

Technical Manual

Overview:

The Application aims to simulate a 3-D universe in a Windows Form.

Libraries Used:

1. DirectX - Microsoft.DirectX.dll
2. Direct3D - Microsoft.DirectX.Direct3D.dll
3. DirectX for Visual Basic – Interop.DxVBLibA.dll

API's Used:

1. Win-32

Classes:

1. Solar
2. D3DApp – Base App class
3. D3DWindow – Base window class
4. Utility – Enquires about availability of Win-32 API

Solar class:

This is the main class of the Application.

It inherits the App class, has all the local members, local methods, event handlers and the application entry-point.

Local Members:

1. Structures of moons and planets – these contain all the properties of individual moons/planets.
2. Vertex – Specifies properties of a custom vertex
3. Constants – Specifies various constants that are used throughout the Application
4. Planets – This contains all the specific data related to the heavenly bodies.

Shape Definitions:

1. Sphere – This is the method to create a sphere mesh using DirectX library, to render all the planets, Sun and moon.
2. Torus – Creates Torus mesh for rendering orbits of the planets.

Constructor:

1. Creates new instance of the solar class.

Local Methods:

1. Paint Handler – Manages background image when form is loaded.
2. Go to Planet – Orients the camera to focus on a specific planet.
3. Stay at Planet – Orients the camera to stay with the planet while it revolves.
4. Resource – This manages access of Resources by their name.
5. Lights – Creates the light objects and initializes them to enable viewing through camera.
6. Color – Make a colored custom vertex
7. Vector – Make a custom 3-D vector.

8. Camera – Initialize the camera and start view .It updates the camera each rendering cycle and has methods for zoom in/out.
9. Devices – Set lighting of the device used to render objects.
10. Labels – Initializes, handles hide/show of labels based on which category of view is opted and fills in all information specific to the planet.
11. Data – Initializes all meshes to draw planets, moons and orbits.
12. Render – The periodically called function at each frame which calls the Draw function.
13. Draw - Has methods to draw planets, orbits, rings and loads textures.
14. Texture – Creates a clone of the original mesh and covers it with the texture image.

Event Handlers / Overrides:

1. Device Reset – Initializes viewport if the device is reset.
2. Mouse clicks – Handle zoom-in and zoom-out events.
3. Timer Events – Timer is enabled to handle periodic events .
4. Radio Events – Switch between modes of view – universe / planet.
5. Checkbox Event – Toggle viewing of orbits.

Application Entry Point:

1. Initialize Component – Initializes all the labels, radio-buttons and checkbox.
2. Main – This is the entry-point of the Application.
3. Friend Events – Handle control events.

D3D App class:

This is a base class which handles resizing, key-down, sleep.

This class is implemented by the main Solar class.

Constructor:

1. Sets window title.
2. Executes command whether to turn in Full screen or Windowed mode.
3. Creates the device for rendering the environment based upon the parameters

Arguments:

0: adapter

D3D.DeviceType.Hardware: Device Type

Me: Render Window

D3D.CreateFlags.SoftwareVertexProcessing: behavior flags

_present Parameters: Present Parameters

4. D3DApp Constructor - Sets up the window and creates the D3D device

Name: Window Title / App Name

Size: Size of window client area

Full Screen: Full screen or Windowed

5. Background color
6. Center this window

Public Interface:

Gets and Sets the size (width and height) of the window.

Event handlers/Overrides:

1. Cancel Resize - This is an event handler used to turn off the automatic resize handling that Managed D3D does behind the scenes

2. On Device Lost - The device has been lost. Some resources will need to be freed to allow the device to be reset.
3. On Device Reset - The device has been successfully reset. Settings lost by the Reset should be set here and any resources that were freed in On Device Lost should be reallocated.
4. On Key Down - Exit when user hits Escape, F/W - Swap between full-screen and windowed mode.
5. On Application Idle - This function is to be bound to the application idle event. As long as there are no messages waiting in the queue it will call Heartbeat as fast as it can.

Local methods:

1. Go Full screen - Modifies the window to be in full-screen mode. If the window is already visible, it hides it, makes all changes and then shows it again.
2. Go Windowed - Modifies the window to be in windowed mode. If the window is already visible, it hides it, makes all changes and then shows it again.
3. _Init Present Params - Initializes our Present Parameters class.
4. Heartbeat - This is called every frame to handle general maintenance render our scene. If this form is not active, we sleep so we don't burn CPU cycles uselessly. Also handles Device Resets.
5. Render - Clears the background, then begins and ends the scene. If additional rendering needed to be done, it would be done here.

Initialize:

Initializes the App window with given Parameters.

D3D Window class:

A window – maintains properties of the window.

Local members:

Contains local variables associated with the window.

Public interface:

1. Client Size – Gets and Sets the size of the client window.
2. Full Screen – Executes whether to enter Full Screen or Windowed mode.

Local methods:

1. Go Full screen - Modifies the window to be in full-screen mode. If the window is already visible, it hides it, makes all changes and the shows it again.
2. Go Windowed - Modifies the window to be in windowed mode. If the window is already visible, it hides it, makes all changes and the shows it again.

Event handlers/Overrides:

Exit when user hits Escape, F/W - Swap between full-screen and windowed mode

Constructor:

1. This makes the window auto-scale based on font size.
2. Sets the background color.
3. Centers the window.
4. Use Windowed mode as default.

On Load:

1. Initialize Component - Initializes the components of the D3D window.

Utility class:

It manages certain user-interface functions.

1. Ask Full Screen - Queries user to find out preference for Full-Screen vs. Windowed.
2. Find16BitMode - Tests for 16-bit buffer formats for compatibility with the current display adapter in full-screen.
3. Loads the first icon from the resources.