1.0 What Is A Virtual World?

*Definition* Virtual: to exist in effect, though not in actual fact.

You are probably familiar with arcade games such as pinball and target games. Many versions of arcade games and simulations are two-dimensional (2-D), with flat images having no depth. Nevertheless, these simulations do allow you to steer a vehicle on a view of the road and scenery ahead. Two-dimensional graphic simulators are often used in flight training and driver education programs. The advantage of the simulator is obvious. When the fighter plane crashes under the hands of the novice pilot, neither the pilot nor the aircraft is injured. When a simulator is implemented in a three-dimensional (3-D) manner, it is called a *virtual world*. Using a virtual world lends a sense of reality to the simulator and increases its effectiveness.

To better understand the difference between a 2-D and a 3-D scene, consider the diagrams shown below, in Figure 1-0-1. The movie set scene-front on the left is a flat mock-up that has no real depth to its existence. When you walk around to the side of the mock-up, you see its true 2-D character. But, on the right is a real building. When you walk around to the side of the building, you see the continuation of its 3-D structure.

![Figure 1-0-1, 2-D movie set mock-up versus 3-D building](image)

So, how is this sense of reality obtained? How do we create a 3-D world on a computer's 2-D monitor screen? To answer these questions, consider what the three dimensions are. In Figure 1-0-2 below, the three dimensions are width (horizontal, left-to-right), height (vertical, top-to-bottom), and depth (front-to-back). In truth, the picture is only two-dimensional, but the artist created the illusion of *depth.*
How does the artist create the illusion of depth? One way is to draw some objects larger and other objects smaller. In Figure 1-0-2, the white rabbit appears closer and the locomotive appears farther away. We know the rabbit is much smaller than a locomotive in real life. So, the size of the objects in the picture fools the human brain into interpreting the rabbit as being closer. Just as an artist can draw objects in different sizes, so can the computer. But, a computer must calculate the positions and relative sizes of the different objects. Even though these calculations appear simple, they can be highly complex, take up a lot of memory, and consume much computing power.

Creating 3-D images is difficult, but not impossible. The computer just does the calculations and displays the image. But, in a flight simulator or a video game the objects must be moving. How is that done? Many of the same techniques are used to give the illusion of motion as are used by the Disney film studio in creating animated cartoons. Animation is a fantasy of vision, an illusion. To perform this illusion, the filmmaker and artist collaborate to create a sequence of artwork frames (drawings or images) where each successive frame has a slightly different view of a scene. The scene is drawn with objects, then same scene is redrawn with the objects positioned in a slightly different place. The same scene is redrawn and the objects moved just a bit more, over and over and over! For example, here is a sequence of frames (left-to-right) showing a ball-like figure in each one:
In animation production, such frames are photographed in sequence on a reel of film. The processed reel of film is run through a movie projector at a rate (frames/second) which displays the sequence faster than the human eye can detect. Therefore, the human brain is tricked into perceiving the ball (in the above sequence of frames) as falling through the air. Instead of creating just one picture, many, many pictures are created and displayed in rapid sequence -- creating an illusion of motion. A similar effect can be created on a computer screen. One way to do this is to rapidly display successive image files on a monitor.

Until recently, desktop computers have not been powerful enough to update changes to 3-D scenes quickly enough to make 3-D animation possible. But, in the early to mid-90's, desktop machines experienced phenomenal advances in memory storage and processor speeds. These hardware improvements make it possible to draw repeated 3-D scenes at animation speeds (upwards of 10 frames or more per second). Using desktop machines to generate repeated 3-D graphic images allowing the creation and control of tiny virtual worlds within color monitors. The user works with the virtual world with a keyboard and a mouse to cause the animation to start, stop, be replayed and interact with the user.

The Difference Between Virtual Worlds and Virtual Realities: Not all 3-D graphics are virtual worlds. Some 3-D graphics systems are called virtual realities. In these 3-D graphics systems, the user wears a Head Mounted Display (HMD) and special glasses (3-D vision is obtained using two tiny displays -- one for each eye). Virtual reality (VR) systems are sometimes referred to as immersive 3-D graphics. While desktop virtual world systems respond to mouse and keyboard controls, the VR systems have no use for the keyboard and mouse so the user is faced with a lack of good user interface controls. Development of user interface controls for VR is a subject of current research. Hardware advances have spurred a real interest in virtual worlds and virtual reality. Unfortunately, the current state of the art with 3-D software development is still in its infancy. Mainly, this is because most development tools are still primitive and require a high degree of programming experience and mathematical understanding. Virtual reality is beyond the scope of this book and is not used in Alice.

Alice is a software development tool for developing virtual worlds. The person using a virtual world created with Alice is able to interact with the 3-D images. The images can be made to move, turn, spin, react to a mouse, and more! The really great thing about Alice is that it is designed with the goal of being easy to learn to use. With a reasonable effort, you can acquire the programming skills and mathematical understandings that allow you to create your own virtual worlds.

It is important to remember that Alice is still under construction. This means that new ground is being explored in developing learning materials to be used with Alice. And, it means that the Alice environment is likely to crash, perhaps frequently. Patience is a vital characteristic when using an edge-of-the-frontier technology.
1.1 An Alice World – Setting the Scene

The first part of creating a virtual world is to create the background. Every virtual world in Alice starts with the same background, a simple blue sky and a green land surface. You will be able to change this simple background by any, or all, of the following means:

- adding objects to the scene,
- changing the background sky by adding images on top of the default sky,
- changing the color of the sky,
- changing the color of the ground, and/or
- changing the texture of the ground.

Each of these modifications will be discussed in the lab section of this chapter.

Figure 1-1-1 – A Default World

Figure 1-1-2 – A Default World with clouds
Figure 1-1-3 – Image in Figure 1-1-2 with the camera moved back

Figure 1-1-4 – Image in Figure 1-1-2 with the camera tilted up
The appearance of the initial scene is determined by components such as:

- The graphic details – background images may be added to scenes. Common examples are clouds in the sky or a sunset image that replaces the blue sky. Figure 1-1-1 contains the default background that starts every world in Alice. Figure 1-1-2 is the result of adding the happysky.mdl background image to the original world. The happysky.mdl image may be added to the default scene by clicking on the Add Object button, selecting the scenery folder, and then selecting the happysky.mdl image.

- The camera -- the perspective of the scenic view is that of the camera. By adjusting the height and tilt of the camera (as well as its position), the view can be changed. Like a camera-crew member at a televised football game, one can turn the camera to get a panoramic view of the scene, or of a particular part of the scene. Starting with the scene from Figure 1-1-2, the scene in Figure 1-1-3 is produced by moving the camera backwards. The camera is moved backwards by clicking on the Inspects button and moving the mouse backwards. Starting with the scene from Figure 1-1-2, the scene in Figure 1-1-4 is produced by clicking on the up arrow of the Tilt box.

- The ground -- the surface can be textured to appear solid as rock or liquid as water. Starting with the scene in Figure 1-1-2, the scene in Figure 1-1-5 is produced by removing the ground texture. Changing the texture of the ground is accomplished by right mouse button clicking the ground icon to the left of the image, and then selecting Display Style, and then Texture, and then Off. Of course, in some scenes, such as in outer space, there is no ground surface.

- The lighting -- the quantity of light available and the angle of lighting can be adjusted to simulate sunrise, sunset, and so on.

It is important to understand that Alice is not a 3-dimensional graphics drawing program. Examples in this book use objects from the extensive gallery provided with Alice. We recommend that you use objects from this gallery, look on the web for other.mdl files or, (for the more adventurous) use Teddy to build your own 3-D objects. You may wish to experiment with Teddy. If so, just read and follow the excellent tutorial provided with Teddy. Take a look at:

http://www.mtl.t.u-tokyo.ac.jp/~takeo/teddy/teddy.htm

Also, Alice is not a sound recording package. We recommend that you look on the web for sound clips. There is a good list of web sites available on the Alice web pages. Take a look at:

You can set up your background by adjusting the camera angles, lighting, and textures of a "bare bones" world, and possibly changing the background view or and/or colors. Once the background has been set up, the world can be populated with animated objects and sound tracks to create your own 3-D animation with sound. This will be the topic of the next few sections.

One last note: In the next section, you will learn how to add objects that inhabit your world. In terms of computer representation, there is no difference between objects used to create the scene (an island, the cloudy sky, sunset, etc.) and objects that inhabit the scene (such as helicopters, white rabbits, tail-less cats). The real difference is objects that are used to create the world scene are static (non-moving) while objects that inhabit the world are dynamic (moving and changing) when the animation runs. As there is at least a conceptual difference between the static objects and the dynamic objects that become part of the animation, they are presented in separate sections.