

您的潜力，我们的动力

Microsoft[®]
微软(中国)有限公司

Direct3D Mobile初体验

马宁

Windows Mobile MVP



您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

本次课程内容包括

- Direct3D Mobile简介
- Direct3D基本知识
- Direct3D Mobile与Direct3D的比较
- 第一个Direct3D Mobile程序
- Direct3D Mobile的对象

什么是Direct3D Mobile?

- Direct3D Mobile为WinCE平台提供3D图形编程的API
- Direct3D Mobile以DirectX 8为基础，结合了DirectX 9的一些功能。
- Direct3D Mobile主要用于游戏、导航系统、CAD软件，或下一代的用户界面
- 支持alpha blending、高质量位图拉伸、3D转换

Direct3D Mobile支持的功能

- 支持 3-D z-buffers.
- 支持深度缓存 (使用 z-buffers or w-buffers).
- 多样性的 light sources 和 types.
- 全部 material 和 texture支持, 包括 mipmapping.
- 准确的 software emulation drivers.
- Transformation 和 clipping.
- Hardware无关性.
- 在全屏模式下支持多个缓冲区的page flipping.
- 支持 rendering to windowed 或者 full-screen applications.
- 访问 image-stretching hardware.
- 独占式的 hardware access.

Direct3D Mobile vs. Direct3D

- Direct3D Mobile的功能是Direct3D 8的子集
- Direct3D Mobile中所有编程对象的名字都与Direct3D不同
- Direct3D Mobile允许使用fixed-point graphics管道
- Direct3D Mobile 只支持 HAL device
- 在同一个时间只支持一个单独设备被初始化
- Direct3D Mobile 只支持单独的 current vertex 和 index stream

向Direct3D Mobile迁移

您的潜力，我们的动力

Microsoft
微软(中国)有限公司

- Direct3D Mobile中所有编程对象的名字需要修改
- Windows Mobile用户界面的尺寸和分辨率需要修改
- Direct3D Mobile的功能是Direct3D 8的子集
- 区分Windows Mobile横屏和竖屏的显示方式
- 考虑设备运算能力和存储空间的限制
- 考虑Windows Mobile设备输入方式与PC的不同

Managed vs. Native

- Managed API
 - Microsoft.WindowsMobile.DirectX.Direct3D namespace
 - 使用C#或者VB.NET编程
- Native API
 - IDirect3DMobile COM接口
 - 使用C++编程

引用**Direct3D Mobile**

- Managed API
 - 引用Microsoft.WindowsMobile.DirectX
 - 引用Microsoft.WindowsMobile.DirectX.Direct3D
- Native API
 - D3dm.dll : Direct3D Mobile的动态连接库
 - D3dm.lib : Direct3D Mobile的入口函数库
 - D3dm.h : 包含Direct3D Mobile的COM接口声明
 - D3dmtypes.h : 包含Direct3D Mobile的结构和枚举变量的声明
 - D3dmcaps.h : 包含capability bits 和 profiles声明

您的潜力，我们的动力

Microsoft[®]
微软(中国)有限公司

DEMO

第一个Direct3D Mobile程序

Device对象

您的潜力，我们的动力

Microsoft
微软(中国)有限公司

- Device是Direct3D Mobile的显示控件
- Device对象允许Direct3D Mobile工作在抽象的硬件设备上
- 包括transformation模块、lighting模块和rasterizing模块
- Device设备包括默认系统设备和当前注册系统设备
- 每个Device均有自己的序号，系统序号为0

设置Direct3D Mobile环境

您的潜力，我们的动力

Microsoft
微软(中国)有限公司

● 创建Device

```
// Now let's setup our D3D parameters
PresentParameters presentParams = new PresentParameters();

// Causes the display to appear in a window rather than
// full screen
presentParams.Windowed = true;

// When a new frame is swapped to the front buffer
// the old frame will be discarded
presentParams.SwapEffect = SwapEffect.Discard;

// 0 gives us the first monitor adapter on the system
// for md3dm you must use the default device type
device = new Device(0, DeviceType.Default, this,
    CreateFlags.None, presentParams);
```

Device的构造函数

```
public Device( int adapter,  
              DeviceType deviceType,  
              Control renderWindow,  
              CreateFlags behaviorFlags,  
              PresentParameters presentationParameters );
```

- adapter:物理显示设备序号，默认显卡为0
- deviceType: 设备类型
 - DeviceType.Hardware, 表示将创建一个硬件设备
- renderWindow: 与设备绑定的窗口
- behaviorFlags: 指定设备创建后的操作
- presentationParameters: 把数据呈现到显示器的方式
 - SwapEffect.Flip, 运行时会创建额外的后备缓冲
 - SwaoEffect.Discard, 如果缓冲没有准备好被显示，则会丢弃缓冲中的内容

设置Direct3D Mobile环境

您的潜力，我们的动力

Microsoft
微软(中国)有限公司

● 绘制

```
//Clear the backbuffer to a blue color
device.Clear(ClearFlags.Target | ClearFlags.ZBuffer,
    System.Drawing.Color.Blue, 1.0f, 0);
//Begin the scene
device.BeginScene();
// Setup the world, view, and projection matrices
SetupMatrices();

//End the scene
device.EndScene();
device.Present();
```


您的潜力，我们的动力

Microsoft
微软(中国)有限公司

DEMO

旋转的三角形

使用**VertexBuffer**生成三角形
设置世界坐标系

Vertex Buffers

- Vertex(顶点) 是Direct3D的最小信息单位
- Vertex用来保存位置、状态和颜色等信息
- Vertex Buffer是顶点的集合用来存储顶点信息
- Direct3D名称空间里叫做CustomVertex的类可以用来储存大多数Direct3D中用到的“顶点格式”数据结构 (vertex format)
- 调用Device.DrawPrimitives方法来绘制Vertex Buffer
- CustomVertex.PositionColored可以保存三维位置信息和颜色信息
- CustomVertex.PositionTextured保存三维位置信息材质信息

创建Vertex Buffer

```
VertexBuffer vb = new VertexBuffer(  
    typeof(CustomVertex.PositionColored), 3, dev, 0,  
    CustomVertex.PositionColored.Format, vertexBufferPool);  
  
CustomVertex.PositionColored[] verts =  
    (CustomVertex.PositionColored[])vb.Lock(0, 0);  
verts[0].X = -1.0f;  
verts[0].Y = -1.0f;  
verts[0].Z = 0.0f;  
verts[0].Color = System.Drawing.Color.DarkGoldenrod.ToArgb();  
  
verts[1].X = 1.0f;  
verts[1].Y = -1.0f;  
verts[1].Z = 0.0f;  
verts[1].Color = System.Drawing.Color.MediumOrchid.ToArgb();  
  
vb.Unlock();
```

设置坐标系

- Device.Transform.World
 - 设置世界坐标系
- Device.Transform.View
 - 设置view matrix, 定义一个eye point
- Device.Transform.Projection
 - 设置projection matrix, 决定从3D转换为2D视图的几何模型

```
device.Transform.View = Matrix.LookAtLH(  
    new Vector3(0.0f, 3.0f, -5.0f),  
    new Vector3(0.0f, 0.0f, 0.0f),  
    new Vector3(0.0f, 1.0f, 0.0f));
```

您的潜力，我们的动力

Microsoft[®]
微软(中国)有限公司

DEMO

显示图象的三角形

使用**Texture**（材质）

Texture

- Texture是用于3D对象表面的位图信息集合
- 任何Bitmap都可以成为Direct3D的材质
- 使用TextureLoader.FromFile从文件加载材质（必须是bmp文件）
- 使用Device.SetTexture为设备设置材质
- VertexBuffer中的顶点信息类型必须为CustomVertex.PositionTextured

您的潜力，我们的动力

Microsoft
微软(中国)有限公司

DEMO

旋转的老虎

使用**Mesh**
设置材质、灯光

Mesh

您的潜力，我们的动力

Microsoft
微软(中国)有限公司

- 使用3D建模软件创建的复杂几何图形
- D3D Mobile支持的文件格式扩展名md3dm
- MD3DM文件可以由.X文件转换生成
- 创建mesh的方法
 - 从文件中加载
`Microsoft.Samples.MD3DM.MeshLoader.LoadMesh`
 - 根据存在的Mesh对象创建
 - 根据指定形状函数创建Mesh对象
`Mesh.Box` `Mesh.Cylinder`
 - 通过Mesh构造函数创建
- 使用Mesh.DrawSubset来绘制

Materials

您的潜力，我们的动力

Microsoft
微软(中国)有限公司

- **Material**属性描述了物体表面漫反射、镜面反射和反射环境光的特性
- **Diffuse** 和 **Ambient**属性决定了材质在一个场景中如何表现环境色和反射光
- 镜面反射时，**Specular**决定反射的颜色，**Power**决定发光的强度

```
Material boxMaterial = new Material();  
boxMaterial.Ambient = Color.White;  
boxMaterial.Diffuse = Color.White;  
device.Material = boxMaterial;
```

Lighting

- 场景中的灯光是默认打开的
- D3D Mobile的灯光由RGB三个颜色组成
- 在场景中灯光不是必须的，但是Material是必须设置的
- 灯光模式：环境光和直射光
- 环境光的光源和方向都是不确定的
- 直射光由一个光源产生，有确定的方向

```
device.Lights[0].Type = LightType.Directional;  
device.Lights[0].Diffuse = System.Drawing.Color.DarkTurquoise;  
device.Lights[0].Direction = new Vector3(  
    (float)Math.Cos(Environment.TickCount / 250.0f),  
    1.0f,  
    (float)Math.Sin(Environment.TickCount / 250.0f));  
device.Lights[0].Enabled = true; // Turn it on
```


Direct3D Mobile开发资源


- **Visual Studio 2005文档**
[http://msdn2.microsoft.com/en-us/library/ms229664\(en-US,VS.80\).aspx](http://msdn2.microsoft.com/en-us/library/ms229664(en-US,VS.80).aspx)
- **MSDN Direct3D Mobile Samples**
- **Mobile DirectX with the Compact Framework**
<http://www.brains-n-brawn.com/default.aspx?vDir=cfmdx>
- **Direct3D Tutorial Wiki**
<http://pluralsight.com/wiki/default.aspx/Craig.DirectX.Direct3DTutorialIndex>
- **GameRes游戏开发网**

获取更多MSDN资源


- **MSDN中文网站**
<http://www.microsoft.com/china/msdn>
- **MSDN中文网络广播**
<http://www.msdnwebcast.com.cn>
- **MSDN Flash**
<http://www.microsoft.com/china/newsletter/case/msdn.aspx>
- **MSDN开发中心**
<http://www.microsoft.com/china/msdn/DeveloperCenter/default.msp>

Question & Answer

如需提出问题，请单击“提问”按钮并在随后显示的浮动面板中输入问题内容。一旦完成问题输入后，请单击“提问”按钮。

 **问题和解答 (无问题)** ▲ ×

在此会议中尚未解答任何问题。

要向演示者提问，请在此处键入问 

提问(A)

删除(D)

问题管理器(Q)

您的潜力，我们的动力

Microsoft®
微软(中国)有限公司

Microsoft®

msdn


MSDN Webcasts