WebTOP Training: Interacting with an X3D Scene Using Xj3D

Resources:

- Xj3D Javadoc: The documentation for the Xj3D API. Virtually the only package you will directly use is **org.web3d.x3d.sai**.
  - Web Address: http://www.xj3d.org/javadoc/index.html
- Xj3D SAI Tutorial: A simple tutorial for using Xj3D.
  - Web Address: http://www.xj3d.org/tutorials/general_sai.html
  - **In this assignment, you will primarily use the following sections:**
    - How to load a world
    - How to change a field value
    - How to listen for field changes
- Paul Cleveland's (kivic) tutorial on finding the source of a X3DFieldEvent
  - Web Address: http://www.web3d.org/message_boards/viewtopic.php?p=2404#2404

Assignment

This assignment assumes you have completed the assignment on creating a sphere and a cube which move in the opposite direction of the other using a Script node. You will need the Transforms, Shapes, and Sensors from that assignment. You will not be using the Script node.

This assignment is effectively the same as the previous one with the cube and sphere, except that you will use the Xj3D API within a Java class to perform the same functionality as the Script node. You will learn how to use Xj3D do the following:

- □ Load and display an X3D scene you have written
- □ Connect to nodes and field via the Scene Access Interface (SAI)
- □ Change field values using SAI
- □ Listen for and respond to changes to fields (input/output)

WebTOP Training: Interacting with an X3D Scene Using Xj3D

Part I: Working through Xj3D's SAI tutorials

□ Set up your development environment for Xj3D (See "Setting up your environment")

- http://www.xj3d.org/tutorials/xj3d_application.html

□ Work through the following sections of the SAI tutorial (see ***Resources*** for URL).

- How to load a world
- How to change a field value
- How to listen for field changes
  (Hint: Whatever X3D files you load will need to be in the working directory.)

Part II: Extending the SAI tutorials

□ Using the "How to listen for field changes" tutorial as a guide, make a Java application that connects to your cube and sphere X3D file in the following manner.

- Connect to the **diffuseColor** field of your cube's Material node.
- Listen for events from the **isOver** field of a TouchSensor on your cube. Whenever you receive a TRUE event value, change the color of the cube using the connection to its Material node's **diffuseColor** field. When you receive a FALSE event (the mouse is has exited the cube's geometry), return the color to its original setting. The choice of colors does not matter.

□ In addition, do the same for your sphere. You will notice that you need to be able to determine which Sensor sent the event. The Xj3D tutorials (at the time of this writing) do not cover this. See ***Resources*** for a link to Paul Cleveland's tutorial on determining the source of an X3DFieldEvent.

Part III: Revisiting the cube and sphere with SAI

Use what you have learned so far to remake your original cube and sphere assignment, but without using a Script node. Instead of routing the outputs of the sensors to a Script node, listen for changes to them by using a X3DFieldEventListener, determining the source of the event, and responding appropriately.