Frame Listeners
OIS

381

Frame listener

Ogre's main loop - any graphics engine's main loop:

1. The Root object calls the frameStarted method on all registered *FrameListeners*.
2. The Root object renders one frame.
3. The Root object calls the frameEnded method on all registered FrameListeners.

A 'cameraMovement' Frame listener would listen for wasd and move camera
A 'control' Frame listener would listen for arrowkeys and move selected entity

```
import ogre.renderer.OGRE as ogre
import ogre.io.OIS as OIS
import SampleFramework as sf

class TutorialFrameListener(sf.FrameListener):
    def __init__(self, renderWindow, camera, sceneManager):
        sf.FrameListener.__init__(self, renderWindow, camera)

    def frameStarted(self, frameEvent):
        return sf.FrameListener.frameStarted(self, frameEvent)

class TutorialApplication(sf.Application):
    def _createScene(self):
        pass

    def _createCamera(self):
        pass

    def _createFrameListener(self):
        pass

if __name__ == '__main__':
    try:
        ta = TutorialApplication()
        ta.go()
    except ogre.OgreException, e:
        print e
```

TutorialFrameListener

Want this to control camera with wasd

```
def _createFrameListener(self):
    self.frameListener = TutorialFrameListener(self.renderWindow, self.camera,
                                                self.sceneManager)
    self.root.addFrameListener(self.frameListener)
```

Camera control

```
RootSceneNode
    Child: cameraYawNode
        Child: cameraPitchNode
            Child: camera
```

- CameraYawNode
  - Camera Movement - Translate
  - Camera Rotation around y-axis - YAW
- CameraPitchNode
  - Camera Rotation around x-axis - Pitch
- No rolling - no rotation around z-axis for now!

Listening for mice/keypress's

```
If key was down:
    do_action
```

"was" ?

If we see that a key is down, we can act on this information, but what happens the next frame? Do we see that the same key is down and do the same thing again? In some cases (like movement with the wasdqe keys) this is what we want to do. However, lets say we want the "T" key to toggle between a light being on or off. The first frame the T key is down, the light gets toggled, the next frame the T key is still down, so it's toggled again...and again and again until the key is released. We have to keep track of the key's state between frames to avoid this problem.

Mice toggling

```
if currMouse.buttonDown(OIS.MB_Left) and not self.mouseDown:
        light = self.sceneManager.getLight('Light1')
        light.visible = not light.visible

self.mouseDown = currMouse.buttonDown(OIS.MB_Left)
```

Keys

```
 # Update the toggle timer.
if self.toggle >= 0:
            self.toggle -= frameEvent.timeSinceLastFrame

 # Swap the camera's viewpoint with the keys 1 or 2.
 if self.toggle < 0 and self.Keyboard.isKeyDown(OIS.KC_1):
            # Update the toggle timer.
            self.toggle = 0.1
            # Attach the camera to PitchNode1.
            doStuff()
```

Quit

add a FrameListener

```
def frameStarted(self, frameEvent):
# If the render window has been closed, end the program.
    if(self.renderWindow.isClosed()):
        return False
```

How about the esc key?

```
if toggle < 0 and self.Keyboard.isKeyDown(OIS.KC_ESCAPE):
    toggle = 0.1
    return False
```

Physics frame listener

1. Simple physics from assignment2
2. copy new position to cube's sceneNode
3. listen for arrow keys
    change velocity's x (left, right) and y (up, down), and z (in, out) components with arrow keys and pgUp, pgDown keys.

Camera control you already have from tutorials.

All done?

Next class: oriented physics