Chapter 1

Introduction

Hello, and welcome to this book about 3D programming with ShiVa. Firstly, I'd just like to say a big "THANK YOU" to StoneTrip (www.stonetrip.com), without whose help this book would not have been possible. For those of you who aren't familiar with either 3D programming, or ShiVa, don't worry because the next couple of Chapters will get you up and running in no time, and throughout this book I'll try to keep it simple and in plain English, so it should be pretty easy to follow.

OK, that's enough rambling, let's get down to introducing the star of this particular show:

ShiVa®

ShiVa is a 3D authoring environment which can be used for creating pretty well any type of 3D application that you can think of, though it does tend to lean a bit towards game programming. The following list is a summary of the capabilities of ShiVa:

- An up-to-date graphics engine with all the effects you could need, from cartoon to ultra-realistic;
- Reflection and refraction abilities;
- Particle and trail systems;
- 2D post-render effects;
- 3D special effects, such as billboarding, fog and skyboxes;
- A built-in terrain creator and editor;
- A built-in physics engine using the ODE engine;
- Keyframe and skeletal animation;
- Automatic pathfinding;
- A WYSIWYG (what you see is what you get) HUD creator and editor;
- The ability to use multiple input devices including the Wiimote®;
- A scripting system using a version of LUA®;
- A built-in networking system;
- Standalone players for Windows, Mac OSX, Linux, PDA, Smartphone, iPhone and the Web.

One of the great things about 3D programming is that you can do pretty much anything you want to do. Of course some things are easy to do, and some are extremely hard, though, more often than not, there are tricks that can produce very good approximations of hard tasks.

WARNING!

ShiVa is aimed at the indie developer and small studio, so you will not be able to make the new Crysis or Far Cry, but you can still make a very professional 3rd Person Shooter!

Some great examples of 3D applications (mainly games, of course, as almost every budding 3D developer wants to make the next blockbuster game) are:

Games:

- Crysis: Warhead (Crytek Budapest)
- Dead Space (EA Redwood Shores)
- Command & Conquer – Red Alert 3 (EA Los Angeles)
- Far Cry 2 (Ubisoft Montreal)

Other Apps:

- Medical
- Architectural

The above applications have some very advanced features and effects that their respective teams of developers have worked on over many years, so don't be too upset when you find out that you can't replicate some of these (such as deformable terrain) without using some sort of trick to make it look similar (let's face it, the "big boys" of the 3D industry have their own little bags of tricks). Also, don't forget about the platform that you're going to develop for, since this will have a massive bearing on what is, and isn't, possible!

The available platforms for 3D applications are extremely varied, and have many different combinations of available options. ShiVa (as at v1.7) caters for most of the mainstream platforms (see above), and there are constant rumours, on the StoneTrip Developer Forum (http://developer.stonetrip.com/), that ShiVa will be adding more platforms in the future.

Another thing that will confuse most beginners is the wide variety of programming languages used in the 3D industry. These are used to write the actual engine (mostly in C as it is by far the fastest common programming language) and include (with my own personal opinion on them):

- C – Fastest, but hardest to program
- C++ – Based on C, slower, easy to program
- Java – Similar speed to C++, easy to program
- Visual Basic – Slowest, very easy to program

IMPORTANT!

You cannot get away from the fact that you **will** need to use some sort of programming language to build a decent 3D application. This is, one of the main drawbacks to any sort of application development.

There are some 3D platforms that claim that you do not need to learn programming. These tend to create apps based on fixed templates, and as such do not allow as much freedom as an environment like ShiVa.

There are also many "scripting" languages that have been developed to sit over the top of the 3D engine. These are used to write code that interacts with the engine, and most are optimised for use with their particular engine. ShiVa, for example, uses a scripting language based on LUA.

OK, that's enough of an intro. The next Chapter is going to be a brief look at 3D programming so you can skip this bit if you want, but for the total newbies I would suggest at least skimming through it, as it will give you a grounding in some of the terms used in the 3D world, and also give you an idea of some of the things that are possible. By the way, since games are the predominant use for 3D applications, I'll be concentrating on them from now on.