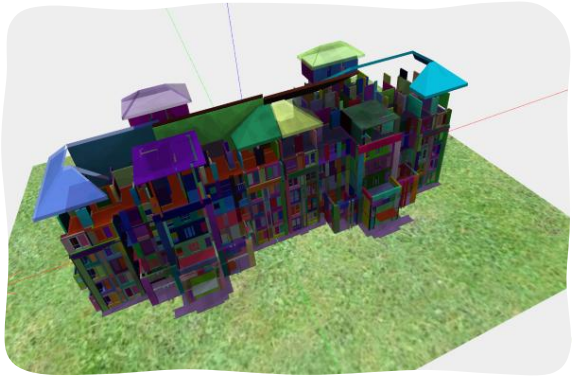
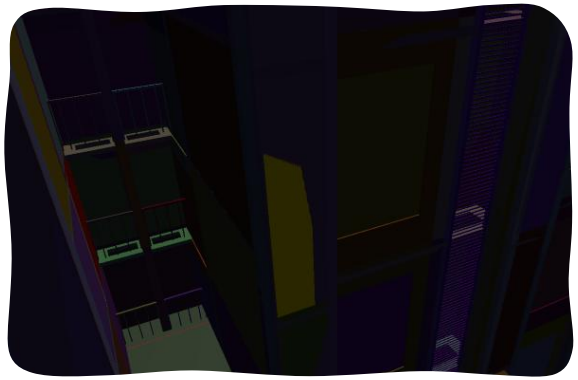
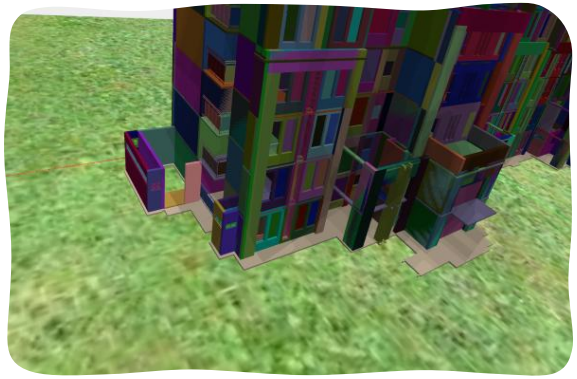
Flexible



Collaboration



Experience



Benefit

基于WebGL的BIM模型网络浏览

林佳瑞 2011310057

2012年4月11日

主要内容

1 背景及现状

2 研究内容

3 主要工作

4 结论与展望

研究背景

BIM蓬勃发展

管理、跟踪

Flash?Java?

Flexible--Web--Mobile

无插件、高效、平台无关

移动设备普及

复杂的客户端

ActiveX?html5?

WebGL

基于OpenGL、解决效率瓶颈、原生javascript

研究现状

- 当前关于**WebGL**的研究与开发非常多，涌现出许多优秀的**WebGL**程序及框架。

WebGL框架

- ✓ O3d
- ✓ Threejs
- ✓ PhiloGL
- ✓ GLGE

WebGL程序

- ✓ Quake2
- ✓ Maze3D
- ✓ Cloudmatch
- ✓ Copperlicht

WebGL资源

- ✓ Firefox&WebGL官网
- ✓ github
- ✓ wiki
- ✓ Google code/group

主要内容

1 背景及现状

2 研究内容

3 主要工作

4 结论与展望

研究内容

1. WebGL理论
图形绘制流程

绘制过程

BIM属性

4. 体积、面积、
名称查看与修改

三维表现

2. 几何、贴图、颜
色、材质、光照

数据交换

5. 4D系统数据导
出与应用

3. 位置、缩放、
旋转、透明等

模型编辑

UI交互 主要实现鼠标、键盘以及
简单触摸的响应。

主要内容

1 背景及现状

2 研究内容

3 主要工作

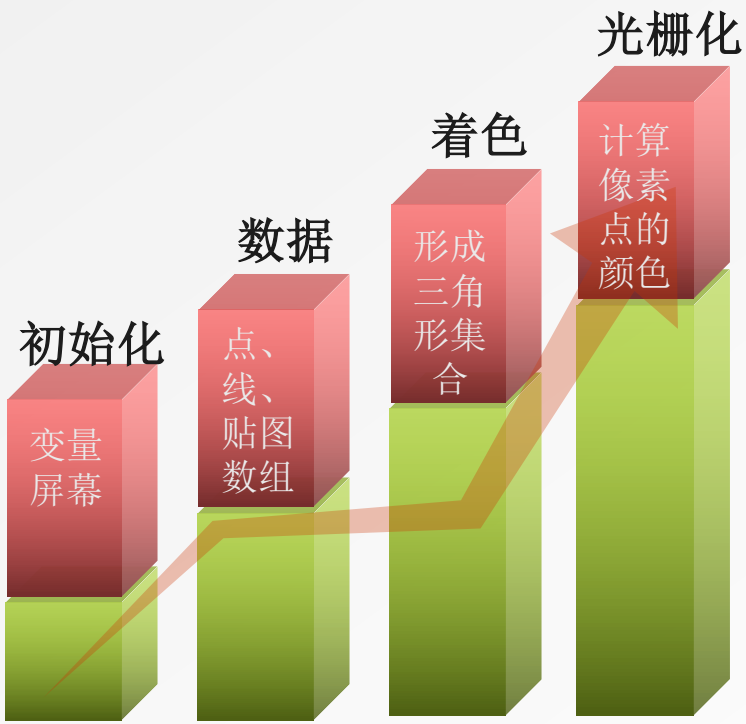
4 结论与展望

网页开发学习

- Javascript和html的语法
- Document object model
- 事件响应机制
- 数据读取
- UI框架

绘制流程理解

绘制流程指：WebGL指如何将模型数据绘制为图像的过程。



WebGL绘制流程

1. 状态机特点

不同属性的图形对应不同状态

基于状态分类绘制可提高效率

---

2. 着色器

分为顶点着色器和片元着色器

可根据需要自己设计不同的着色器

三维表现

三维表现主要包括几何、颜色、贴图、材质、光照等内容。

- 几何表达

Cube/Plane/Sphere以及面模型

--→THREE.Cube/Plane

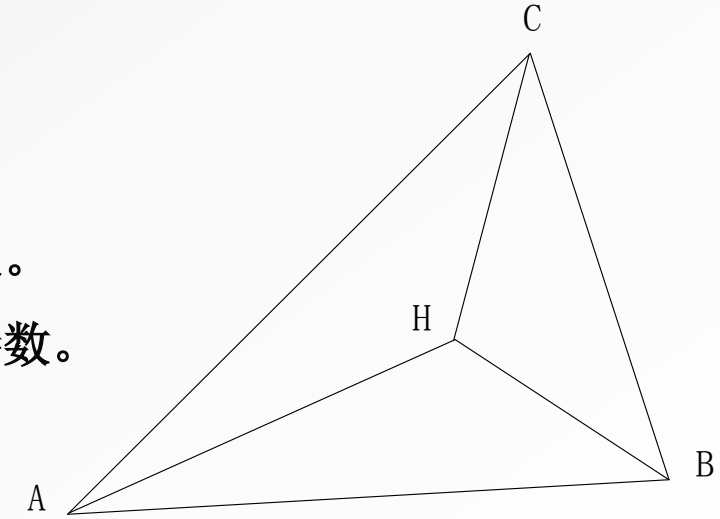
- 颜色、贴图及材质

基于顶点的颜色计算面内任意一点的颜色。

反映物体表面的透明度、折射、反射等参数。

面积插值方法。

--→THREE.PhongMaterial.



- 光照

物体能否形成阴影、能否接受阴影。

--→THREE.SpotLight.

三维图形编辑

向量参数

字符参数

数字参数

事件响应机制

属性绑定到相应的UI控件

只读属性的处理

实体选择

获取点击位置

形成视线向量

递归求交

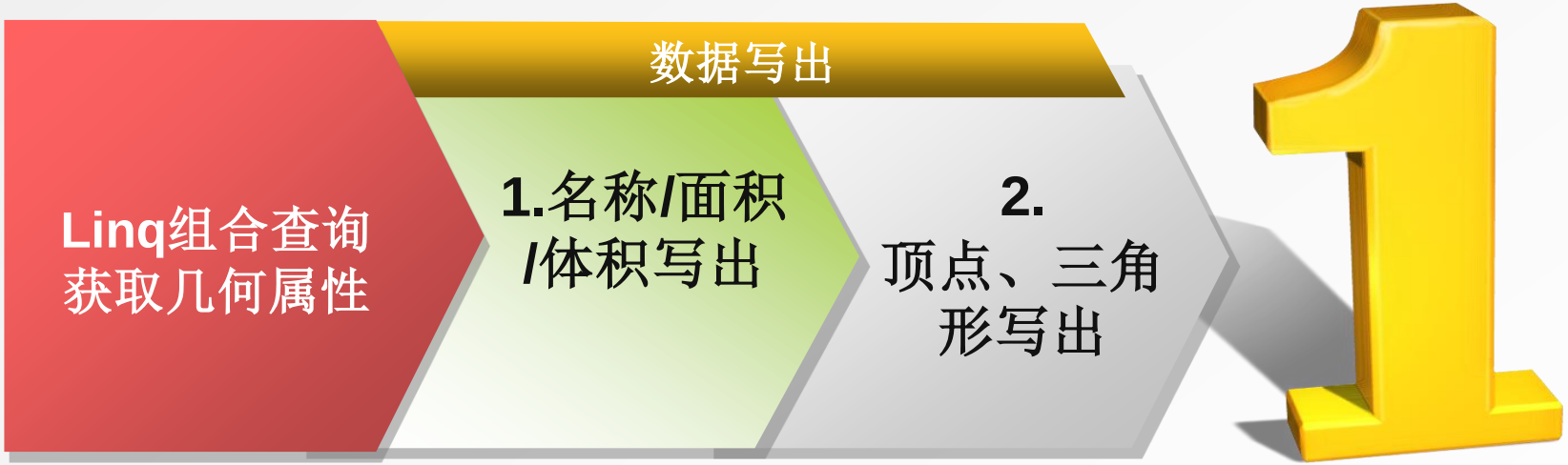
返回对象+刷新界面

数据交换

// face 10 0
0 0 0
4 0 0
0 5 0
……
f 0 1 2
……
// geo 6 0
0 0 3
4 0 0
0 0 0
……
f 0 1 2
……

数据交换

主要是4D中数据的导出及其在本研究中的应用




```
var geoProperties =
    from record in _gpProjectData.GraphicsData.BuildingEntityTable.ToRecordList()
    join geo in _gpProjectData.GraphicsData.GeoPropTable
        on record.GeometricPropertiesID equals geo.ID
    join mesh in _gpProjectData.GraphicsData.RepresentationMeshTable
        on geo.MeshID equals mesh.ID
    select new{
        Name = geo.Name, SurfaceArea = geo.SurfaceArea, Volume = geo.Volume, TriList
        = mesh.TriangleList,Offset = new Vector3(geo.OffsetX,geo.OffsetY,geo.OffsetZ)
    };
```

```
foreach (TriangleRecord tri in property.TriList) {
    Vector3 v;
    v = tri.FirstNode + property.Offset;writer.WriteLine(String.Format("{0} {1} {2}",v.X,v.Y,v.Z));
    v = tri.SecondNode + property.Offset;writer.WriteLine(String.Format("{0} {1} {2}",v.X,v.Y,v.Z));
    v = tri.ThirdNode + property.Offset;writer.WriteLine(String.Format("{0} {1} {2}",v.X,v.Y,v.Z));
}
int i = 0;
foreach (var tri in property.TriList) {
    writer.WriteLine(String.Format("f {0} {1} {2}",i++,i++,i++));
}
```

数据交换

数据解析器构造形式

```
new IO.XHR({url:xx,onSuccess:func,callBack:func});
```

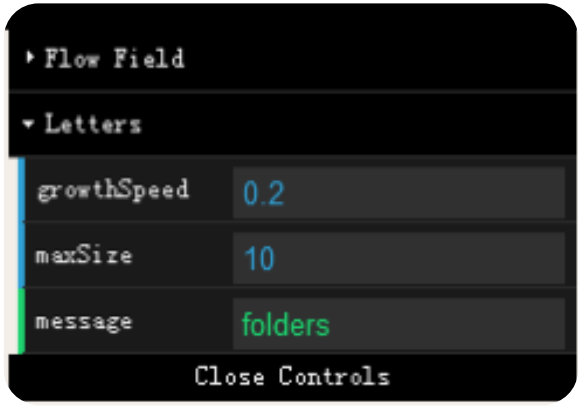
onSuccess后面为文件解析函数

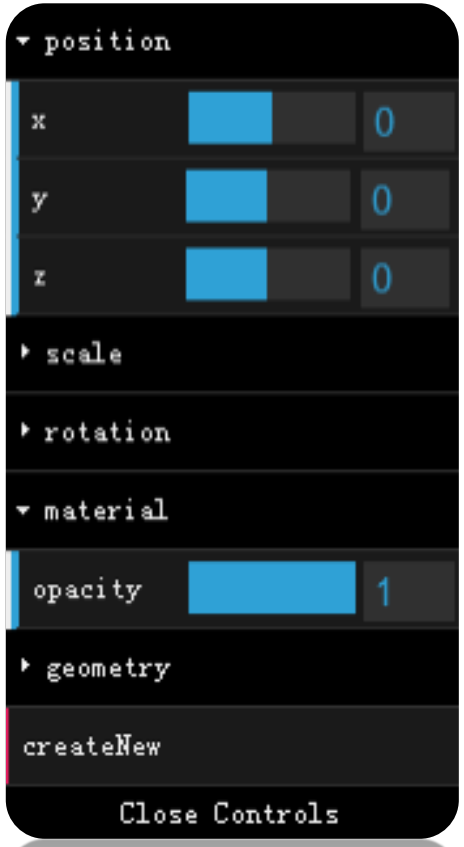
callBack后面为完成解析后的回调函数

```
new IO.XHR({
        url: 'prjgrpdt-simple.txt',
        onSuccess: function(data) {
            //process data
            callBack(model);
        });
```

UI设计

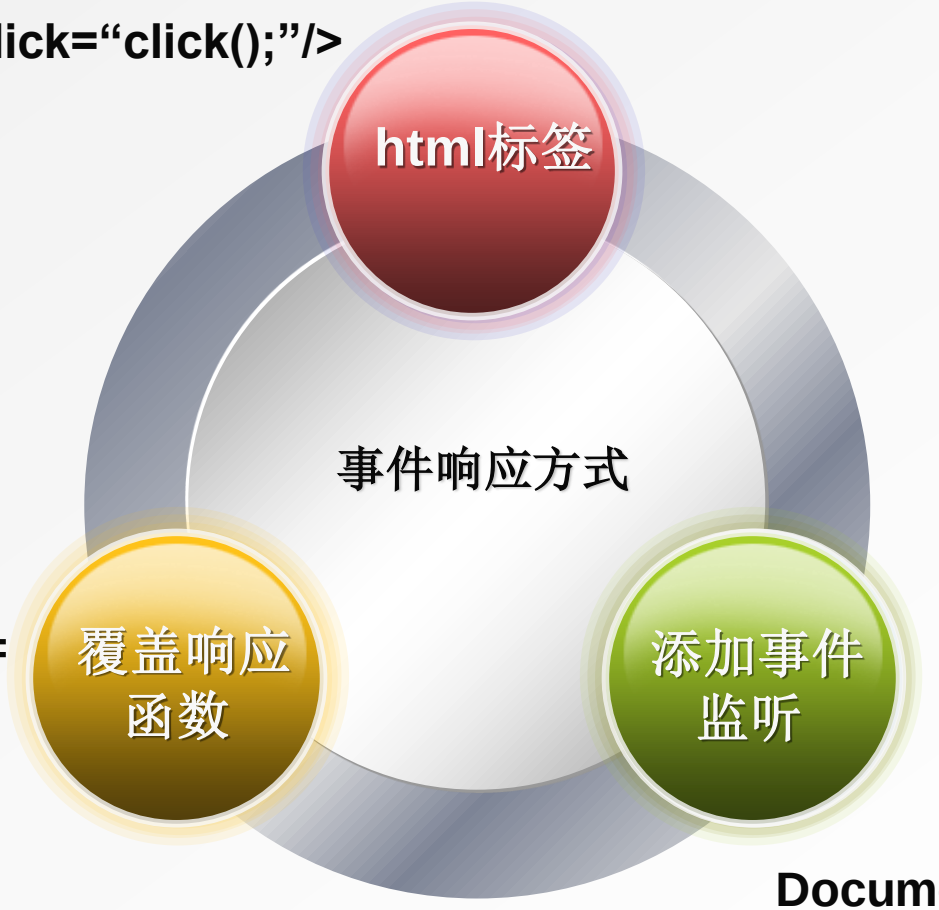
本研究UI交互框架采用了轻量级的dat.gui.





UI事件响应

```
<body onclick="click();"/>
```




```
window.onclick=
function(){
    //add code
};
```

```
Document
.addEventListener();
```

UI功能表

通过用户交互主要可实现以下三类响应。

属性编辑查看

实体位置、尺寸、透明等属性的查看编辑。

模型浏览

模型旋转、放大、平移的响应。

其他

全屏模式的进入与退出，系统当前状态的截屏响应。

三维显示效率优化

实体合并

- 合并相同材质的实体
- 不再支持单个实体的选择

几何体共用

- 形状相同或相似的实体共用几何体
- 实体形状改变复杂

状态分组

- 将绘制状态一致的实体分为一组
- 减少绘图状态修改此时

主要内容

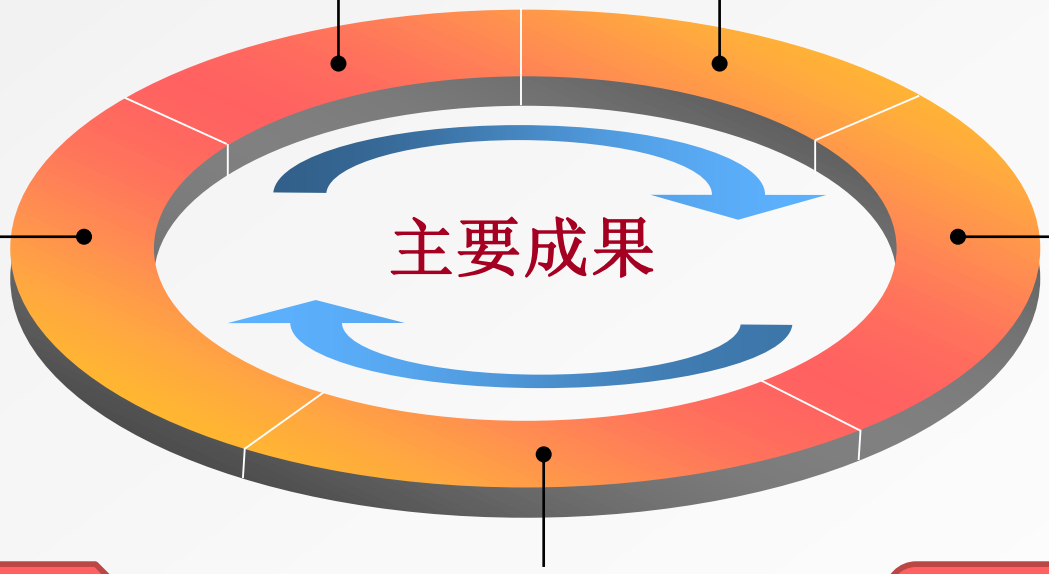
1 背景及现状

2 研究内容

3 主要工作

4 结论与展望

成果与不足



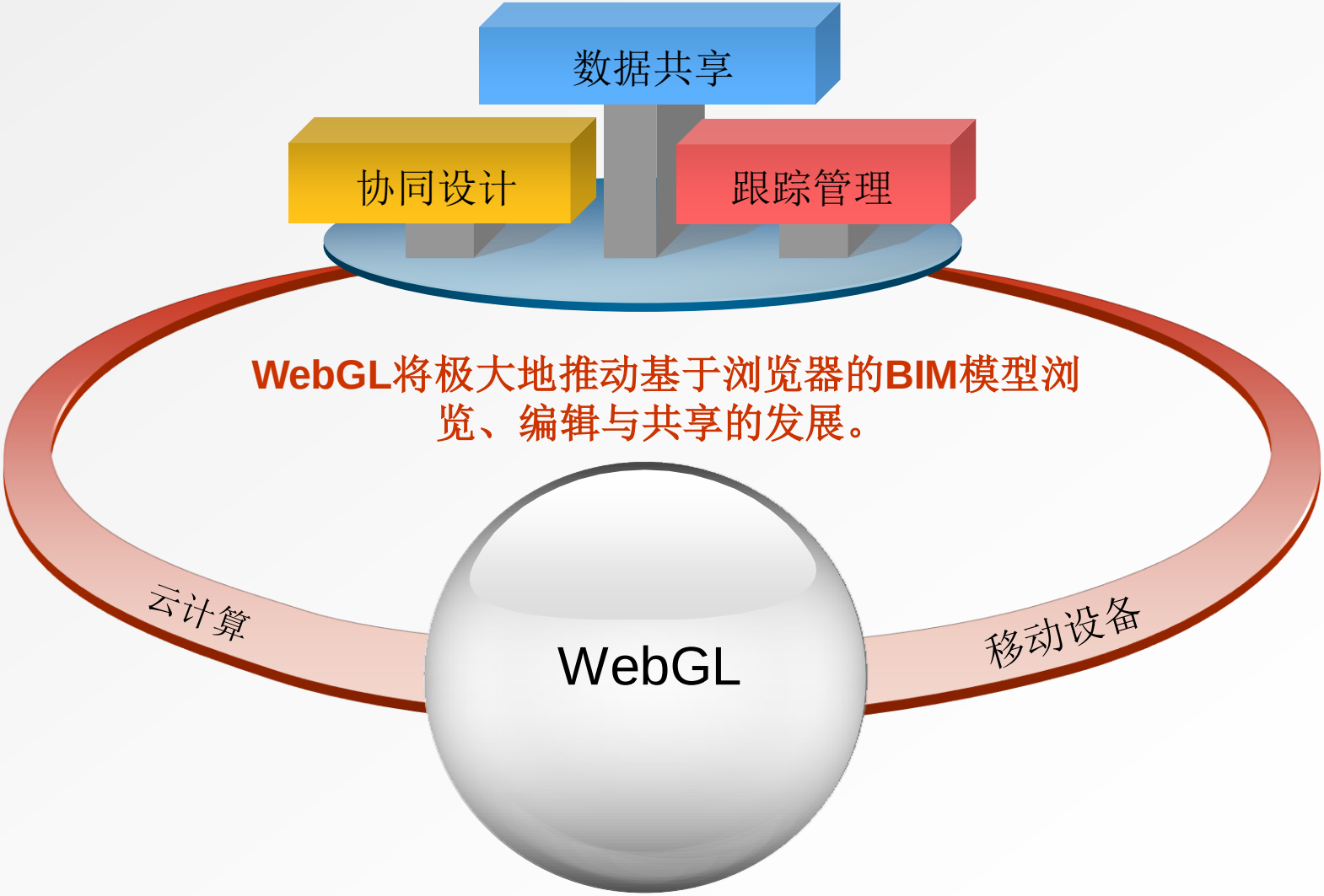
贴图光照待改善

显示效率是问题

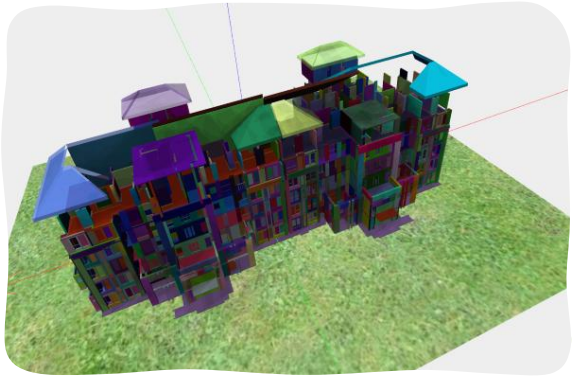
数据交换不完备

数据保存待开发

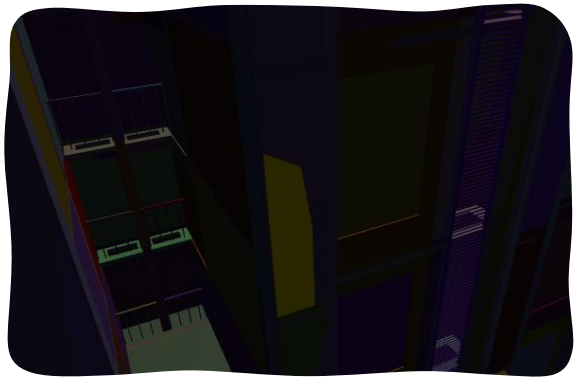
未来展望


数据共享
协同设计
跟踪管理
WebGL将极大地推动基于浏览器的BIM模型浏览、编辑与共享的发展。
云计算
WebGL
移动设备

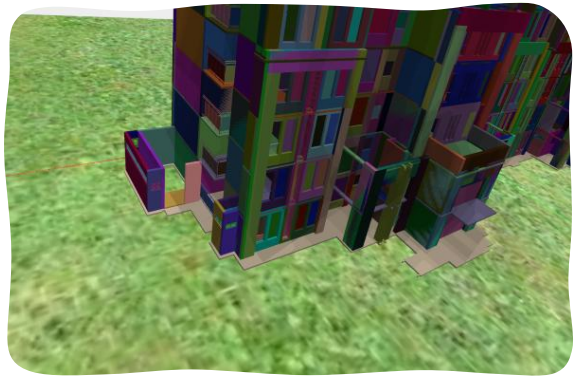
Flexible



Collaboration



Experience



Benefit

Thank You!

林佳瑞 2011310057

2012年4月11日