

World of Warcraft Programming:
A Guide and Reference for Creating WoW Addons

魔兽世界编程宝典

——World of Warcraft Addons完全参考手册

James Whitehead II
(美) Bryan McLemore 著
Matthew Orlando
杨柏林 张卫星 王聪 译



```
Icons\Spell_Shadow_ShadowW  
SendMessage(self:GetName() toSt  
ForClicks("AnyUp", "AnyDown"  
Load></OnLoad><OnClick> ..  
SendMessage(self:GetName() toSt  
Scripts></Button></Ui>  
to: /  
com/wow/w  
'XMLSch  
read  
read  
read
```

创建WoW插件的指导和参考手册

清华大学出版社



World of Warcraft Programming:
A Guide and Reference for Creating WoW Addons

魔兽世界编程宝典

——World of Warcraft Addons完全参考手册

WoW不仅仅是一个缩写词……

阅读完本书，您将学会如何对魔兽世界界面进行自定义修改。如果您初学编程，将从本书学习到Lua和XML的基础知识，以及如何开始编写自己的第一个插件。如果您已经熟悉一些原始插件，那么可以直接跳到本书的第三和第四部分，使用模板、函数钩、自定义图形、状态标题等。本书有助于您发布自己的插件，使WoW更加有趣。

本书内容

- ✦ 如何响应事件，创建窗体和使用WOW API与游戏交互
- ✦ 通过一系列交互式的示例、值和变量，以及自定义函数和控制结构来学习Lua编程语言
- ✦ 学习基本和高级的函数和控制结构
- ✦ 编写结构良好的XML和有效地使用模式
- ✦ 实现已保存的变量、slash命令、安全模板、动态窗体、解析工具提示等
- ✦ 创建自定义图形、滚动窗体和下拉菜单
- ✦ 轻松地在综合性的API参考文献、API目录、事件和工具集中找出特定的函数

作者简介

James Whitehead II 创建的插件包括PerfectRaid、Clique、TomTom和其他可以从www.wowinterface.com网站下载的插件。他还是*Hacking World of Warcraft*一书的作者之一。

Bryan McLemore早在2005年就加入了魔兽世界UI社区，他是Ace Project和WowAce.com的创始人之一。

Matthew Orlando创建的插件包括BuyEmAll、CogsBar、MacroTalk和其他流行的WoW插件。他不仅精通Lua语言和魔兽世界API，还维护Cogwheel's Complete Macro Guide。

上架建议：计算机图形/游戏编程

读者信箱：wkservice@vip.163.com

投稿信箱：bookservice@263.net

 **WILEY**
wiley.com

ISBN 978-7-302-22239-2



9 787302 222392 >

定价：118.00元

魔兽世界编程宝典——
World of Warcraft Addons
完全参考手册

清华大学出版社

北 京

James Whitehead II, Bryan McLemore, Matthew Orlando

World of Warcraft Programming: A Guide and Reference for Creating WoW Addons

EISBN: 978-0-470-22981-1

Copyright © 2008 by Wiley Publishing, Inc.

All Rights Reserved. This translation published under license.

本书中文简体字版由 Wiley Publishing, Inc. 授权清华大学出版社出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字: 01-2009-2534

本书封面贴有 Wiley 公司防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

魔兽世界编程宝典——World of Warcraft Addons 完全参考手册/(美)怀特黑德(Whitehead, J.) 等著;
杨柏林 等译. —北京: 清华大学出版社, 2010. 6

书名原文: World of Warcraft Programming: A Guide and Reference for Creating Wow Addons

ISBN 978-7-302-22239-2

I. ①魔… II. ①怀… ②杨… III. ①计算机网络—游戏—程序设计 IV. ①G899②TP311.5

中国版本图书馆 CIP 数据核字(2010)第 054392 号

责任编辑: 王 军 李楷平

装帧设计: 孔祥丰

责任校对: 成凤进

责任印制: 杨 艳

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 清华大学印刷厂

经 销: 全国新华书店

开 本: 185×260 印 张: 55.5 字 数: 1351 千字

版 次: 2010 年 6 月第 1 版 印 次: 2010 年 6 月第 1 次印刷

印 数: 1~3000

定 价: 118.00 元

产品编号: 030671-01

译者序

魔兽世界(WoW)无疑是一款倍受青睐的游戏。它是著名的游戏公司——暴雪娱乐(Blizzard Entertainment)公司制作的第一款网络游戏，属于大型多人在线角色扮演游戏。玩家可以在其中扮演自己喜欢的英雄四处冒险，完成挑战性的任务，探索未知领域，加入公会与志同道合的人一起体验游戏的乐趣等。其精美的画面，引人入胜的游戏情节，让无数玩家废寝忘食流连忘返。据最新的统计数据显示，目前魔兽世界在全球累计用户数已经突破了1100万，甚至超过了现在地球上许多小国的全国人口总数。自2005年4月正式公测以来，现在已有500多万注册用户见证了这款史诗般的游戏势不可挡的市场占有，曾经运营过它4年的九城公司也在商业上取得了很大的成功。

魔兽世界游戏之所以能取得如此巨大的成功，在很大方面上得益于它的开放插件机制。魔兽世界插件是魔兽世界的辅助工具类程序，它允许玩家定制符合自己个人爱好的游戏界面，并允许玩家根据需要改变相应功能。毫无疑问，魔兽世界也为魔兽世界插件创造了巨大的商机。目前流行的魔兽世界插件主要包括3类。第一类是整合插件类，安装之后就可以使用。这些整合插件有一个共同的特征是插件包中的任何一个插件都是无法单独使用的。第二类是单体插件，就是只能实现某一种或几种功能的插件。相对于整合插件，单体插件的功能多样，可以随意选择、随意组合以及调试系统等。第三类是根据原有的单体插件，修改或者整合以后派生的插件，这种插件现在越来越流行。

魔兽世界的游戏客户端包括两个主要部分：游戏世界和用户界面。游戏世界是一个三维世界，它包括很多角色和互动对象。这些元素在脚本语言的接口中是不容易访问并且不能改变。用户界面由游戏客户端的其他元素组成，包括按钮、窗体、地图和选项窗口等。用户可以编写插件来增加或改变已有的元素从而增强功能，或者以一种不同的方式显示信息。魔兽世界编程，即利用Blizzard提供的官方编程接口编写脚本程序，对WoW默认的界面进行自定义修改。

本书由浅入深、全面细致地介绍了如何编写程序实现各种插件。本书不需要任何编程基础，书中大量通俗易懂的示例程序可供读者参考和实践。首先，本书介绍了Lua和XML语言，之后分别讨论了各种插件的构成方式，并通过一些详细的实例揭示了插件的实现方法。此外，本书最后还提供了综合性的API参考文献、API目录、事件和工具集等。本书能够满足不同层次用户的需求，包括各种新手插件用户，中级插件作者以及资深插件开发人员。相信在本书的帮助下，您能够创造出属于自己的、绚丽多彩的、功能丰富的魔兽世界插件！

本书共 31 章，其中正文中第 1 章，第 6~15 章，第 17 章、第 18 章、第 20~29 章由浙江工商大学计算机与信息工程学院杨柏林副教授译(共计 23 章)；第 2~5 章、第 16 章、第 19 章、第 30 章、第 31 章由郑州大学软件技术学院张卫星老师译(共计 8 章)；北京航空航天大学虚拟现实与技术国家重点实验室的王聪同学完成附录等部分的翻译工作。最后，译者要感谢郑州大学信息工程学院的吴云鹏等同学，他们也为本书的翻译做了很多工作。敬请广大读者提供反馈意见，读者可以将意见发送到 wkservice@vip.163.com，我们会仔细查阅读者发来的每一封邮件，以求进一步提高今后译著的质量。

杨柏林

2010 年 3 月 26 日

作者简介

James Whitehead II 是许多流行的魔兽世界插件的作者，他的作品有 PerfectRaid、Clique、LightHeaded、TomTom 以及 www.wowinterface.com 里给出的许多其他小插件。从魔兽世界 Beta 版问世起，他就开始开发插件，并且一直活跃在魔兽世界 UI 论坛和 Lua 社区中。当他正在用自己的众多账号中的一个玩魔兽世界的时候，您可能会在游戏服务器的 Stormrage(US)区中遇到他，或者在 Ravencrest(EU)区中看到 he 正在兴奋地练级。

Jim 毕业于雪城大学(Syracuse University)，他在那里获得了计算机专业的学士和硕士学位。现在，他正在英国牛津大学攻读博士学位，同时他也是该校计算实验室(Computing Laboratory)的助教。在空闲时间里，他喜欢代表伍斯特学院(Worcester College)参加划船比赛，或者与朋友们一起野外探险。

Bryan McLemore 从 2005 年初就成为魔兽世界 UI 社区的一名成员。在开发了 KC_AutoRepair 插件后，他成为 Ace Project 和 WowAce.com 的创始人之一，以 Kaelten 的 ID 与 Brent Miller(魔兽世界 UI 社区的注册账号)共同工作。在加入 Brent 的官方部门后，Bryan 开始将越来越多的时间花在 WowAce 上，而不是开发自己的插件上；然而，他仍然坚持继续开发 OneBag 和 KC_Items 插件。闲暇时，他一般在美国服务器上的 Whisperwind 区中玩魔兽世界。

另外，在他的其他空闲时间里，他还负责开发 Ace3，并且专注于开发自己的插件，同时维护几个相关的网站。**Bryan** 最近与妻子和女儿搬到加利福尼亚州，开始从事 Web 开发和系统管理的工作。他们现在居住在旧金山市。

Matthew Orlando 是诸多魔兽世界插件的作者，这些插件包括 BuyEmAll, CogsBar 和 MacroTalk。他具有 13 年的编程经历，并从 2006 年初开始活跃于插件社区中。除了精通 Lua 语言和魔兽世界 API 外，他还维护着 “Cogwheel’s Complete Macro Guide”，这是一个关于魔兽世界宏指南的重要论坛。

Matthew 目前与妻子 **Juliella** 居住在加利福尼亚州的塞巴斯坡(Sebastopol)市。他在 Kistler Vineyards 公司从事客户数据服务、数据库编程和网站开发等方面的工作。

前言

魔兽世界游戏(WoW)自 2004 年 11 月 23 日发售以来,已迅速地成为有史以来最流行的视频游戏之一。目前它 900 多万的用户,它已经家喻户晓了。魔兽世界是一个引人入胜的环境,您不需要多少时间上手便可以定制角色、探索新的世界,以及与朋友组团合作。一些玩家晚上花费 4~6 小时在公会进行组队,试图打败最后、最强大的怪物。另一些玩家则喜欢 PVP 战斗,将时间花在竞技场(Arena)或是战场(Battleground)上,试图提高他们的声望。还有一些玩家仅仅是享受通过游戏与团队或是朋友在一起的时间。魔兽世界满足了各种玩家的需求,这很可能是它如此成功的主要原因。

游戏实现各种娱乐风格的一种方式,便是以插件的方式定制用户界面。对于那些具有技术灵感,或是单纯地追求完美的玩家,暴雪公司将其用户界面的定制完全开放,允许这些玩家改变其所有功能。插件可以是简单地改变生命条的颜色、增加一个显示随机表情的新命令,也可以复杂到提供多功能的网络日历,以允许公会组织和协调事件。除此之外,暴雪继而提供了对用户界面社区的推动和支持,这种方式是其他游戏开发人员所从未尝试过的。

用户界面社区在过去的几年中迅速地增长,并且没有停止的迹象。编写本书的目的在于为读者提供所需要的工具来为魔兽世界用户界面进行自定义修改,包括对编程语言、术语以及插件结构的介绍。目前,急需编写的插件成千上万个,本书也为您提供了实现它们的方法。

读者定位

本书可以为以下对象提供帮助:希望学习如何修改已有插件的新手插件用户,希望在作品中添加更多令人兴奋的功能的中级插件作者,以及资深插件开发人员——他们可以将本书作为极其复杂的魔兽世界用户界面系统的参考。本书假定,读者对魔兽世界游戏十分熟悉,并且对编程感兴趣。了解任何一门编程语言的读者将会发现书中的很多概念十分熟悉。

具有极少、甚至没有编程经验的读者应当首先着重于本书的第 I 部分,该部分讲授了 Lua——一门用于编写插件的编程语言。尽管没有任何编程经验的读者会学习到足够的知识来创建和修改插件,但是他们或许仍然希望(或者需要)从其他的资料中得到更加通用的编程课程。

对于具有编程经验的读者,前面的几个章节会非常简单。其中的示例可以使这些读者十分迅速地掌握 Lua 编程语言的基本语法。若读者已经对高级脚本语言(如 Python)非常熟悉,则可以简单地跳过前几章,然后直接进入第 II 部分,即基本的插件编程。这一部分详

细介绍了魔兽世界插件系统是如何工作的，并引导读者完成第一个插件的编写。

已经编写过插件的读者或许希望直接跳到本书的第III部分。该部分的章节引入了一些特殊概念，并展示了在插件编写工作中使用这些概念的示例。同时，也包括对一些更加复杂的系统(如工具提示、下拉菜单和状态头)进行了详细讨论。

此外，本书的第IV部分包括了关于魔兽世界 API 的极为详细的参考，包括事件和工具集。

本书的组织方式

本书分为 4 部分，其主题的复杂程度逐渐由低到高。

第 I 部分介绍了 Lua 和 XML，使读者快速地上手这两门语言，便于创建插件。

第 II 部分讨论了插件的初步构成，以及事件驱动(event-driven)的魔兽世界插件编程的基本概念。在本部分，读者将创建自己的第一个插件，并熟悉 WoW 中的常用 API。

第III部分通过从头到尾实现一些插件，将读者引入更加高级的主题中。

最后，第IV部分为整个 API 的完备参考，包括函数、工具集、事件以及安全模板。

网站上的信息

每隔几个月，暴雪会为魔兽世界发布新的补丁，补丁可能会引入新的内容、修正已有错误，甚至会极大地改变游戏机制。结果是，本书所涉及的内容会随着时间的改变。为了处理这个问题，我们已经为本书建立了一个相应的网站：<http://wowprogramming.com>。我们不期望对核心概念进行彻底的改动，但任何特定的细节改动都会在网站中列出，包括这些改动是如何影响本书内容的。除了作为一个内容勘误表之外，该网站也包含了本书所有参考内容的在线版本。

进一步的信息

魔兽世界用户界面社区是一个非常令人兴奋的场所，其中包含了许多无限可能的想法，以及这些想法的定制和实现。魔兽世界本身已经是一个很有趣的游戏；使用它作为平台来开发插件，在为提供帮助的同时，增强他们的游戏体验，对于作者本身也是一种鼓励。

目 录

第 I 部分 学 习 编 程

第 1 章 关于魔兽世界的编程	2
1.1 定制用户界面	2
1.1.1 插件是如何工作的	2
1.1.2 插件能够做什么	3
1.2 Lua 新手上路	3
1.2.1 下载和安装 WowLua	4
1.2.2 在网上使用 Lua	4
1.2.3 下载并安装 Lua 解释器	5
1.3 探究您的 Lua 解释器	6
第 2 章 探讨 Lua 基础知识	7
2.1 使用 Lua 解释器	7
2.1.1 运行命令	7
2.1.2 错误信息的理解	8
2.1.3 使用历史信息来做改变	8
2.1.4 退出解释器	9
2.2 处理数字	9
2.2.1 基本算术运算	9
2.2.2 科学记数法	10
2.2.3 十六进制表示法	11
2.2.4 理解浮点	11
2.3 理解值和变量	12
2.3.1 探讨值和它们的类型	12
2.3.2 使用变量	13
2.4 使用字符串	16
2.4.1 比较字符串	16
2.4.2 多个字符串的连接	16
2.4.3 将数字转换成字符串	17
2.4.4 将字符串转换成数字	17
2.4.5 引用字符串	17
2.4.6 转义特殊的字符	19
2.4.7 获得字符串的长度	20

2.5 布尔值和运算符	21
2.5.1 使用 and 运算符	21
2.5.2 or 运算符的使用	21
2.5.3 not 运算符的使用	22
2.5.4 理解 nil 值	22
2.6 探讨作用域	23
2.6.1 块(Block)	23
2.6.2 组块(Chunk)	24
2.7 小结	24
第 3 章 基本的函数和控制结构	25
3.1 使用函数	25
3.1.1 创建函数	25
3.1.2 局部函数	26
3.1.3 函数的参数和返回值	26
3.1.4 函数作为 Lua 值	27
3.2 用 if 语句进行判断	28
3.2.1 简单的条件	28
3.2.2 复杂的表达式	29
3.2.3 扩展的条件语句	29
3.2.4 显示个人问候信息	29
3.3 用 while 语句来重复动作	30
3.3.1 计算阶乘	31
3.3.2 while 和 repeat 之间的差异	32
3.4 用数值执行 for 循环	32
3.4.1 计算阶乘	34
3.4.2 循环条件的求值	34
3.4.3 循环中的变量作用域	34
3.5 小结	35
第 4 章 使用表	36
4.1 用表来存储数据	36
4.1.1 创建和索引表	36
4.1.2 从表中清除元素	37

4.1.3	字符串关键字的快捷方式	37
4.1.4	创建有内容的表	38
4.2	把表当作数组使用	39
4.2.1	创建数组	39
4.2.2	获取数组的长度	40
4.2.3	在数组中添加元素	40
4.2.4	从数组中删除元素	41
4.2.5	对数组中的元素排序	42
4.3	用名称空间使用表	43
4.3.1	创建 util 名称空间	43
4.3.2	向 util 中添加函数	43
4.4	表的面向对象编程	44
4.4.1	创建非面向对象计数器	45
4.4.2	把表作为简单的对象	45
4.4.3	用冒号调用对象方法	46
4.4.4	用冒号定义函数	46
4.4.5	创建更佳的计数器	47
4.5	利用元表对表进行扩展	48
4.5.1	添加元表	48
4.5.2	定义元方法	49
4.5.3	旁路元表	54
4.6	小结	54
第 5 章	高级函数和控制结构	55
5.1	多值返回	55
5.2	将十六进制转换成 RGB	55
5.3	指定多个值	56
5.4	返回值丢失	56
5.5	WoW 中的多个返回值	57
5.6	接受可变数目的参数	58
5.7	声明变参函数	59
5.8	结合 select() 函数使用	60
5.9	范型 for 循环和迭代器	61
5.10	范型 for 语句的语法	61
5.11	遍历表的数组部分	61
5.12	遍历完整的表	62
5.13	表的清除	63
5.14	其他的迭代器	63
5.15	对表的数组排序	63

5.16	定义样例数据	64
5.17	默认的排序顺序	64
5.18	创建比较函数	64
5.19	创建更复杂的比较函数	65
5.20	小结	66
第 6 章	Lua 标准库	67
6.1	表库	68
6.1.1	table.concat (table [, sep [, i [, j]]])	68
6.1.2	table.insert (table, [pos,] value)	68
6.1.3	table.maxn (table)	68
6.1.4	table.remove (table [, pos])	69
6.1.5	table.sort (table [, comp])	69
6.2	数学函数库	69
6.3	字符串函数	72
6.4	格式化新字符串	73
6.5	模式匹配	75
6.5.1	字符类型	75
6.5.2	模式选项	77
6.5.3	模式捕获	77
6.5.4	模式锚点	78
6.5.5	模式的例子	78
6.6	模式匹配函数	78
6.7	Lua 在 WoW 插件中的补充	80
6.8	函数别名	81
6.9	小结	82
第 7 章	学习 XML	83
7.1	XML——一种标记语言	83
7.1.1	XML 与 HTML 的关系	84
7.1.2	XML 的组件	84
7.1.3	创建规范的 XML	85
7.2	验证 XML 文档	86
7.2.1	规范定义的样例	86
7.2.2	XML 文件的例子	87
7.2.3	魔兽世界中的 XML	88
7.2.4	使用 GradientType	89
7.2.5	探讨模式	89

7.3 小结.....	89	9.4.1 为插件命名 (创建目录).....	108
第 II 部分 魔兽世界编程		9.4.2 创建.toc 文件.....	109
第 8 章 魔兽世界编程概述.....	91	9.4.3 创建框架.xml 文件.....	109
8.1 游戏中运行和测试代码.....	91	9.5 使用外部库.....	109
8.1.1 安装.....	91	9.6 小结.....	109
8.1.2 使用/script 和/run.....	92	第 10 章 在 XML 中创建窗体.....	110
8.1.3 显示输出.....	92	10.1 了解 UI 对象.....	110
8.2 在游戏中编辑代码.....	93	10.1.1 为对象进行命名.....	110
8.2.1 TinyPad.....	93	10.1.2 指定父对象.....	111
8.2.2 Omnibus.....	94	10.1.3 设置对象尺寸.....	112
8.2.3 WowLua.....	94	10.1.4 锚定对象.....	113
8.3 编写自定义 print() 函数.....	94	10.1.5 将窗体和图形元素分层.....	115
8.4 插件和脚本的局限性.....	95	10.1.6 一般属性.....	117
8.5 用 Blizzard 插件编写资源.....	95	10.2 创建纹理.....	118
8.5.1 用户界面自定义工具.....	95	10.2.1 添加颜色.....	118
8.5.2 用户界面和宏论坛.....	97	10.2.2 添加渐变效果.....	118
8.6 论坛资源.....	97	10.2.3 添加图形元素.....	120
8.6.1 Wowwiki.....	97	10.3 使用字体字符串创建文本.....	120
8.6.2 魔兽版本监视器.....	97	10.3.1 使用模板.....	121
8.7 小结.....	97	10.3.2 使用的例子.....	121
第 9 章 插件解析.....	98	10.3.3 进一步自定义.....	122
9.1 开发属于您自己的插件 文件夹.....	98	10.4 探讨窗体类型.....	123
9.1.1 暴雪的插件.....	98	10.4.1 按钮(Button).....	123
9.1.2 自定义插件.....	99	10.4.2 复选按钮 (Check Button).....	123
9.2 插件组件和文件.....	99	10.4.3 颜色选择(ColorSelect).....	124
9.2.1 内容表格文件(.toc).....	99	10.4.4 编辑框(Editbox).....	124
9.2.2 Lua 脚本文件.....	104	10.4.5 游戏工具提示 (Game Tooltip).....	124
9.2.3 XML 文件.....	104	10.4.6 消息窗体 (Message Frame).....	125
9.2.4 媒体文件.....	104	10.4.7 小地图(Minimap).....	125
9.3 本地化您的插件.....	106	10.4.8 模型(Model).....	125
9.3.1 可用的本地语言.....	106	10.4.9 滚动信息窗体(Scrolling Message Frame).....	126
9.3.2 本地化的原因.....	106	10.4.10 滚动窗体 (Scroll Frame).....	126
9.3.3 鼓励玩家建议.....	106		
9.3.4 实现本地化.....	107		
9.4 创建插件框架.....	108		

10.4.11	简单的 HTML 窗体	126	12.5.1	<OnLoad>	145
10.4.12	滑动器(Slider)	126	12.5.2	<OnEvent>	145
10.4.13	状态栏(Status Bar)	127	12.5.3	<OnClick>	145
10.4.14	飞行路线窗体	127	12.5.4	<OnDragStart>	145
10.5	小结	128	12.5.5	<OnDragStop>	145
第 11 章	向 XML 窗体中添加行为	129	12.6	为 CombatTracker.lua	
11.1	理解事件和脚本	129	添加函数		146
11.1.1	窗体脚本	129	12.6.1	CombatTracker_	
11.1.2	游戏事件	129	OnLoad(frame)		146
11.2	使用脚本响应窗体事件	130	12.6.2	CombatTracker_ OnEvent	
11.2.1	<OnEnter>和		(frame, event, ...)		147
<OnLeave>		130	12.6.3	CombatTracker_	
11.2.2	<OnLoad>	131	ReportDPS()		148
11.2.3	<OnEvent>	131	12.7	测试 CombatTracker	149
11.2.4	<OnClick>	132	12.7.1	窗体拖拽	149
11.2.5	<OnUpdate>	133	12.7.2	右击报告: 第 I 部分	150
11.3	可用的窗体脚本	133	12.7.3	测试战斗跟踪	150
11.4	使用窗体方法改变窗体	135	12.7.4	右击报告: 第 II 部分	150
11.4.1	常用方法	135	12.8	小结	151
11.4.2	指定类型的函数	135	第 13 章	运用魔兽世界 API	152
11.5	创建并使用模板	136	13.1	理解魔兽世界 API	152
11.5.1	模板有用的原因	136	13.1.1	常规 API	152
11.5.2	新建 XML 模板	136	13.1.2	类库 API	153
11.5.3	使用 XML 模板	137	13.1.3	FrameXML 函数	154
11.6	使用默认 UI 工具集模板	138	13.1.4	受保护函数	154
11.7	小结	139	13.1.5	单位函数的	
第 12 章	创建您的第一个插件:		使用与关闭		155
CombatTracker		140	13.2	创建简单的单位窗体	156
12.1	定义规格说明	140	13.2.1	创建窗体	157
12.1.1	CombatTracker		13.2.2	添加数据域	157
用户要求		140	13.2.3	设定窗体事件	
12.1.2	找准游戏事件	140	处理程序		160
12.2	创建插件架构	142	13.3	使用 API	164
12.3	定义 CombatTracker 的		13.3.1	显示和隐藏窗体	164
XML 窗体		143	13.3.2	实现简单的更新函数	165
12.4	测试 CombatTrackerFrame	144	13.3.3	显示生命和法力值	167
12.5	向 CombatTrackerFrame		13.3.4	更新敌对信息	169
添加操作脚本		144	13.4	小结	173

第 14 章 使用 AddOn Studio 创建

- 基本插件** 174
- 14.1 开始使用 AddOn Studio 174
- 14.2 可视化设计您的界面 175
 - 14.2.1 使用 Toolbox 176
 - 14.2.2 设置属性 176
- 14.3 编写代码 177
 - 14.3.1 监听事件 177
 - 14.3.2 自动添加事件处理程序 178
 - 14.3.3 验证单位是否存在 178
 - 14.3.4 添加“关闭”按钮
的代码 180
- 14.4 部署插件 180
- 14.5 小结 181

第III部分 高级插件技术**第 15 章 利用编程模板** 183

- 15.1 向 TargetText 添加增益 184
 - 15.1.1 创立基本模板 185
 - 15.1.2 定义基本的行为 186
- 15.2 创建增益图标 188
 - 15.2.1 创建栅格 189
 - 15.2.2 设定锚点 189
 - 15.2.3 创建特定的 wrapper
函数 190
- 15.3 更新增益 191
 - 15.3.1 对增益进行遍历 191
 - 15.3.2 对更新操作进行封装 193
- 15.4 小结 194

第 16 章 脱离 XML 编写插件 195

- 16.1 探索基本的窗体创建 195
 - 16.1.1 使用 CreateFontString()
函数 196
 - 16.1.2 使用 CreateTexture()
函数 196
- 16.2 创建更好的玩家窗体 197
 - 16.2.1 设置 Lua 197
 - 16.2.2 创建窗体 198

- 16.2.3 使它工作 199
- 16.3 比较 LUA 窗体和
XML 窗体 200
- 16.4 小结 203

第 17 章 使用安全模板 204

- 17.1 了解污染(Taint) 204
 - 17.1.1 启用污染日志 205
 - 17.1.2 执行污染 205
 - 17.1.3 变量污染 207
 - 17.1.4 蔓延的污染 208
 - 17.1.5 理解被保护的窗体 209
 - 17.1.6 无污染地设置
窗体属性 209
- 17.2 使用安全模板 210
 - 17.2.1 为动作按钮定义行为 210
 - 17.2.2 做一些简单的选择 214
- 17.3 使您的单位窗体充满生机 218
 - 17.3.1 移除禁止代码 218
 - 17.3.2 接受单击 219
 - 17.3.3 显示和隐藏窗体 219
 - 17.3.4 设置动作 219
- 17.4 小结 220

第 18 章 创建命令 221

- 18.1 检测基本的命令 221
- 18.2 小提示、技巧以及
好主意 222
 - 18.2.1 象征输入 222
 - 18.2.2 将确认用于模式匹配 223
 - 18.2.3 假跳转 224
 - 18.2.4 好主意 225
- 18.3 小结 226

第 19 章 用函数钩改变已有行为 227

- 19.1 什么是函数钩 227
 - 19.1.1 修改返回值 228
 - 19.1.2 函数钩工具脚本 228
 - 19.1.3 安全地钩一个函数 230
 - 19.1.4 安全地钩住脚本 230

19.2	决定什么时候使用函数钩.....231	第 22 章	创建滚动窗体.....253
19.2.1	理解钩链(Hook Chain).....231	22.1	使用滚动结构.....255
19.2.2	寻找替代.....232	22.1.1	定义滚动窗体.....255
19.3	设计插件: MapZoomOut.....233	22.1.2	添加子滚动条.....255
19.3.1	创建定时器.....233	22.1.3	创建滚动条.....257
19.3.2	创建 MapZoomOut.....234	22.2	创建伪滚动窗体.....258
19.4	小结.....236	22.2.1	添加滚动条.....261
第 20 章	创建自定义图形界面.....238	22.2.2	利用鼠标滚轮 进行滚动.....261
20.1	创建图形的普遍规则.....238	22.3	小结.....262
20.2	GIMP.....238	第 23 章	创建下拉菜单.....263
20.2.1	新建图片.....239	23.1	创建简单的下拉菜单.....263
20.2.2	添加图形组件.....239	23.1.1	添加切换按钮.....264
20.2.3	保存纹理.....240	23.1.2	创建下拉窗体.....264
20.3	Adobe Photoshop.....240	23.1.3	初始化下拉菜单.....265
20.3.1	新建图像.....241	23.1.4	设置下拉菜单.....267
20.3.2	添加图形组件.....241	23.1.5	测试下拉菜单.....267
20.3.3	创建透明度通道.....241	23.2	创建多级下拉菜单.....268
20.3.4	保存图像.....242	23.3	创建高级下拉菜单.....269
20.4	Paint Shop Pro.....243	23.3.1	功能菜单项.....270
20.4.1	新建图像.....243	23.3.2	CheckBox 菜单项.....270
20.4.2	添加图像组件.....244	23.4	调色板菜单项.....271
20.4.3	创建透明度通道.....244	23.5	自动创建菜单.....272
20.4.4	保存图像.....244	23.6	小结.....274
20.5	测试纹理.....245	第 24 章	工具提示.....275
20.5.1	没有按钮出现.....246	24.1	操作工具提示.....275
20.5.2	出现绿色方块.....246	24.1.1	使用“:Set...”函数.....275
20.6	小结.....247	24.1.2	包含自定义信息的 工具提示.....276
第 21 章	使窗体移动.....248	24.1.3	创建自定义工具提示.....277
21.1	拖动和放置.....248	24.2	解析工具提示数据.....278
21.2	标题区域.....248	24.2.1	基础结构剖析.....278
21.3	使用 OnMouseDown 和 OnMouseUp.....249	24.2.2	基本扫描循环.....278
21.4	使用 OnDragStart/ OnDragStop.....249	24.3	小结.....279
21.5	给窗体赋予生命.....250	第 25 章	使用状态标题.....280
21.6	渐隐.....251	25.1	创建第一个状态标题.....280
21.7	小结.....252	25.1.1	重设鼠标单击.....282

29.19	光标	693
29.20	配置变量	694
29.21	调试	694
29.22	阵营	695
29.23	GM 调查	695
29.24	GM 请求	695
29.25	公会	695
29.26	公会银行	697
29.27	查看	698
29.28	副本	698
29.29	物品栏	699
29.30	物品	699
29.31	物品链接	701
29.32	物品文本	701
29.33	知识库	702
29.34	键盘	702
29.35	寻求组队/寻求成员	703
29.36	拾取	704
29.37	Mac	704
29.38	Macro	705
29.39	邮件	706
29.40	地图	707
29.41	商人	707
29.42	其他	708
29.43	不同的单击	709
29.44	金钱	709
29.45	移动	709
29.46	NPC	710
29.47	小队	710
29.48	宠物	711
29.49	请求	713
29.50	玩家	713
29.51	配置	715
29.52	PvP	715
29.53	任务	716
29.54	团队	718
29.55	技能	719
29.56	社交	719
29.57	凹槽	719

29.58	声音	720
29.59	法术	720
29.60	形态	721
29.61	统计信息	722
29.62	天赋	723
29.63	目标	723
29.64	飞行器	724
29.65	跟踪	724
29.66	交易	724
29.67	交易技能	725
29.68	训练师	726
29.69	UI	727
29.70	单位	728
29.71	实用程序	730
29.72	视频	731
29.73	语音	732
29.74	区域	733

第 30 章 事件参考 734

第 31 章 工具集参考 794

31.1	工具集 API	794
31.1.1	UIObject	794
31.1.2	FontInstance	794
31.1.3	Region	795
31.1.4	VisibleRegion	796
31.1.5	LayeredRegion	796
31.1.6	Font	797
31.1.7	Texture	797
31.1.8	FontString	798
31.1.9	Frame	798
31.1.10	GameTooltip	801
31.1.11	EditBox	805
31.1.12	Slider	807
31.1.13	Model	807
31.1.14	MessageFrame	808
31.1.15	Cooldown	809
31.1.16	ScrollFrame	809
31.1.17	ColorSelect	810
31.1.18	Minimap	811