CHAPTER 1 Getting Started

This chapter will discuss a few very basic things about Lua and World of Warcraft addons before we dive into the Lua language in Chapter 2. You are also going to see a few useful tools we will use throughout the book.

What Is Lua?

Lua is a programming language developed in 1993 by Roberto Ierusalimschy, Luiz Henrique de Figueiredo, and Waldemar Celes at the Pontifical Catholic University of Rio de Janeiro in Brazil. Lua is Portuguese for moon; the correct pronunciation is “LOO-ah.” Lua is available under an MIT license, a very unrestrictive license that basically allows you to use Lua for anything as long as you keep the copyright notice and license.

The advantages of Lua are its extensibility, simplicity, efficiency, and portability. It is easy to write modules that add functionality, and it’s also easy to embed Lua as a scripting language in another program like a game. Scripting languages are programming languages that are used in a specific part of another larger application. So World of Warcraft (written in C++) uses Lua for its user interface (UI). There are actually many games out there that use Lua to script the user interface or for configuration files (for example, Warhammer Online or Far Cry). But it is also possible to write whole programs in Lua, as there are extensions available that allow you to use libraries or frameworks like wxWidgets (wxLua), the .NET Framework (LuaInterface), or 3D engines like OGRE (Lugre) from Lua. These extensions are also called bindings, as they bind Lua to a more powerful complex framework or library.

Lua is easy to learn; the syntax is simple and clear. One can read and understand small Lua scripts without knowing anything about the language. So it is possible to use Lua as a language for configuration files that can be edited by people who don’t know anything about programming. Lua is a dynamically typed language, which basically means that you will have a lot of freedom and flexibility while programming. You will learn more about this feature in Chapter 2. Despite its simplicity, Lua is a very powerful multiparadigm programming language. It provides imperative, functional, and object-oriented paradigms to write your scripts. I will show what these paradigms mean for you in the following chapters. You also don’t need to take care of memory management; Lua provides a very good incremental garbage collector. A garbage collector is a program that runs interleaved with your script and frees any unused memory your program had previously used; incremental means that it is running all the time, working in small steps.

Even though it is a fully featured powerful multiparadigm language, Lua is still extremely fast. The virtual machine (a program that takes Lua code and executes it) that has been used since Lua 5.1 is one of the fastest such programs for scripting languages out there. There is also a just-in-time compiler (a program that translates Lua to native machine code for your computer on the fly) for x86 architectures available that is even faster.

Lua can quickly compile and execute long scripts with thousands of lines of code and easily handle huge amounts of data. It is possible to write Lua scripts that create other Lua scripts consisting only of huge tables with hundreds of thousands of entries, to create persistent data in your script. The Lua compiler can quickly load and compile these data files on demand.

If you think you need to install a few megabytes of software to get Lua running on your computer, think again. The Lua virtual machine is just over 160 KB, and the compiler adds only another 200 KB. (These file sizes are for Lua 5.1.4 on Windows x86.)

Lua is written in plain ANSI C, so it is very portable. It runs on almost every machine and operating system you have ever heard of, and some you haven’t; for example, Lua has been successfully compiled on the LEGO MINDSTORMS NXT platform.

A good place to get additional information about Lua in general is its web site, <http://www.lua.org> .

What Are WoW Addons

The user interface of World of Warcraft can be modified using Lua and XML. XML is a powerful data description language, but you don’t need to worry about it until Chapter 5. It is possible to replace the whole default UI with highly customized so-called addons. Lua is used as scripting language, while XML can be used as a simple way to create graphical user interface element.

Blizzard Entertainment, the publisher of WoW, provides a very powerful user interface API (application programming interface, a set of functions that can be used to communicate with the game from a scripting language, in this case Lua) that allows the player to script almost every part of the game. There are addons for almost everything, from unit frame replacements and action bars over chat utilities to in-game mini-games like Bejeweled.

Users quickly developed addons that Blizzard considered too powerful because they were close to bots. Addons like Decursive were able to scan the raid for debuffs and dispel them with one click. Another addon let the user select the best target and healing spell for healers just by pressing a single key. Healers who used these mods could just press the same key over and over again to play perfectly during a fight. Another mod even allowed you to run certain predefined paths, like corpse runs in Molten Core.

So patch 2.0 restricted certain actions to “secure code,” which is code from the default UI. The restricted actions are targeting, casting spells, and movement-related functions. However, many functions may still be used under certain circumstances, so it is still possible to create a raid frame mod. But it is no longer possible to create mods like those just mentioned, which automate combat-related tasks during combat.

The default UI provides all functionality a normal user needs to play the game, but addons can provide advanced functions or a different theme. Figure 1-1 shows the default UI that is provided by the game. A heavily modified user interface that replaces almost everything could look like the UI illustrated in Figure 1-2.

The user interface in Figure 1-2 replaces almost everything. It looks like the UI of a completely different game, replacing even the default font that is used by the game. You can also see the replaced minimap, action bars, and tool tips.

Everyone can go to a web site and download an addon and install it, though only a few of the 11.5 million World of Warcraft players are able to create mods. You will be one of this select few after you’ve read this book. You probably can’t wait to get started, so let’s see some useful tools we’ll use in the book.

Useful Programs, Tools, and Web Sites

Blizzard provides almost no documentation for its interface API, but you can read the source code of the default UI to figure out how the provided functions work. There are a few web sites that can help you by providing Blizzard’s code and documentation for the interface API. And there are a few in-game tools that can aid you while debugging or writing code.

So this section presents some web sites and programs we are going to use throughout the book.

These web sites contain valuable information. You will need them quite often, especially the sites with documentation for API functions.

http://www.wowwiki.com is a site that you are going to use all the time. It is a wiki that provides a lot of valuable information about the interface API. The most important pages in this wiki, which you are going to need very often, are shown in Table 1-1.

The web site http://wowcompares.com allows you to browse or download the code of the default user interface. The site also keeps track of changes to the default UI code. You can compare or download unified diffs (files that contain all the changes between two versions) of different versions of the World of Warcraft interface

WowAce / Curse

http://www.wowace.com is the web site of the Ace project, which now belongs to http://www.wowace.com. You can find Ace-related mods as well as documentation for the Ace framework there. Curse is the biggest web site for addons, so it is a good idea to publish the addons you are going to write there.

UI & Macros Forum

The official UI & Macros forum at <http://forums.worldofwarcraft.com/board>. html?forumId=11114&sid=1 is a good place to look for official statements. You can also find interface-related patch notes there

DeadlyBossMods.com

If you have any questions about this book and its example code, feel free to ask in my forum at <http://www.deadlybossmods.com>.

Lua.org / Lua-users.org

The official Lua web site is http://www.lua.org. You can download or browse the Lua source code and read the official Lua reference manual there. You can also read the first edition of the book there. The site http://lua-users.org is an unofficial wiki for Lua.

In-Game Tools

In-game tools are World of Warcraft addons that can help you to debug or edit your addons in the game.

DevTools

DevTools is a must-have addon for developers. You can get it on WoWInterface.com: http://www.wowinterface.com/downloads/fileinfo.php?id=3999. It provides you several slash commands, one of which is

This evaluates an expression and shows its value. It is capable of displaying complex data structures in your chat frame, so you can easily debug them.

Another powerful feature of this mod is its event trace frame (see Figure 1-3), which can be shown by using /devents .

The mod has a few more useful features; you should read its readme file, which explains everything

Swatter

Swatter provides an improved error frame, which is needed because the default frame that displays Lua errors in World of Warcraft is unusable. Unlike the default error frame, Swatter can handle more than one error message at the same time and it shows a stack trace for debugging. You can see an error message that was caught by Swatter in Figure 1-4.

This addon is included in some addon packages, such as Auctioneer, but a stand-alone version is available at http://zip.morganna.org/libs/!Swatter/

WoWLua

WoWLua is an in-game script editor with Lua syntax highlighting. You can write, save, load, and execute Lua scripts in World of Warcraft with this addon. Figure 1-5 shows the editor with a simple Lua script.

Download: <http://www.wowinterface.com/downloads/info7366-WowLua.html>

TinyPad is similar to WoWLua. It is an in-game text editor that can execute text as a Lua script. Unlike WoWLua, it does not provide syntax highlighting, but it supports working with item links, so it is useful when you are debugging a mod that uses or creates item links

Interface AddOn Kit

The Interface AddOn Kit is a program that automatically extracts the default UI. Extracting and working with the default UI is covered in Chapter 3.

Download: <http://us.blizzard.com/support/article.xml?articleId=21466>

MPQ Editor

The MPQ Editor is a powerful program that allows you to browse and modify the MPQ archives used by World of Warcraft. An MPQ file is an archive format used by Blizzard’s games; these archives contain all the data files used by the game. Using this program is also covered in Chapter 3.

Download: dppl6++sss\*vavqh]\*jap+aj+ilm+`ksjhk]`\*dpih

IDEs

You don’t need a fully featured integrated development environment (IDE) at all to work with Lua, as a normal text editor is sufficient. An IDE usually combines a text editor with a compiler and a debugger. The problem with an IDE when programming for World of Warcraft is that your debugging environment is normally the game. But an external program cannot access the game, so you will spend a lot of time debugging your mod in-game. That is not to say that an IDE is completely useless. A good IDE can help you by providing basic features like syntax highlighting and checking or autocompletion while writing your addon.

I’ll provide a customized version of SciTE that I use to write addons. You can find it in the code download package for this book on <http://www.apress.com> as well as on my site, http://www.deadlybossmods.com. But I will also present some other IDEs here. It is up to you which IDE or editor you use.

LuaEdit

LuaEdit is a fully featured IDE for Lua, but without any support for World of Warcraft. The debugger especially can be very powerful. It allows you to view Lua’s internal stack and all local variables, set breakpoints, and execute your script line-by-line.

This IDE is a good choice when you are developing a stand-alone Lua script. Its use for World of Warcraft addons is limited as you won’t be able to use most of the debugging features the IDE provides.

Download: dppl6++sss\*hq]a`ep\*knc

WoW UI Designer

WoW UI Designer is a very powerful IDE that can edit Lua scripts as well as XML files. It even has a GUI designer for World of Warcraft (see Figure 1-7). However, the IDE is still in the beta state and the Lua editing part especially lacks a few features; for example, there is no autocompletion for API functions available.

Its GUI designer is a great tool if you need to create a simple graphical user interface like a configuration menu without digging through large XML files. Its XML editor is also quite good because it can check whether your XML is valid and also whether will be recognized by the game. You will learn more about XML in Chapter 5 when we discuss frames.

Download: dppl6++sss\*sksejpanb]\_a\*\_ki+`ksjhk]`o+ejbk0...)SkSQE@aoecjan\*dpih

AddOn Studio

AddOn Studio is the most powerful IDE available for World of Warcraft. It is based on Microsoft Visual Studio, but it is a stand-alone program with everything you need to run included in the 56 MB download.

This IDE offers everything you might need while developing addons: a basic WYSIWYG GUI designer (see Figure 1-8), Lua syntax highlighting, and even basic IntelliSense support for World of Warcraft API functions. Note that IntelliSense and similar autocompletion features for a dynamically typed language like Lua can never be as powerful as they are for a language 10 CHAPTER 1 N GETTING STARTED with static typing like C++ or Java. The IDE cannot know the type of your variables or the methods of your objects as they are created dynamically when your program runs.

Download: dppl6++sss\*\_k`alhat\*\_ki+S]n\_n]bp=``KjOpq`ek

SciTE

SciTE (Scintilla-based Text Editor) is the editor I use to develop mods. It does not have fancy features like a GUI designer, but it is simple, fast, extensible, easy to use, and powerful. It supports Lua syntax highlighting, autocompletion, and call-tips for World of Warcraft API functions. SciTE can be fully configured through plain-text files. You can change almost everything with these config files,such as the syntax highlighting, code folding, and the behavior of the editor. Figure 1-9 shows SciTE with a simple Lua script.

The editor also embeds Lua as a scripting language, so you can write Lua scripts to add new functionality to your IDE. This can be a very powerful feature; it is covered in the last chapter of this book, where we look at other uses for Lua.

SciTE is available for Windows, Linux, and OS X, unlike the previously mentioned IDEs, which are only available for Windows.

Download: SciTE is included in the source code download for this book, at http://www.apress.com