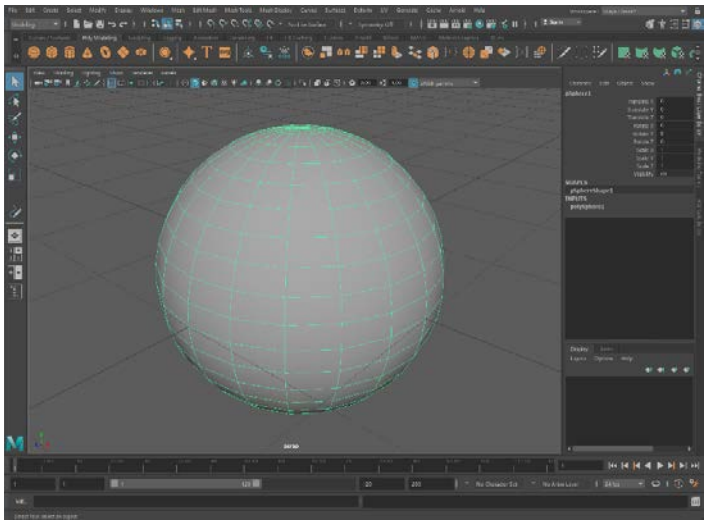
# 3D Imaging Applications for Visualization

3D imaging is very complex with many different file formats and surface types. Some but not all 3D applications support animation as well as 3D modeling. The most common 3D surfaces are:

- **Polygonal –** Surfaces are composed of many small planar triangles. Polygonal surfaces are the least intensive to render, but can require more disk and memory storage. This is the most common 3D surface type.
- **Parametric –** Surfaces are generated mathematically via parameters and functions. Parametric surfaces can require less disk and memory resources than a comparable polygonal model. Like vector graphics, parametric surfaces are completely scalable but are much more expensive to render and are often converted to polygonal triangles at render time.
- **Volumetric –** A three dimensional space is diced into small cubes which are assigned properties to describe what exists in each cubic volume. Minecraft, for example, uses simple Voxels to describe its world and everything in it. In research, volumetric surfaces are often used in scan data and medical imaging. Models that are 3D printed are sometimes converted to a volume in the process. Volumetric surfaces can also be defined by density fields where a surface is created at the point where density crosses a set threshold.
- **Particles –** Particles are not a surface type, but are as important to understand for visualization purposes. A particle is minimally a 3D position. Particles will often represent a point sample of a simulation and contain additional information at each sample such as direction, velocity, mass, density, and age. Simulations often have particles numbering in the millions. Though commonly used to represent fluids and gases, particles can represent distributions or systems of any sort. Rendering particles requires assigning a primitive or surface of some type to each particle.

## Interface



Most 3D applications follow a similar interface convention. In the center is a 3D viewport that can be split and configured to show multiple views of the scene. To the left of the viewport, a toolbox containing common selection and transform tools. Above the viewport, tabbed tool panels usually organized by mode or function. To the right of the viewport, information panes and a layer panel. Below the viewport, a time slider and playback controls if the application supports animation.

## Working with 3D

Three dimensional modeling and animation programs are very complex and demand the use of a minimal set of hotkeys for basic navigation. Most applications employ similar conventions or can be configured to behave similarly. We will start with some simple navigation, and quickly create a physics simulation in Maya.

**1: Navigating in 3D**

- Click the tab labeled **Poly Modeling** above the viewport.
- Click the **Cube Icon** in the left hand side of the tab to add a polygon cube.
- Hold the **Alt** key, press the **Left Mouse Button,** and **Drag** the mouse to orbit.
- Hold the **Alt** key, press the **Right Mouse Button,** and **Drag** the mouse to zoom.
- **Scroll** the **Mouse Wheel** to zoom.
- Hold the **Alt** key, press the **Middle Mouse Button,** and **Drag** the mouse to pan.
- **Left Click** in the empty space of the viewport to deselect the cube.
- **Left Click** on the cube to select it. Notice information related to the cube appears in the **Channel Box** on the left and the cube is highlighted in the viewport.

**2. Setting up a Simple Simulation**

- In the upper right corner, click the **Workspace** dropdown and select **MASH.**
- With the cube selected, click the **Create MASH Network** icon in the MASH tab to the upper left. You should have a line of cube instances in the viewport.
- In the **Channel Box,** to the right, click the **MASH1_Distribute** tab.
- In the **MASH1_Distribute** tab, set **Distribution Type** to **Grid.**
- Set all three **Distance** values to 8.
- Set all three **Grid** values to 8. You should now have a cubic array of cubes in the viewport.
- Press **f** to focus the view on the cube array.

**3. Adding Dynamics**

- Click the **Outliner** tab to the left of the viewport.
- Click the **MASH1** node to select it.
- Click the **MASH Editor** tab next to the outliner tab.
- Click the **+** icon on the **MASH** node and select **Dynamics** from the dropdown to add dynamics.
- In the **Channel Box,** to the right, the **MASH1_Dynamics** parameter pane should be visible.
- Under **Physical Properties,** set **Friction** to **0.5.**
- Under **Physical Properties,** set **Damping** to **0.25.**
- Under **Physical Properties,** set **Bounce** to **0.6.**
- In the **Play Bar,** below the viewport, set the playback range to **300** to extend playback
- **Press Play** to see the simulation.

## 3D application options

Most of the commercially available 3D applications offer free or discounted licenses for students.

**Free**

- Blender – blender.org

**Commercial**

- Maya – autodesk.com
- 3DSMax – autodesk.com
- Houdini – sidefx.com
- Modo – foundry.com
- Cinema 4D – maxon.net