

CS488 Computer Graphics

Section 1 (Craig Kaplin)

Assignment 3

Submitted by:
Shawn Henderson
sbhender
00183318

1.0 Additional Documentation

1.1 Data Structure

Static Members

- last_id is an integer storing the last id used, this is assigned to the node id on construction, and last_id is incremented. This ensures that each node has a unique numeric id.
- hitList is an integer set storing the list of currently selected geometry nodes. This was stored here as it seemed to be more associated with the Nodes than the viewer. Viewer accesses this and updated it appropriately, and the Nodes use it to determine if they are selected to draw themselves differently.
- nodeMap is a map from integers to SceneNode pointers, the integer being integer names of the node.
- nodeNameMap is similar to the above, except maps string names to the node pointers. This is mainly used for picking given the name.

SceneNodes were given get_id and get_name accessors, as well as single SceneNode pointers to their parent.

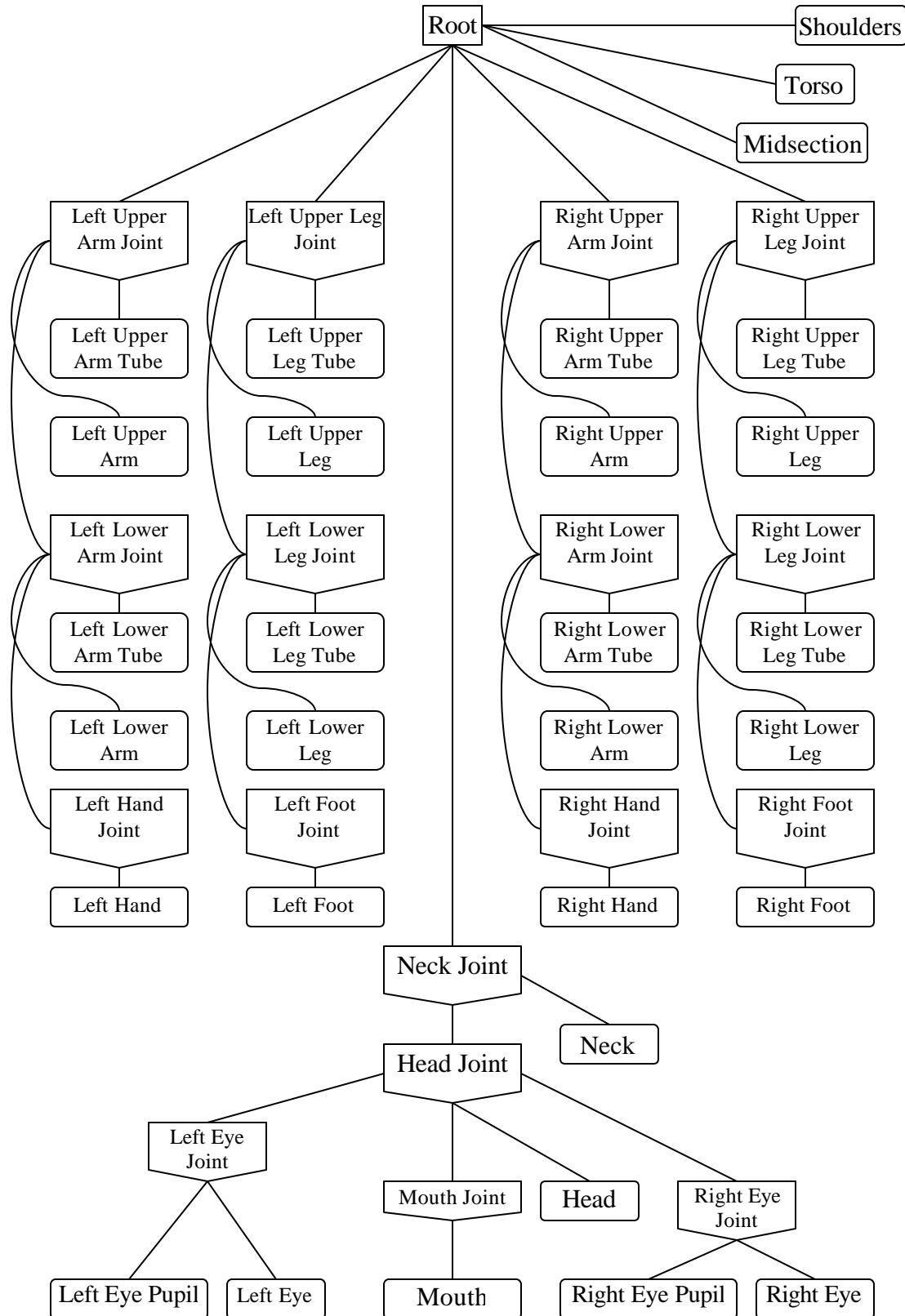
A get_joint() method was included that returns the joint immediately higher in the scene graph, it is recursive so that geometry nodes can be nested inside other kinds of nodes before ultimately the joint that controls the node. If no joint exists for the node in question, a null pointer is returned.

The JointNode's JointRange struct was altered to include a variable storing the current angle selected by the viewer for that joint.

JointNodes separate out the transformation matrix that is set as the joint is created, and the ones that respond to the joint's angles as they are changed. To combat floating point inaccuracies as rotations occur, the initial transformation matrix is never altered (except as done so by the scene definition file). The angles of the joints are stored, and when the transformation needs to be fully expressed and used, the base transformation is then multiplied with rotation matrices created from those joint angles.

JointNodes provide set_joint_* and get_joint_* methods for both joint axis 'current angle' variable. Also reset_joint was created to reinitialize the current angle variable to the initial one as specified when the joint is created.

1.2 Puppet Hierarchy



1.3 Extra Primitives

Due to the complexity of the shapes involved to create this puppet, and to limit the number individually selectable, many were modeled as single units. All except sphere are defined as face and vertex lists.

- Torso - core section of the puppet
- Midsection – plastic tubing center of torso
- Head - Full head except eyes and lower mouth
- Upper Arm - shape doubling as right and left upper arm and leg portions consisting of two bars and rotation cap
- Lower Arm - shape doubling as right and left lower arm and leg, single bar and rotation cap
- Hand - shape doubling as right hand and foot
- Mouth - Carved out shape used as lower mouth
- Neck - Complex shaped made to resemble corrugated plastic pipe
- Cube - used for pupils
- Tube - used as foam tubing around arm and left bars
- Shoulders – a pair of stretched cylinders

2.0 Extra Features

2.1 Select Menu

Select All, Select None and Select Inverse all behave as the names suggest.

The Name List allows the picking of each geometric element given its name. This was not implemented as specified for the Pick menu, as selection with the mouse has been implemented. The List of menu elements are not checkboxes, as the on-screen visual indicator is sufficient.

2.2 Options Menu

Display Grid - Displays a grid plane in the three axis intersection at the origin of the puppet.

Display Joints - Displays a thick wireframe sphere at the locations of the joints, which are transformed in the same way as the children nodes.

Display in Wireframe - Displays the puppet as a coloured lit wireframe to allow easier view of the joints when displayed.

2.3 Additional Joints

All joints use only one degree of freedom, with the exception of the head, eyes, hands, and feet, which have two. The third mouse button will affect the eyes, hands and feet in the same way as the head, if they are selected.

2.4 Selection

There are some joints that have multiple geometry nodes associated with it as direct children. They are both selectable, and will influence the joint for rotation. Selecting both at the same time has no affect on joint rotation sleep.

3.0 Acknowledgements

The display grid and the trackball code were taken from intDemo by Greg Veres.
The picking buffer analysis loop code was adapted from samples in the Red Book.