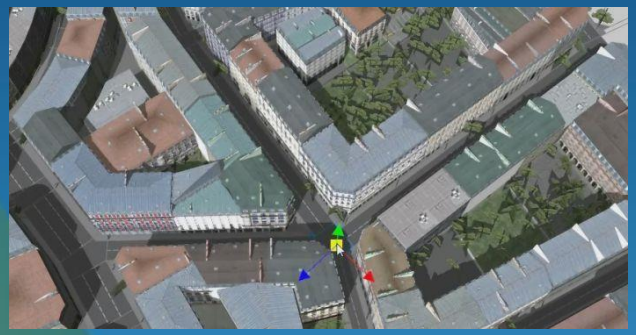
2013 Esri China Developer Summit

参数化编辑

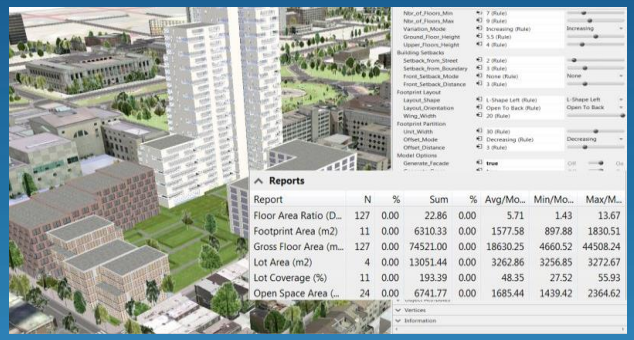
添加楼层

添加屋顶

动态编辑



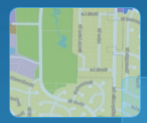
报表



esri China BEIJING

2013 Esri China
Developer Summit

CGA 基础



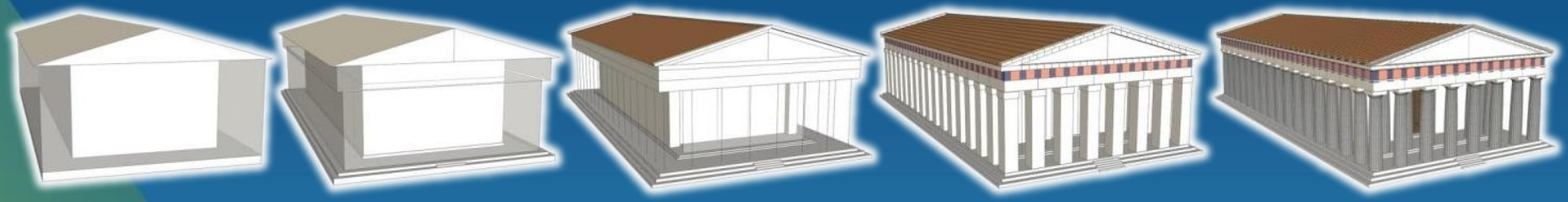
esri China
BEIJING

CGA 语言

CGA：Computer Generated Architecture；

CityEngine的独特语言；

越来越详细地迭代“进化”模型。



CGA 规则


```
Lot -->
        extrude(rand(10,100))
        s('0.5, '1, '0.5)
        center(xz)                    Building
```

Rule:

- **规则描述了shape 生成模型的过程。**

- Lot 是 initial shape;
- 在 Lot 的基础上做更改，产生Building;
- Lot 被替代;
- Building 称为叶子模型/Leaf Model。

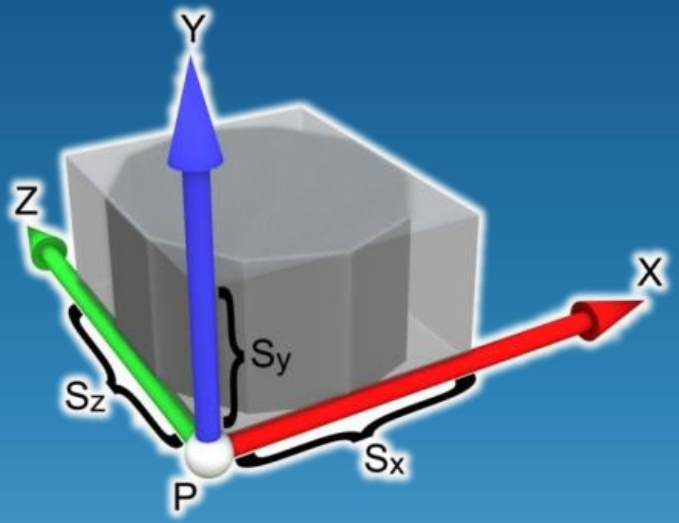
esri China BEIJING

CGA shape


```
Lot -->
        extrude(rand(10,100))
        s('0.5, '1, '0.5)
        center(xz)                    Building
```

Shape：

- ShapeSymbol：规则名字
- Parameters：参数
- Attributes：模型的构成与形状
- Geometry：几何对象，包含纹理、颜色、图形
- Scope：外接长方体范围
- Pivot：模型坐标系



CGA 编辑器

- 打开 .cga 文件
- Ctrl + Space 或者 **Alt + /** 命令提示
- Errors 红色下划线
- Warnings 黄色下划线
- 分屏显示、可视化

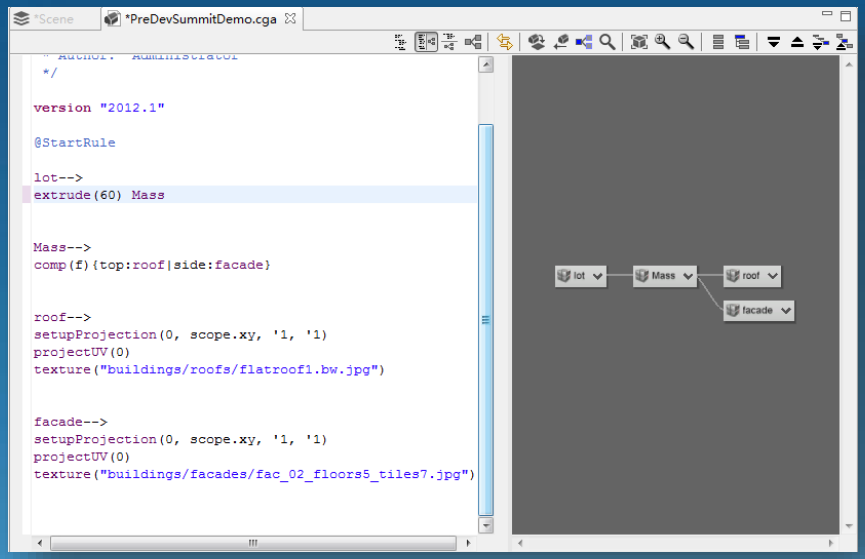
```
13# our first rule
14A --> extru
```

```
extrude(float a) -
extrude(x|y|z|world.x|world.y|world.z, float b) -
```

```
202
203 Shop ->
204     set(material.colormap, shopTexture(ceil(rand(6))))
205     setupProjection(0, scope.xy, scope.sx, scope.sy) projectUV(0)
206
```

CGA 编辑器

- 打开 .cga 文件
- Ctrl + Space 或者 **Alt + /** 命令补全
- Errors 红色下划线
- Warnings 黄色下划线
- 分屏显示、可视化



一个简单的规则


```
Lot --> extrude(10) Mass
```

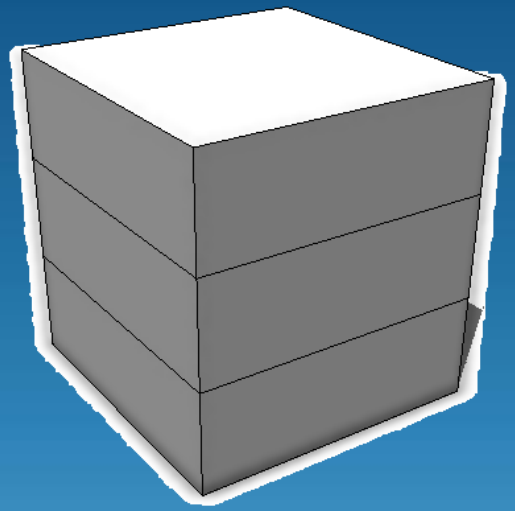
- Leaf Shape 形成结果模型

Shape 的替换


```
Lot  --> extrude(10) Mass                    Rule #1

Mass -->
     split(y){ ~4 : Floor. }*                Rule #2
     Floor
```

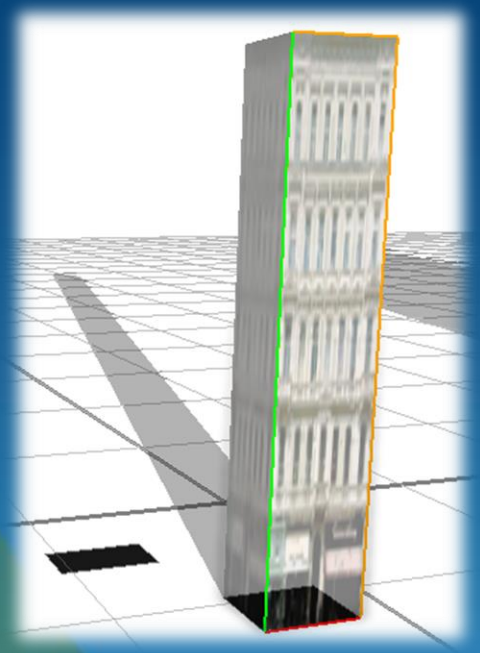
- Rule #2 是基于Mass的规则；
- Mass 被 Floor 替换。

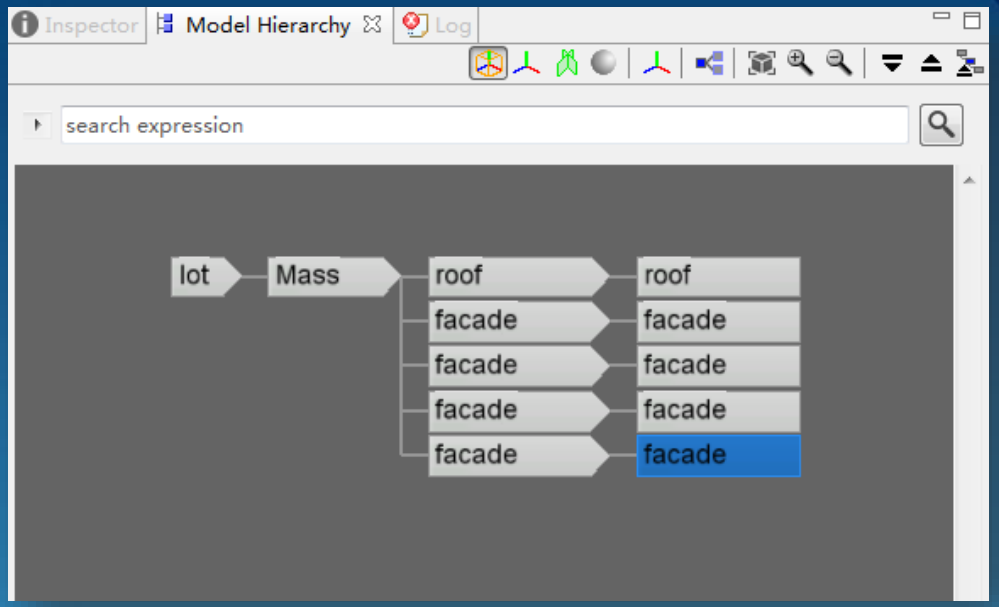


CGA 模型树查看器

查看已经生成的模型的层次；

对编辑和分析规则十分有帮助。





CGA 语法示例


```
attr height = 20
const groundfloor_height = 20
Lot --> extrude(height) Mass
Mass --> comp(f) { top : Roof.
                 | front : Frontfacade
                 | side : Facade}
# Facade
Facade -->
    setupProjection(0, scope.xy, 1,0.5, 1)
    split(y){groundfloor_height : Groundfloor |
    ~1 : UpperFloors}
Groundfloor -->
    case scope.sx > 10 : color("#cccccc")
    else : color("#ffcccc")
```

Boolean，float， string 表达式

`1, 0.5, ("#cccccc"), scope.sx > 10`

CGA 指定关键字

`attr, top, front, case`

CGA 操作

`extrude(height), comp(f)`

规则

`Lot, Mass, Facade`

自定义属性、常量、函数

`height, groundfloor_height`

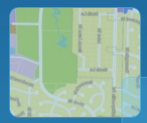
注释

`#Facade, //, /* ... */`

esri China BEIJING

Demo：城市建筑物

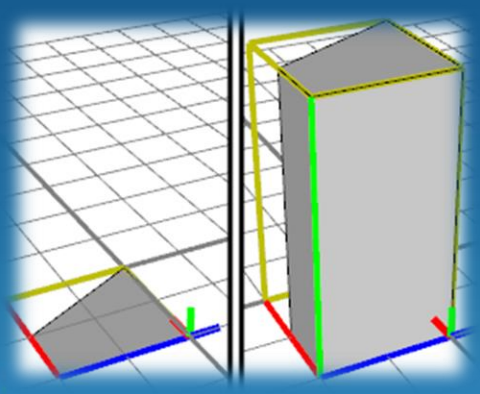


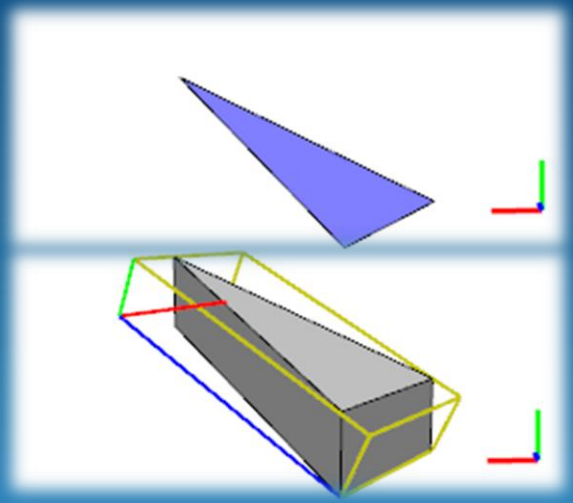



CGA：拉伸

2D Footprint → 3D building

```
extrude(height)
extrude(axisWorld, height)
```

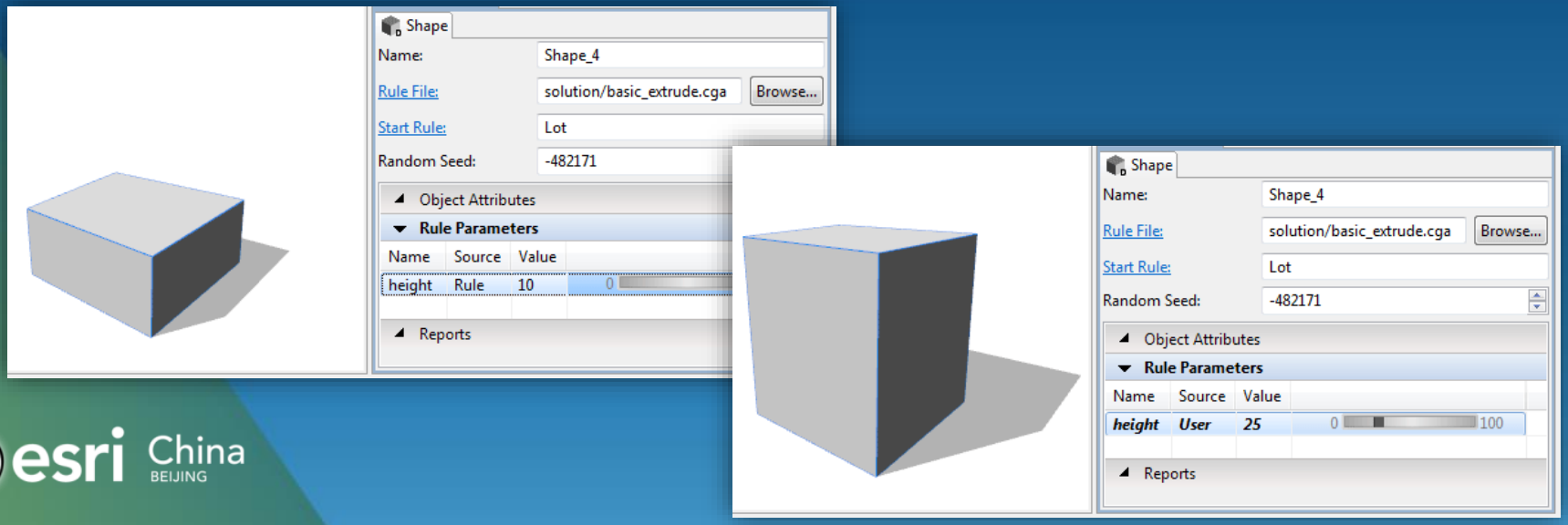




CGA：添加属性

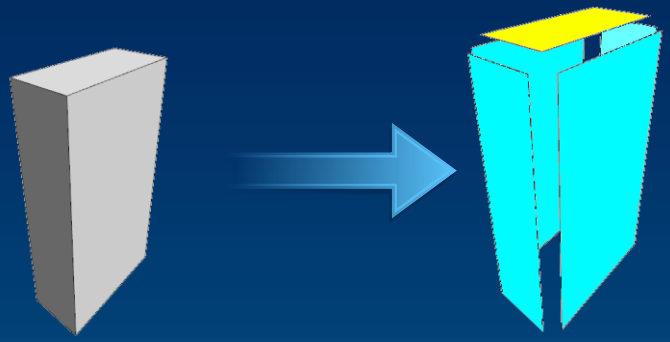

```
attr height = 10
Lot --> extrude(height) Mass
```

规则文件：添加属性 height；
查看器：属性 –>参数；
参数可以在查看器中修改。



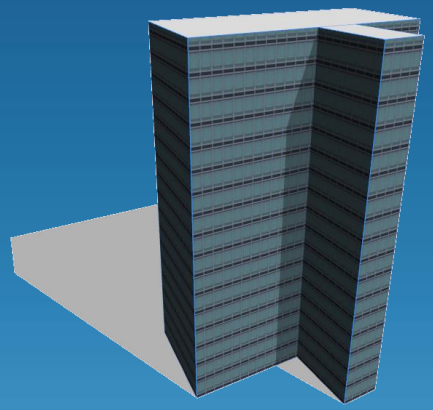
CGA：组件分割

分割模型的不同组件:



- 面组件分割:
  - `comp(f){top : Roof | side : Facade}`
- 边和节点分割:
  - `comp(e), comp(v)`
- 通过不同的语义进行选择

  `top, side, vertical, left, aslant, …`



CGA：纹理贴图

`setupProjection(uvset, axes, width, height)`

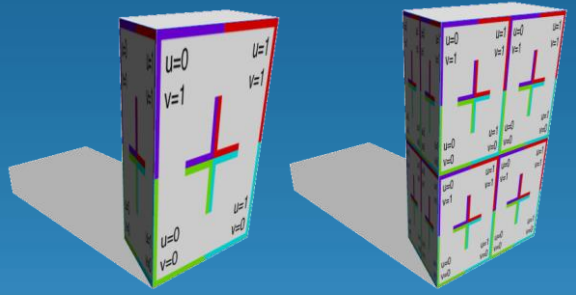
- 设置图片的贴图矩阵
- `width`、`height` 参数控制图片长宽或比例

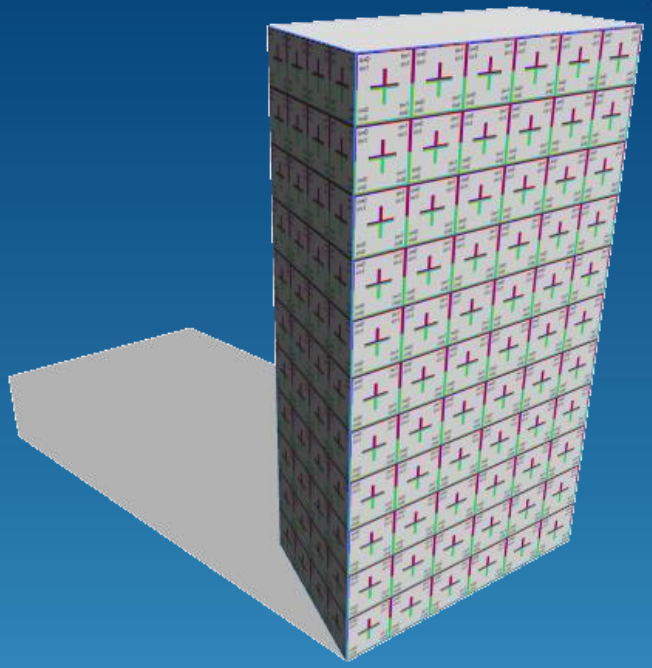
`projectUV(uvset)`

- 创建纹理图片坐标系

`texture("builtin:uvtest.png")`

- 加载一幅纹理图片

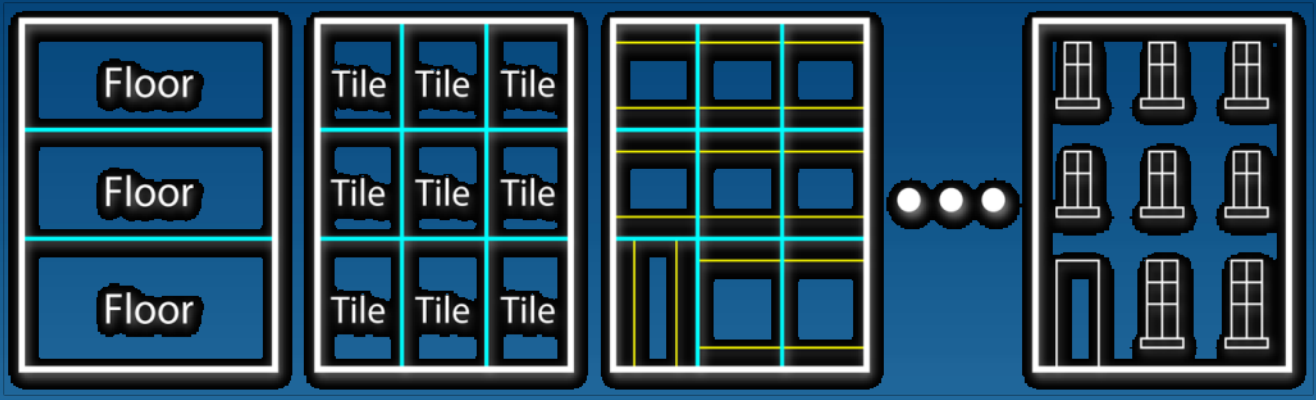




细化表面建模

通常情况下的表面细化结构，

Facade ⇢ Floor ⇢ Tile ⇢ Wall & Window/Door



CGA: 重复分割

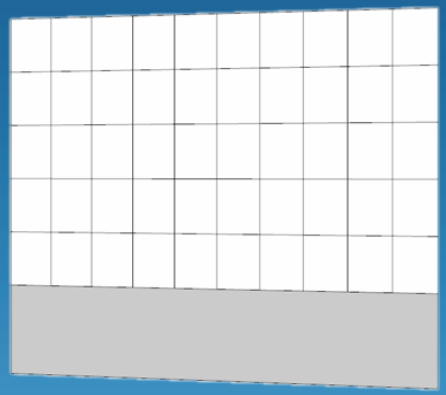
星号（*） 表示重复分割：

```
split(y){~width : A}*
```

~ 表示取平均值，确保适合模型表面尺寸。

常规分割和重复分割可以组合使用：

```
split(y){groundfloorheight : Groundfloor
        | {~ floorheight    : Floors}* }
```



CGA: “节奏”分割

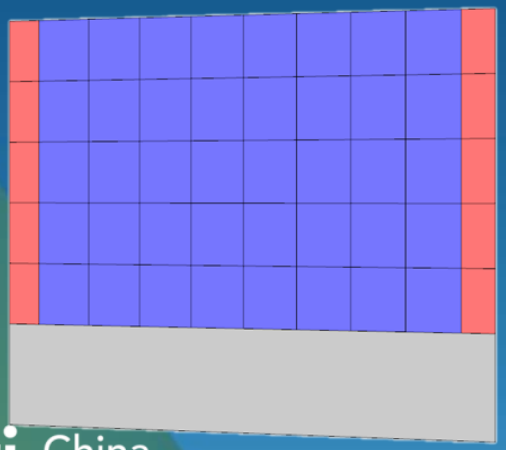
利用常规分割和重复分割组合使用。

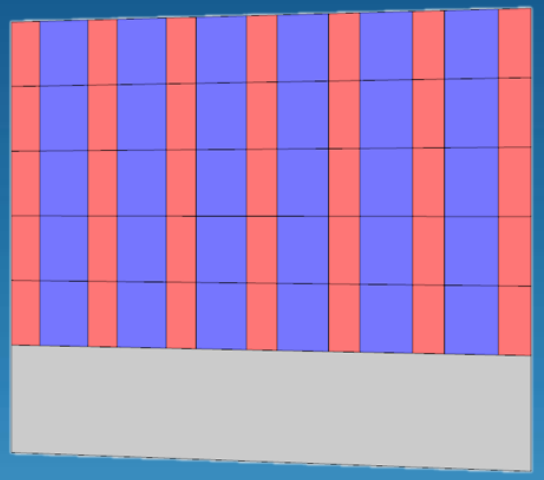
- A B* A

```
split(x){ widthA: TileA
        | {~ widthB : TileB}*
        | widthA : TileA }
```

- { A B }* A

```
split(x){ { widthA: TileA
          | ~ widthB : TileB }*
          | widthA : TileA  }
```



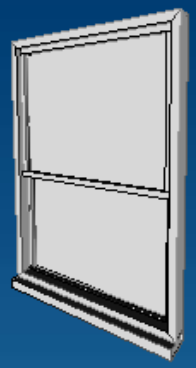


CGA: 插入模型素材

将外部的模型插入当前shape。

```
Asset --> i("asset.obj")
```

- 插入任意 obj 文件
- 插入静态模型（例如：树木、窗户等）
- 插入的对象可以被CGA规则进一步处理。

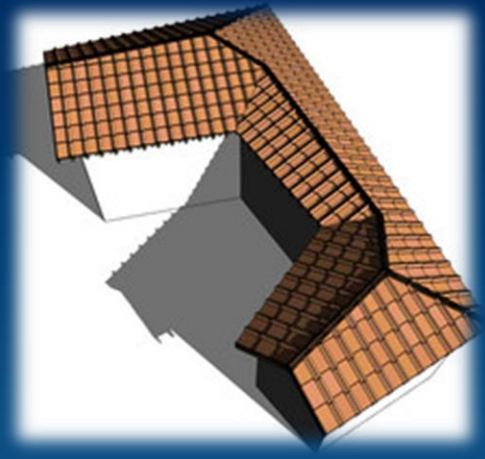


Demo：郊区房屋

CGA：屋顶样式

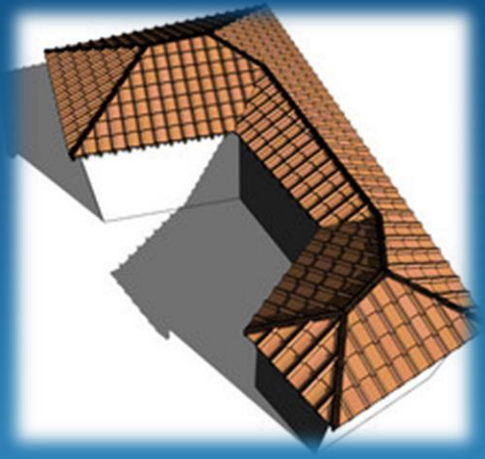
屋顶样式的控制：

- `roofGable()`
- `roofHip()`
- `roofPyramid()`
- `roofShed()`



额外设置：

- **屋顶倾角**
- **屋檐悬挂长度**



CGA：种植树木

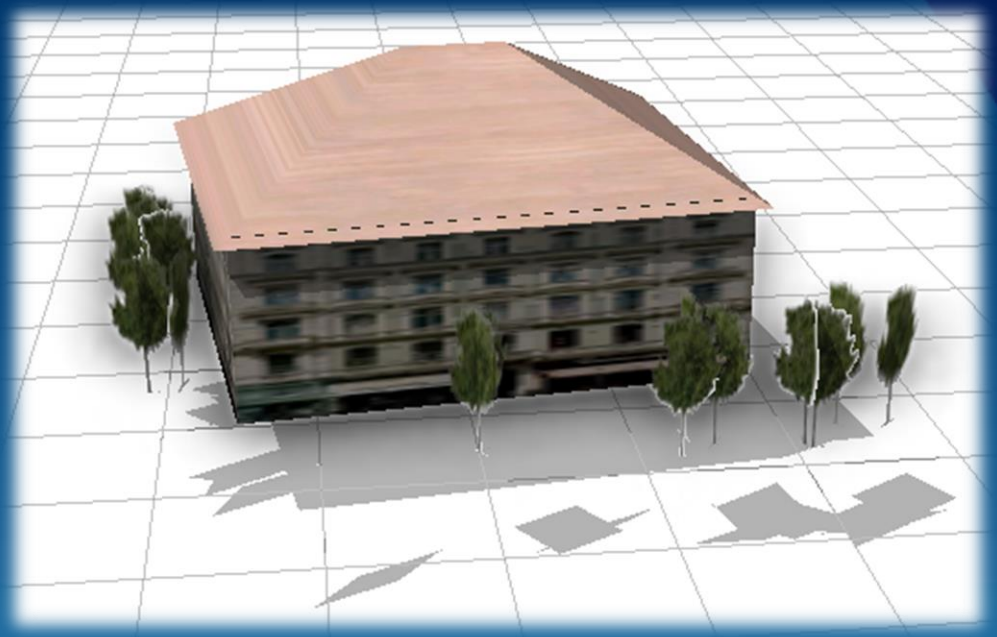
示例：

分割房屋边界：

- `shapeL ()`
- `shapeU ()`
- `shapeO ()`

种植树木：

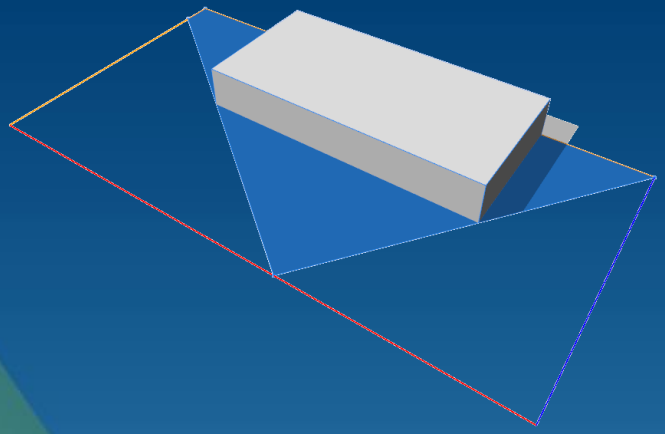
- `Scatter ()`

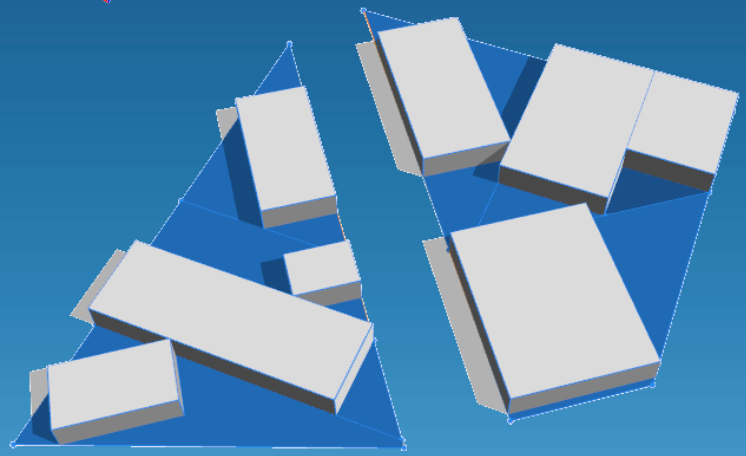


CGA：内接矩形

`innerRect`

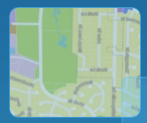
将 Shape 转化成内接矩形，建立规则的房屋。





2013 Esri China
Developer Summit

报表

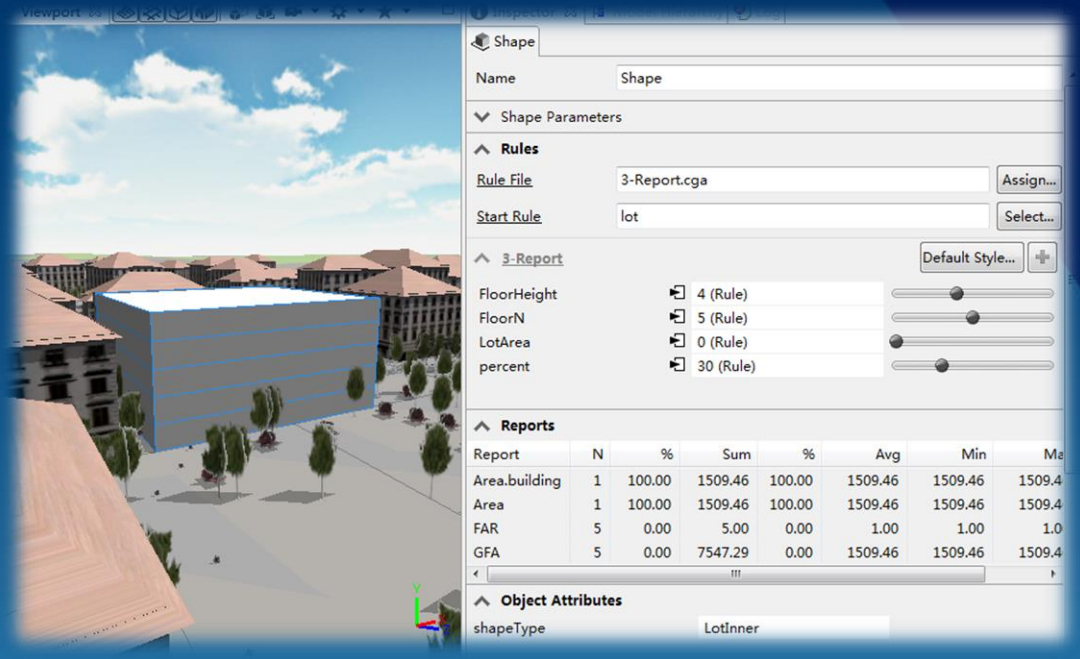


esri China
BEIJING

报表操作

在模型建立时收集数据；

可在规则任意位置使用。



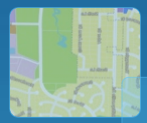

```
report(key,value)
```

示例

```
OfficeFloor -->
        report("GFA.Office",geometry.area)
```

2013 Esri China
Developer Summit

CityEngine Web Scene



esri China
BEIJING

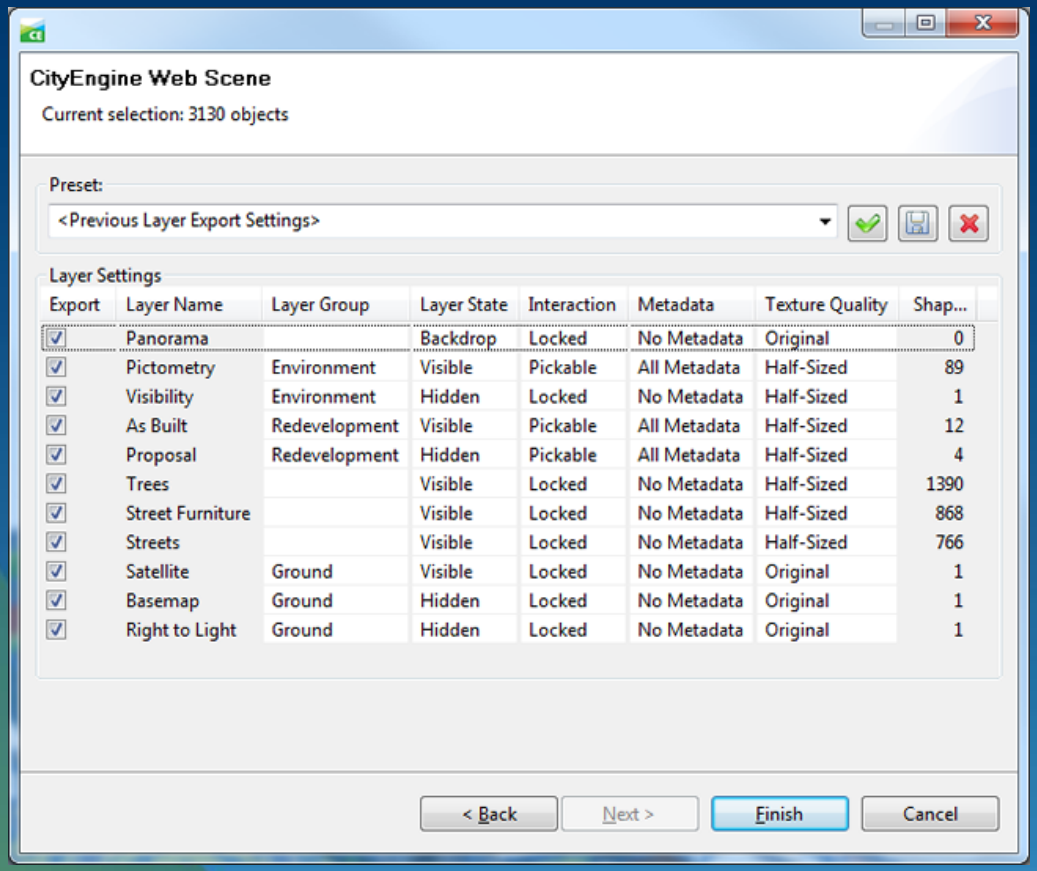
CityEngine Web Scene

在浏览器中查看 3D 城市景观和其他 3D 场景的 Web 应用程序。

- 基于 WebGL 技术
- 在 Web 浏览器中查看 3D 内容
- 无需安装插件

导出 3ws

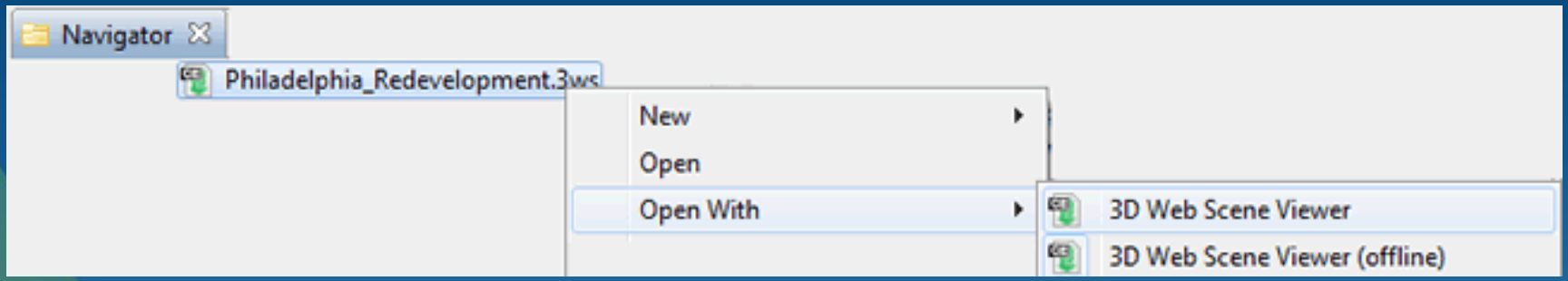
File → Export Models... → CityEngine Web Scene



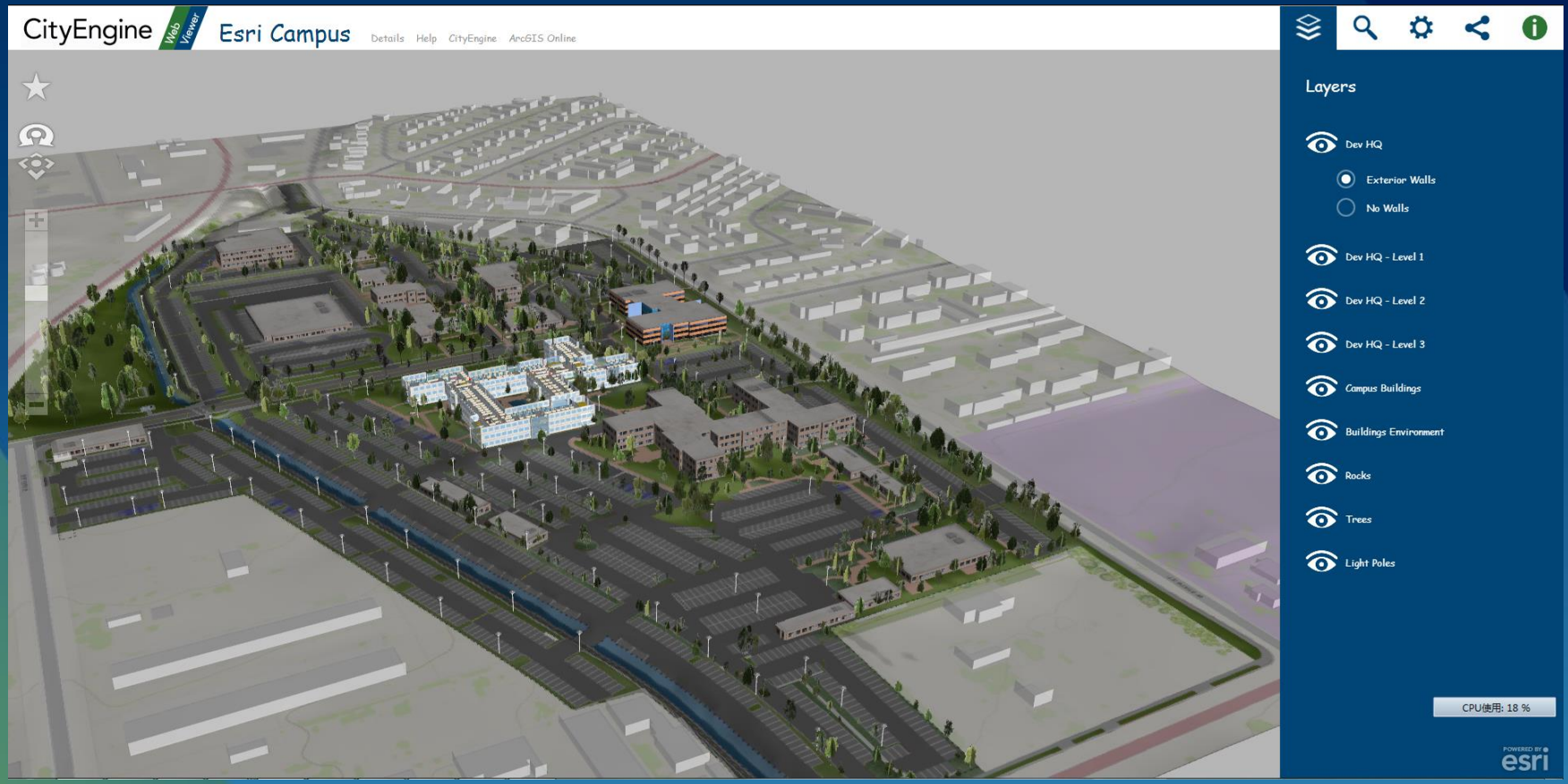
*.3ws 文件

浏览场景

Open With → CityEngine Web Scene Viewer



查看场景

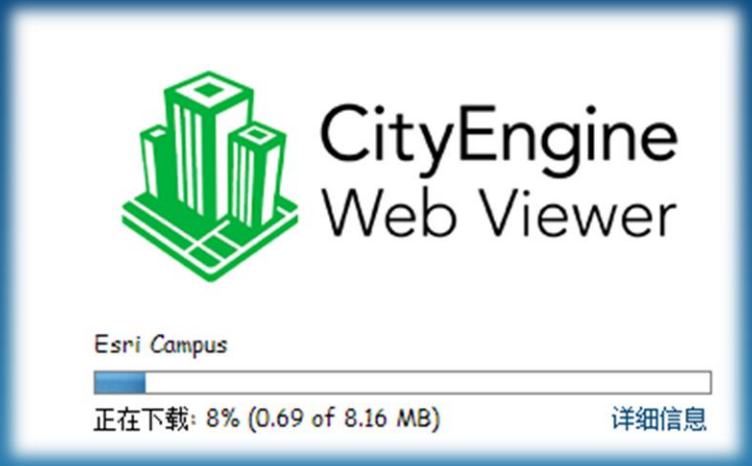

CityEngine
Web Viewer
Esri Campus
Details
Help
CityEngine
ArcGIS Online
Layers
Dev HQ
Exterior Walls
No Walls
Dev HQ - Level 1
Dev HQ - Level 2
Dev HQ - Level 3
Campus Buildings
Buildings Environment
Rocks
Trees
Light Poles
CPU使用: 18 %
POWERED BY
esri

分享场景

- Share as...
  - 发布到ArcGIS Online
  - 打包 zip 文件
- 上传到本地服务器

查看场景

- 缩放、平移和旋转来导航地图；
- 选择书签；
- 选择要查看的特定图层；
- 搜索对象、属性等；
- 更改光照和阴影；
- 通过各种社交媒体进行共享；
- 查看地图的其他信息。

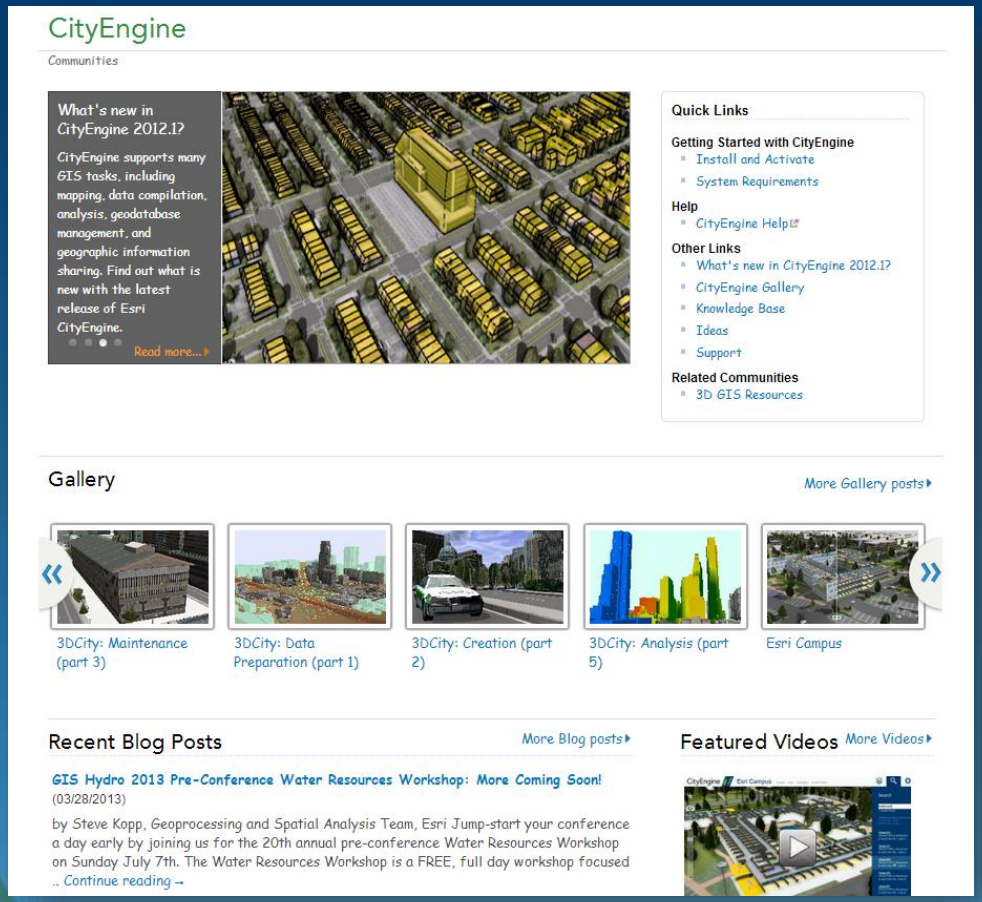


学习资源

2013 Esri China Developer Summit

CityEngine Community:

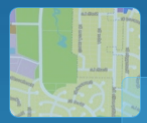
http://resources.arcgis.com/en/communities/city-engine/



esri China BEIJING

2013 Esri China
Developer Summit

谢 谢！



esri China
BEIJING