

# Real-Time 3D Techniques for Games (59-6920-00L)

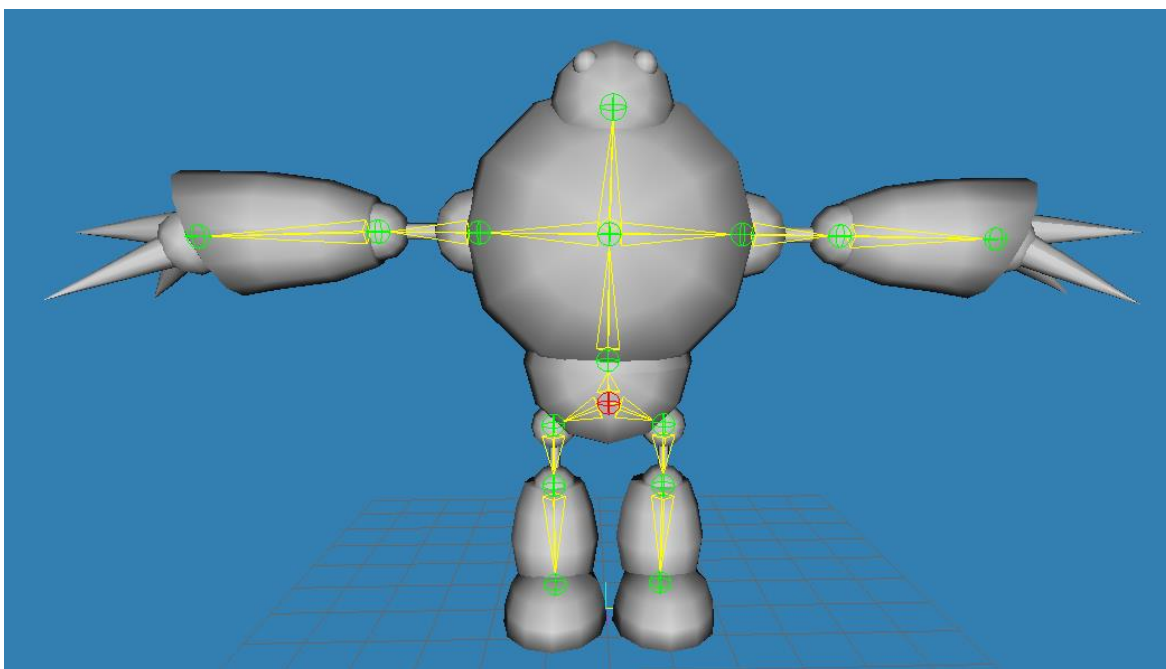
## Semester 1 Assignment

### Additional information for Animated Hierarchies

The robot animation hierarchy contains 16 separate animating parts that create a humanoid “bone” structure. Each bone (apart from the root) also has a parent bone and an offset position relative to that parent bone which are given in the table below. This data is also provided in the form of a text-based data file see: hierarchy.txt. The “Dot X Files” directory contains 15 DirectX “.X” format files for each of the visible parts. These have been named according to the bones which control them for ease of reference. The scale difference between 3D Studio and DirectX means you will need to divide these offset values by 10.

Bone	Name	Parent	Offset XYZ
1	root	none	1.027778, 75.644722, 0.000000
2	pelvis	root	-0.250011, 15.250000, -0.000005
3	body	pelvis	0.500099, 43.749992, 0.000003
4	left_shoulder	body	46.000000, 0.000000, -0.009992
5	left_elbow	left_shoulder	34.250019, -0.499817, -0.004262
6	left_wrist	left_elbow	55.250008, -0.999710, 0.003968
7	right_shoulder	body	-44.500023, 0.500000, -0.000021
8	right_elbow	right_shoulder	-33.999996, 0.250229, -0.000194
9	right_wrist	right_elbow	-60.000381, -1.750183, 0.007156
10	neck	body	0.249983, 36.625015, 25.999998
11	left_hip	root	19.500000, -7.724991, 0.000000
12	left_knee	left_hip	0.000006, -22.200001, 0.000000
13	left_ankle	left_knee	-0.800152, -36.399994, -0.000098
14	right_hip	root	-19.500000, -7.724991, 0.000000
15	right_knee	right_hip	0.000006, -22.000000, 0.000000
16	right_ankle	right_knee	0.199911, -36.799995, 0.000039

**Note:** The offset position given for the root bone is the offset between the floor position and the root.



## Animation Pipeline:

You have also been provided with three DAE files containing the animation data for the idle, attack and die animations. The steps below detail the process of exporting data out of Maya and into files which can be used by your application. If you repeat the same process then it should be possible to create new animations and load them into your application as well.

Animations must be exported using the COLLADA exporter with ONLY the “Joints and Skin” and “Animations” options checked under “Filter export”. It is also critical that the files are exported using “Export All” rather than “Export Selection” otherwise the bone order is exported incorrectly. The .DAE files were renamed as text files so that can be edited in Visual Studio then the regular expression below was used to match irrelevant parts of the file and replace them with an empty string. This isn’t strictly necessary as your code should be able to ignore the irrelevant bits itself, but it makes it easier for humans to interpret!

```
(^.*<?xml.*(.*\n)+?.*</asset>.*$\n)|
(^.*<source.*(interpolations|tangents).*(.*\n)+?.*</source>$\n)|
(^.*<sample.*sampler.*(.*\n)+?.*</sampler>.*$\n)|
(^.*<technique profile="MAYA">.*(.*\n)+?.*</technique>$\n)|
(^.*<channel.*>.*$\n)|
(^.*<library_materials.*(.*\n)+?.*</COLLADA>.*$\n)
```

## Interpreting the Data

Note that the “input-array” data represents the time (in seconds) of each key frame from the start of the animation and the “output-array” data represents the corresponding transformation data for each keyframe. Some data (such as translation data) may contain more than one piece of data for each keyframe as indicated by the “stride” setting. This is true of the X,Y and Z translations, but ONLY when a file has been exported containing keyframes for all three. If you create a new animation which doesn’t include keyframes for all three translation axes then they may be included separately in the file.

You may use third-party libraries such as tinyxml (<http://www.grinninglizard.com/tinyxml2/index.html>) to help you load the data. Although this will help you create a tidy and robust solution, it is not a shortcut and it is unlikely that it will save you any time overall. If you get desperate then you may reformat the original data into whatever form you like (to help you load it), but the most marks will be awarded for data-driven approaches and hard-coding should only be a very last resort.

Good luck!

Jake