清華大學
Tsinghua University

基于OSG的建筑倒塌场景模拟

——土木工程CAE课程大作业报告

土博13班　曾翔

指导老师：胡振中

2014年04月17日

Contents

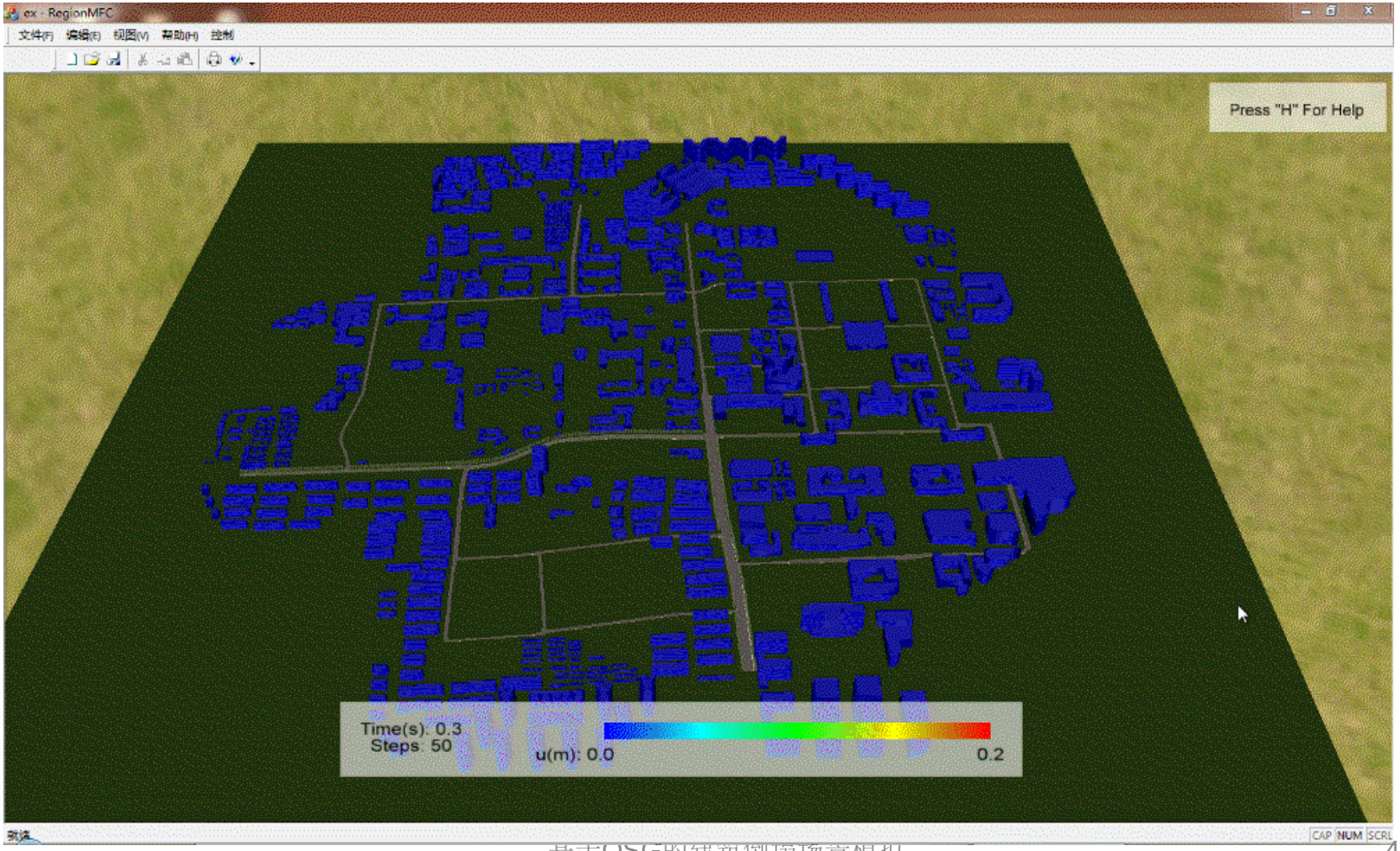
1.选题背景回顾

- “清华校园震害预测及展示”程序

1.选题背景回顾

清華大學 Tsinghua University

- “清华校园震害预测及展示”程序



1.选题背景回顾

清華大學 Tsinghua University

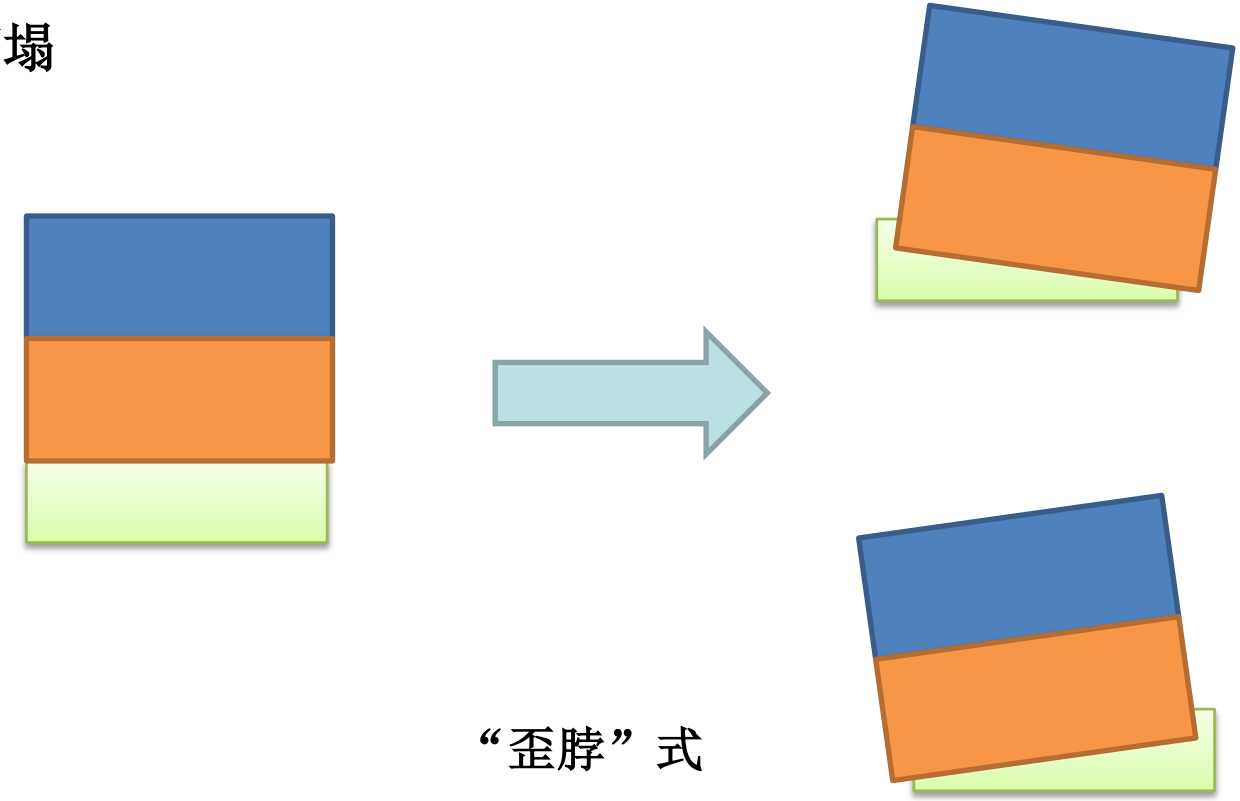
- **OSG：OpenSceneGraph，是一个基于OpenGL的开源三维图形引擎**

- 研究内容：
  - 一栋建筑
  - 给定各层位移时程和倒塌破坏限值
  - 利用**OSG**实现简单的倒塌场景模拟

1.选题背景回顾

- 模拟不同的倒塌场景

1） 左右倾塌

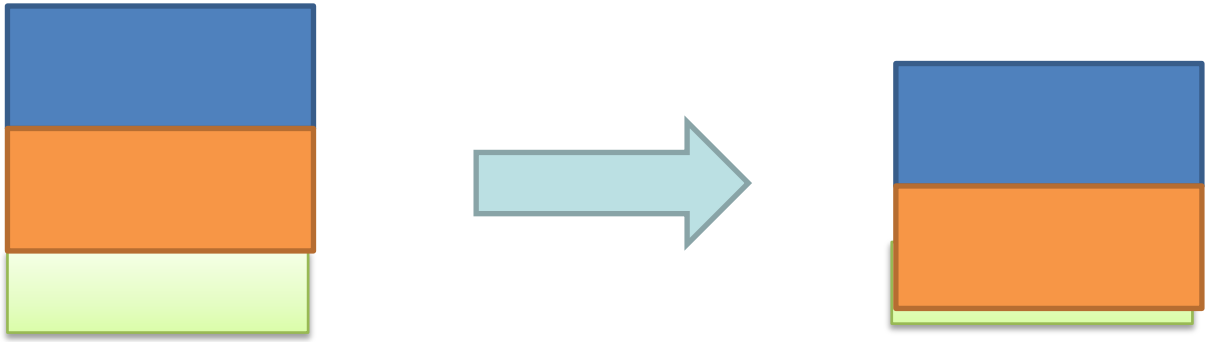


“歪脖”式

1.选题背景回顾

- 模拟不同的倒塌场景

2）底层压垮

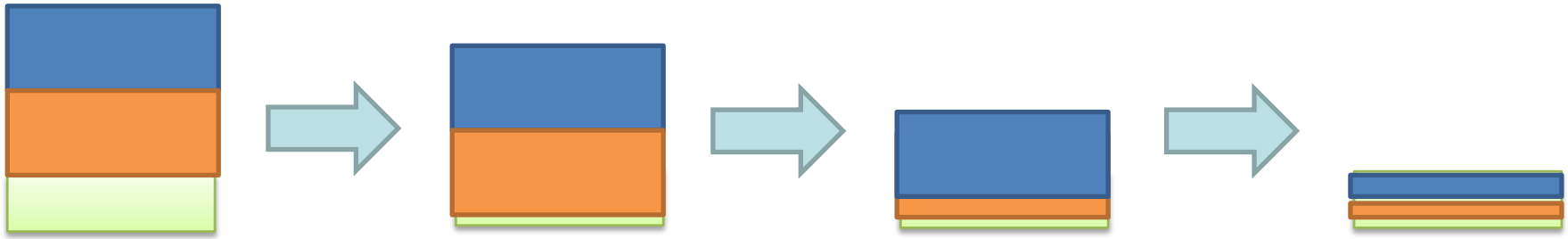


“给跪”式

1.选题背景回顾

- 模拟不同的倒塌场景

3） 连续倒塌

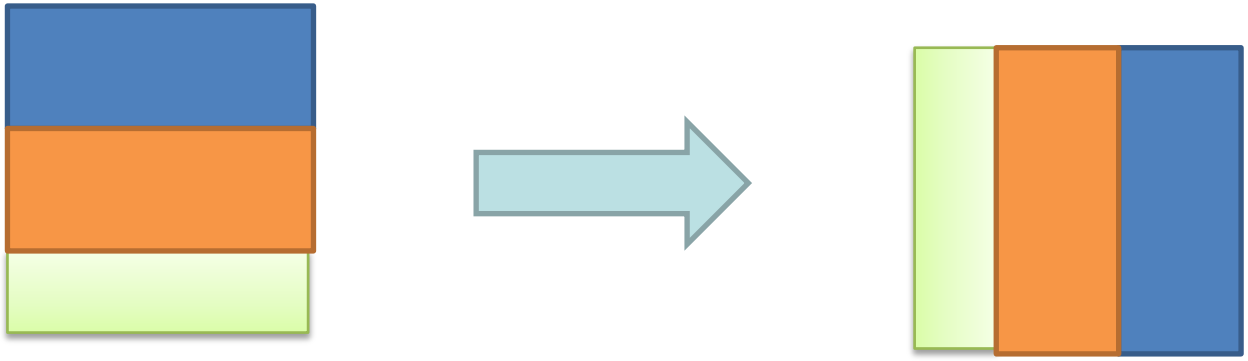


“前赴后继”式

1.选题背景回顾

- 模拟不同的倒塌场景

4） 整体倾覆



“脆脆”式

Contents

2.OSG知识学习

清華大學 Tsinghua University

OSG的安装、编译与环境配置

CMake

土木系大部分同学一辈子都不会碰的工具 = =

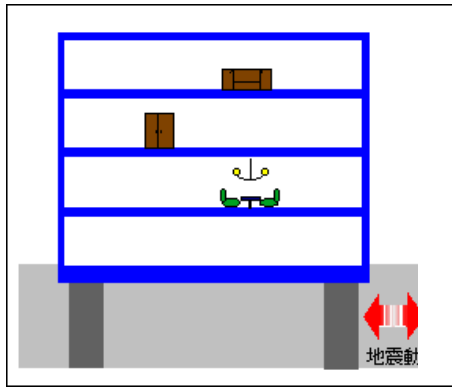
OSG程序的运行流程

```
//常规程序
Main()
{
语句1;
语句2;
…
语句n;

Return 0;
}
```

```
//OSG 程序
Main() {

语句1;
语句2;
…
语句n;

Viewer.run();

Return 0; }
```

模型定义

//循环执行每帧的操作，形成动画



2.OSG知识学习

清華大學 Tsinghua University

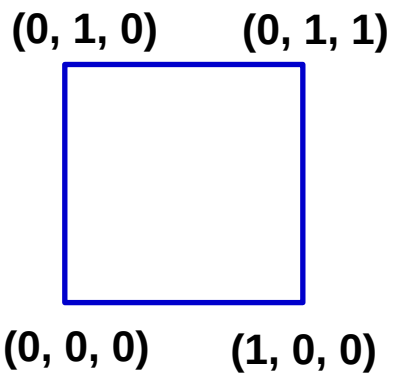
OSG的场景组织结构

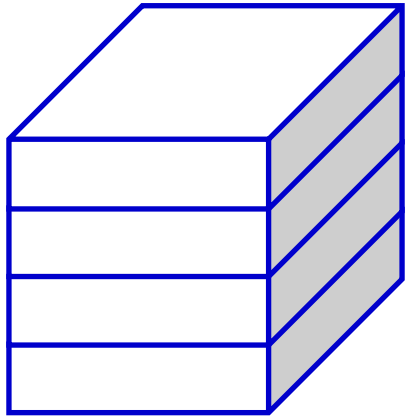
2.OSG知识学习

清華大學 Tsinghua University

OSG绘制图形

```
osg::ref_ptr<osg::Vec3Array> roofVertices = new osg::Vec3Array;
roofVertices->push_back(osg::Vec3(0, 0, 0));
roofVertices->push_back(osg::Vec3(0, 1, 0));
roofVertices->push_back(osg::Vec3(1, 1, 0));
roofVertices->push_back(osg::Vec3(1, 0, 0));

osg::ref_ptr<osg::Geometry> houseRoof = new osg::Geometry;
houseRoof->setVertexArray(roofVertices);
houseRoof->addPrimitiveSet(new osg::DrawArrays(osg::DrawArrays::QUADS,0,4));
```



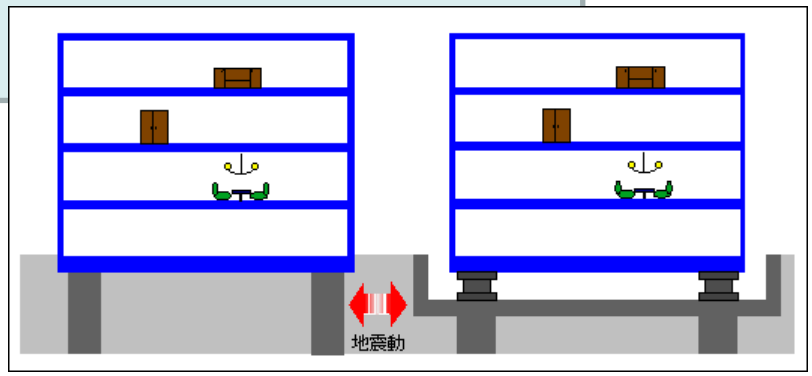


2.OSG知识学习

OSG的更新回调——osg::Drawable::UpdateCallback

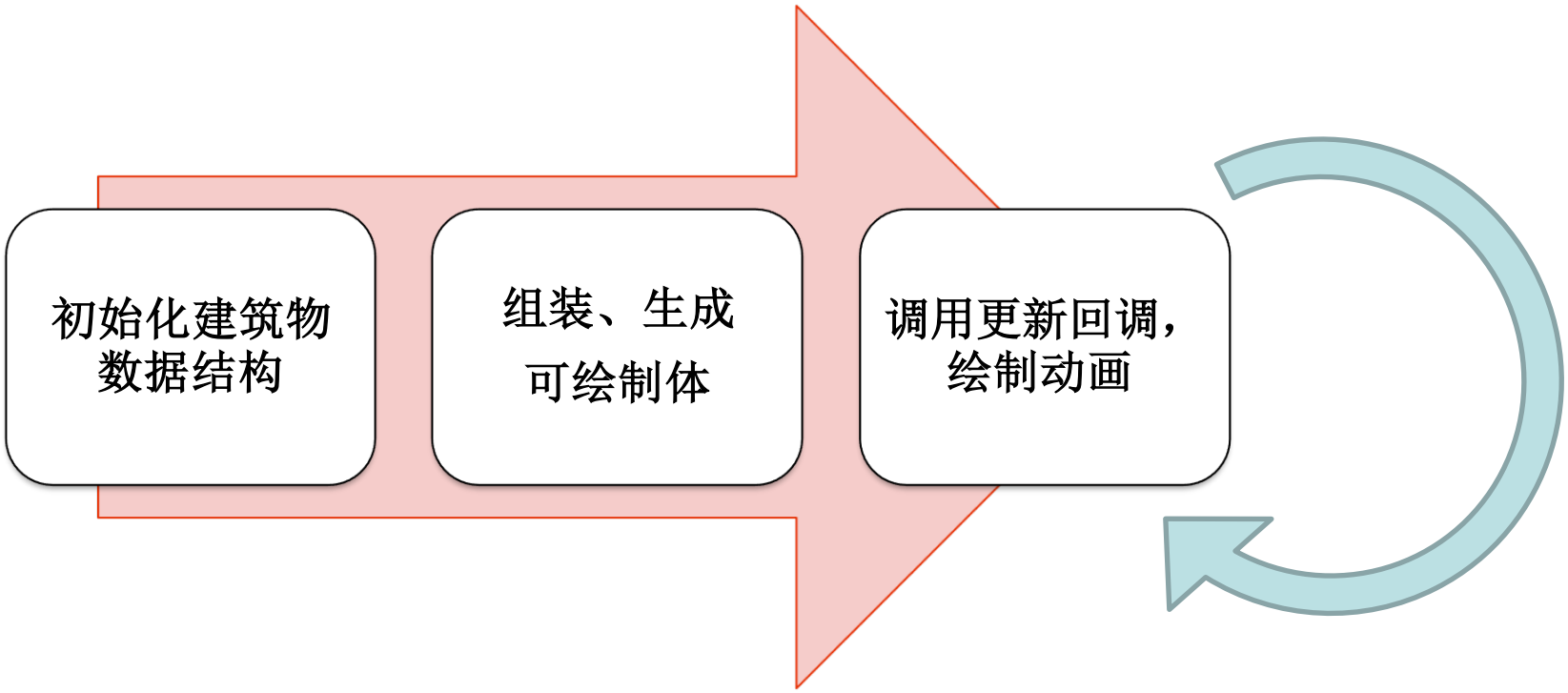
- 每一帧都会自动调用UpdateCallback::update()方法；
- 重载update()方法，实现节点沿x轴的平移：

```
void update(osg::NodeVisitor *nv , osg::Drawable * drawable)
{
   osg::Geometry *geom = dynamic_cast<osg::Geometry*>(drawable);
   osg::Vec3Array *vertices = geom->getVertexArray();
   vertices[0].x()+=1;
   vertices->dirty();
}
```



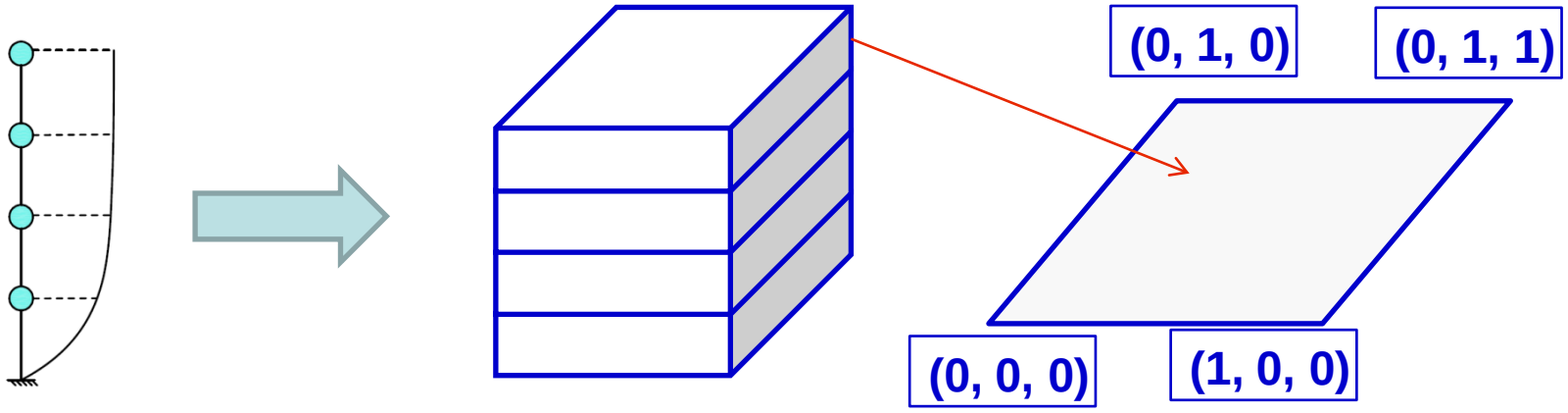
Contents

3.程序框架



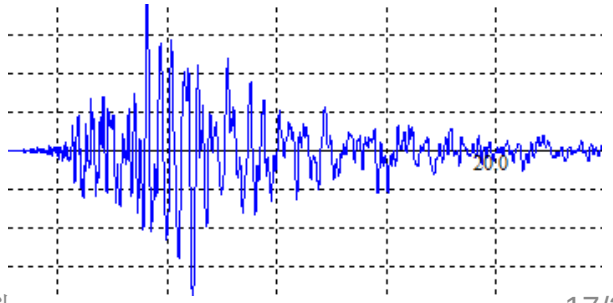
初始化数据结构

1. 设置建筑层数、位置、面积等参数，生成关键点坐标



n 个代表点　　4n 个关键点

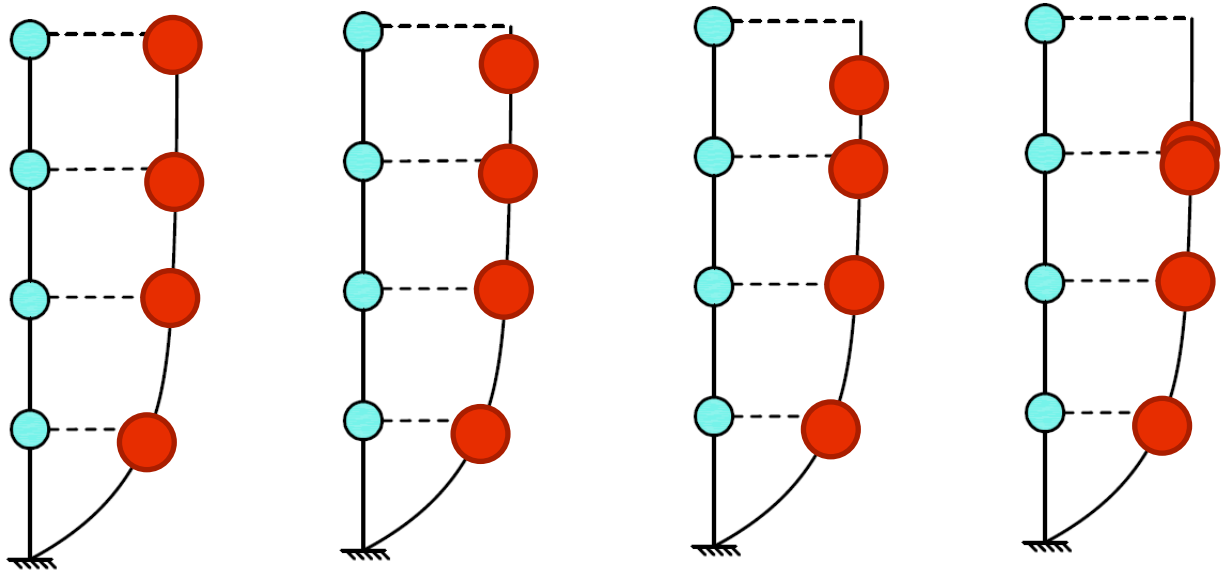
2. 读取各层代表点的NTimePoint个位移时程点
   - 生成disp[n][NTimePoint]数组



初始化数据结构

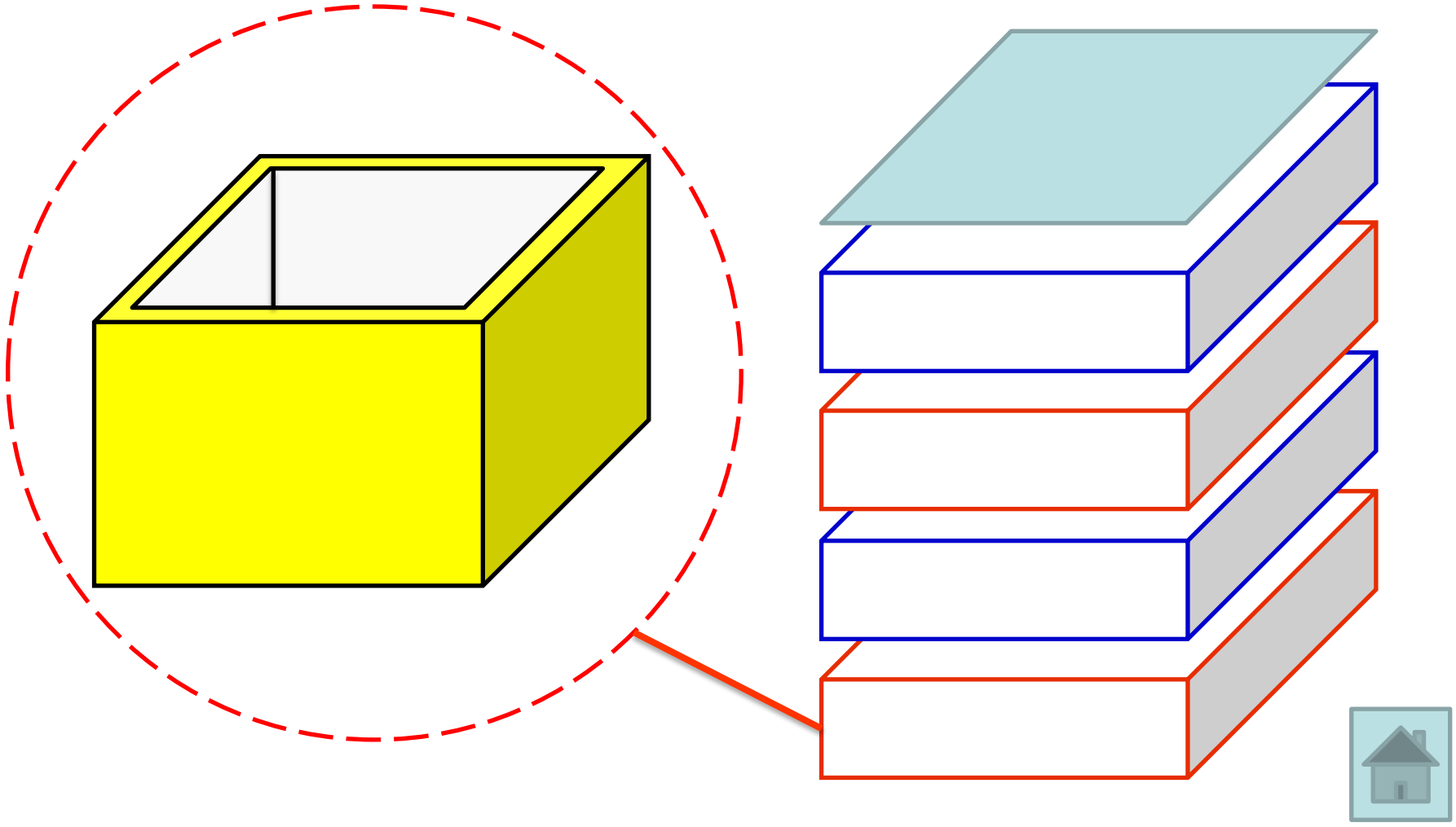
3. 判断是否倒塌

- 层间位移角超限则倒塌
- 判断倒塌模式，生成代表点的倒塌轨迹，用数组clps[n][clpsFrame]保存
- 考虑加速度，dZ= - gt
- 倒塌在clpsFrame 步之内完成



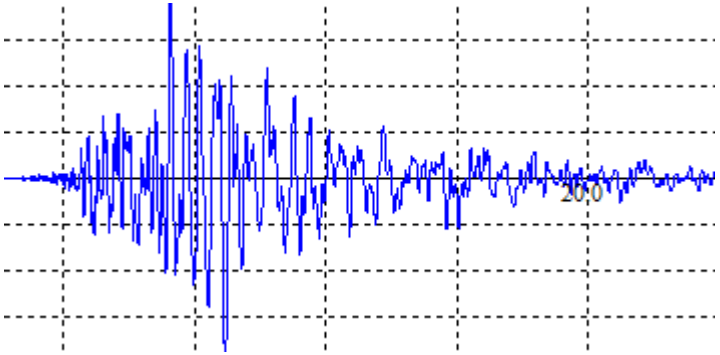
例：顶层倒塌时，代表点的轨迹

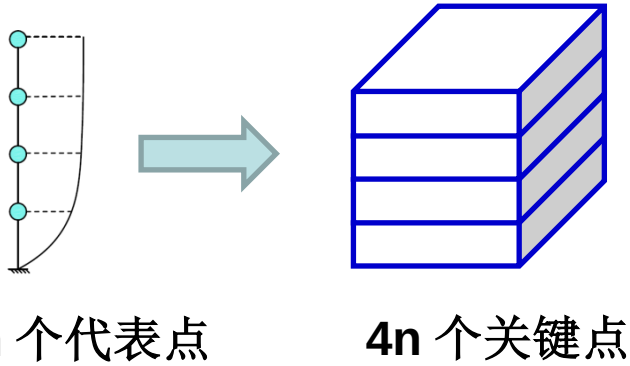
生成可绘制体



调用更新回调

1. 倒塌前：每帧读取代表点的位移时程，进而更新关键点的点坐标



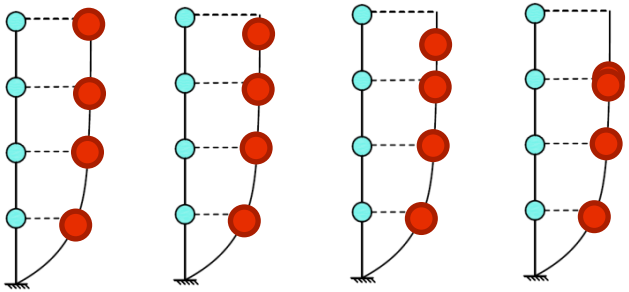


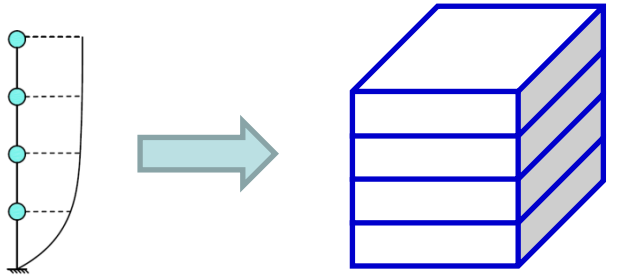
n 个代表点　　4n 个关键点

- disp[n][NTimePoint]数组

调用更新回调

2. 判断是否倒塌。
   - 如果建筑在第 k 帧倒塌，则从第k帧开始，读取倒塌轨迹数组
   - 更新关键点的坐标

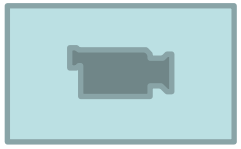




n 个代表点　　4n 个关键点

- clps[n][clpsFrame]数组

Contents



谢谢！