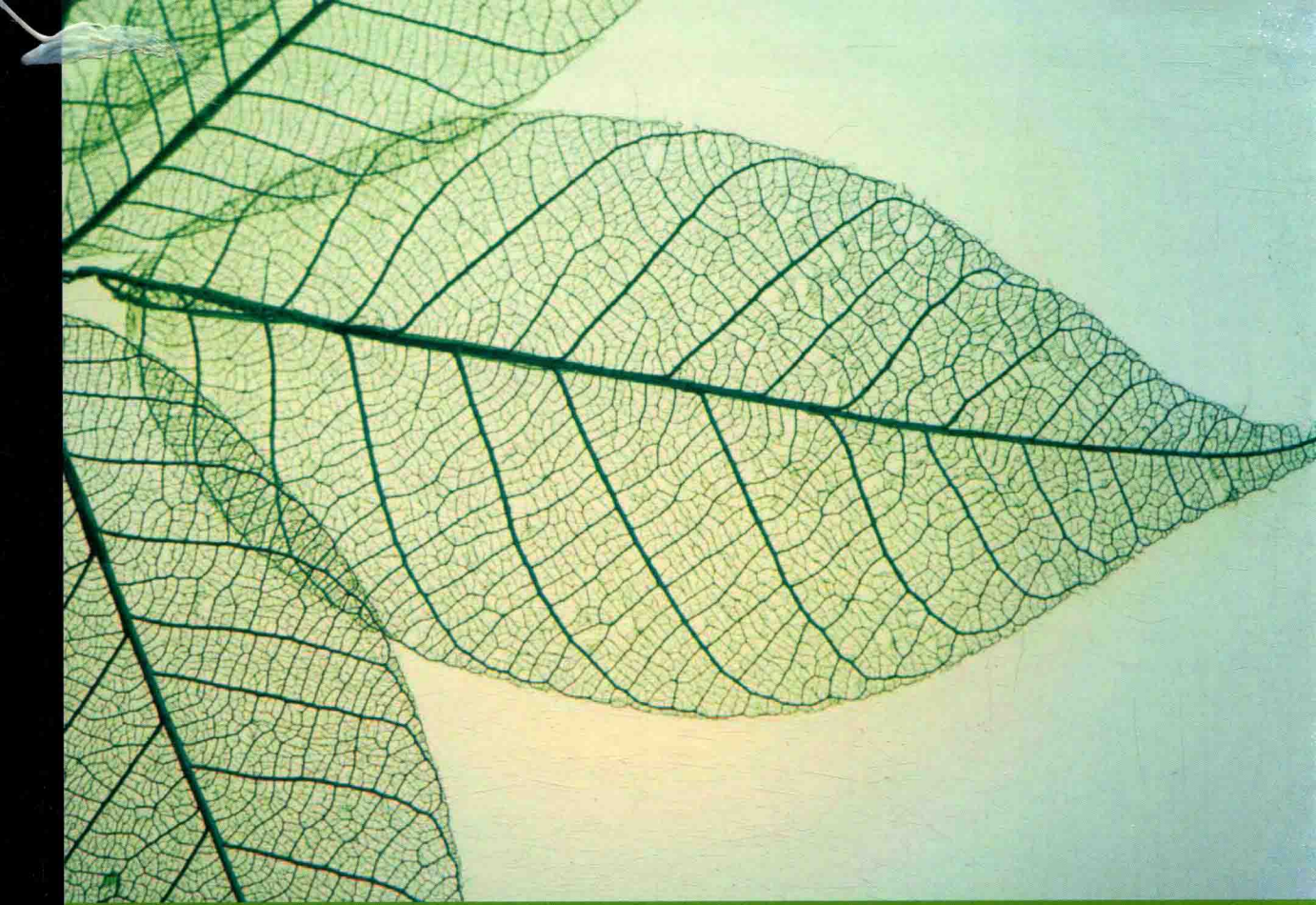




[美]Matt Smith Chico Queiroz 著 刘天慧 译

Unity 5.x实战开发指南

Unity 5.x Cookbook

清华大学出版社



Unity 5.x 实战开发指南

[美] Matt Smith Chico Queiroz 著

刘天慧 译



清华大学出版社

北 京

内 容 简 介

本书详细阐述了与 Unity 相关的基本解决方案, 主要包括核心 UI, 道具栏 UI, 2D 动画, 创建贴图 and 材质, 相机, 光照效果, 控制 3D 动画, 位置、运动和导航, 音频, 与外部文件和设备协同工作, 利用附加特性和优化操作改善游戏以及编辑器扩展等内容。此外, 本书还提供了相应的示例、代码, 以帮助读者进一步理解相关方案的实现过程。

本书适合作为高等院校计算机及相关专业的教材和教学参考书, 也可作为相关开发人员的自学教材和参考手册。

Copyright © Packt Publishing 2015. First published in the English language under the title *Unity 5.x Cookbook*.

Simplified Chinese-language edition © 2018 by Tsinghua University Press. All rights reserved.

本书中文简体字版由 Packt Publishing 授权清华大学出版社独家出版。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: 01-2016-5193

本书封面贴有清华大学出版社防伪标签, 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

Unity 5.x 实战开发指南 / (美) 马特·史密斯 (Matt Smith), (美) 奇科·奎罗斯 (Chico Queiroz) 著; 刘天慧译. —北京: 清华大学出版社, 2017

书名原文: Unity 5.x Cookbook

ISBN 978-7-302-47368-8

I. ①U… II. ①马… ②奇… ③刘… III. ①游戏程序-程序设计 IV. ①TP317.61

中国版本图书馆 CIP 数据核字 (2017) 第 124145 号

责任编辑: 贾小红

封面设计: 刘 超

版式设计: 魏 远

责任校对: 马子杰

责任印制: 王静怡

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 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: 三河市金元印装有限公司

经 销: 全国新华书店

开 本: 185mm×230mm 印 张: 31.5 字 数: 632 千字

版 次: 2017 年 12 月第 1 版 印 次: 2017 年 12 月第 1 次印刷

印 数: 1~3000

定 价: 99.00 元

产品编号: 068948-01

译者序

Unity 是近几年非常流行的一款 3D 游戏开发引擎（特别是移动平台），它的特点是跨平台能力强，支持 PC、Mac、Linux、iOS、Android、网页、虚拟现实（VR）和增强现实（AR）系统等几乎所有的平台，移植便捷，3D 图形性能出众，为众多游戏开发者所喜爱。在手机平台，Unity 几乎成为 3D 游戏开发的标准工具。本书则在此基础上讨论较为高级的开发技术和解决方案，

具体而言，本书提供了上百个 Unity 游戏开发方案，主要涉及 UI（用户界面）方案、2D 动画以及物理特性、物理着色器、相机的控制和增强技术、Unity 中的光照系统、角色动画、游戏中的音效和分轨音乐以及 Unity Editor 中的设计方案。

在本书的翻译过程中，除刘天慧之外，郭志杰、白永丽、赵洪玉、米玥、潘冰玉、李强、皮雄飞、史云龙、王巍、孙年果、程聪、朱利平、王晓晓、解宝香、李保金、王梅、林芮、刘鹤、张骞、张博、王烈征、李垚、张颖、张弢、刘君、李强、沈旻等人也参与了本书的翻译工作，在此一并表示感谢。

限于译者的水平，译文中难免有错误和不妥之处，恳请广大读者批评指正。

译者

前 言

游戏开发通常被视为一类较为复杂的任务，且包含了广泛的内容，并被认为是一个跨学科领域，其中包括人工智能、角色动画、数字绘图以及音频编辑。相关内容可具化为数百或数千个多媒体和数据资源。相应地，特定的软件应用程序将此类资源数据整合至单一产品中。

游戏引擎是一类较为特殊的软件，且相对复杂、深奥，其应用方式也十分烦琐，缺少相应的灵活性且价格高昂，一般仅供大型项目或核心程序员使用。在此基础上，Unity 应运而生。

Unity 体现了游戏开发过程中的开放性，作为引擎和多媒体编辑环境，Unity 兼具应用性和多样性特征。该软件包含了免费版本和高级版本，其中，高级版涵盖了更为丰富的特性。在本书编写时，Unity 支持下列平台的应用程序开发。

- ❑ 移动平台：包括 Android、iOS、Windows Phone 以及 BlackBerry。
- ❑ Web 平台：包括 WebGL。
- ❑ 桌面平台：包括 PC、Mac 以及 Linux 平台。
- ❑ 游戏机平台：包括 PS4、PS3、Xbox One、XBox 360、PlayStation Mobile、PlayStation Vita 以及 Wii U。
- ❑ 虚拟现实（VR）和增强现实（AR）系统：包括 Oculus Rift 和 Gear VR。

当今，Unity 在开发人员社区中常被提及，其中聚集了大量的学生和开发爱好者。一些大型的商业组织（包括工作室开发人员和国际化工作室）也采用 Unity 制作出了大量的游戏作品，相信读者在相关平台上也有所耳闻。

本书提供了上百个 Unity 游戏开发方案。其中，某些方案针对多媒体特性阐述了 Unity 应用程序技术，包括与动画协同工作，以及预安装资源包系统的应用。其他方案则与基于 C# 脚本的游戏组件开发相关，包括与数据结构协同工作、数据文件的管理以及针对计算机控制角色的人工智能算法。

如果读者希望采用一种有组织的、直观的方式开发高质量游戏，则需要学习如何创建有效的游戏组件，并处理某些常见问题，以使 Unity 更好地为读者服务。

本书内容

第 1 章介绍 UI（用户界面）方案，通过可视化交互元素、相关方案有助于提升游戏的娱乐性和体验值。在本章中，读者可领略到大量的 UI 技术，例如，文本和图像的更新操作、定向雷达、倒数计时器以及自定义鼠标指针。

第 2 章主要讨论玩家采集项，例如，通过按键开启门、武器装备的弹药量以及选择项（例如法术和攻击行为）。针对玩家的状态显示问题，本章提供了各种文本和图形解决方案。例如，玩家是否显示信息项，以及可采集项的最大数量。

第 3 章介绍功能强大的 2D 动画以及物理特性。本章所列举的方法有助于读者理解 Unity 中不同动画元素之间的关系；除此之外，本章还将探讨身体不同部位间的运动方式，以及精灵板图像文件（包含了精灵对象帧图像序列）的使用方式。

第 4 章讲述 Unity 5 中的物理着色器，并在此基础上理解贴图和材质的应用方式。对于图像编辑技能而言，这无疑是一类优秀的资源。

第 5 章讨论游戏相机的控制和增强技术，其中涉及独立相机和多台相机的工作方案。

第 6 章针对 Unity 中的光照系统特性讲解了大量的实际解决方案，例如，饼干纹理、反射贴图、光照贴图、光照和反射探头（probe）以及过程式纹理。除此之外，本章还介绍了投影仪的应用。

第 7 章集中讨论角色动画，并描述了 Unity 动画系统 Mecanim 的应用方式。本章涵盖了大量的主体，包括角色构建、过程式动画以及布娃娃系统的物理内容。

第 8 章针对计算机控制的对象和角色讨论了特定的方法，并可丰富游戏体验，其中包括生成点、检测点以及路点。除此之外，相关内容还包含了群集对象，以及针对路径自动搜索和躲避障碍物的 Unity NavMeshes 应用。

第 9 章介绍游戏中的音效和分轨音乐，并通过脚本描述了运行期内的声音控制，即 Reverb Zones 以及 Audio Mixer。

第 10 章提供了游戏外部数据的增强方式，包括添加可再生的内容以及站点间的通信方法。另外，本章还涉及 Unity 云的自动化构建以及项目的构建方式。因此，读者可方便地应用在线版本控制系统，例如 GitHub。

第 11 章介绍游戏附加特性的添加方法（例如，慢动作和在线游戏的安全方案）。同时，本章还介绍了游戏代码的性能分析方法和改进措施。

第 12 章主要涉及 Unity Editor 中的设计方案。这里，Editor Extensions 涵盖了脚本和

多媒体内容，进而可与自定义文本、游戏参数的 UI 表示、Inspector 和 Scene 面板中的数据，以及自定义菜单和菜单项协同工作。这一类内容可简化工作流，并可使开发人员快速、简洁地实现其目标。

准备工作

读者阅读本书的全部准备工作则是下载 Unity 5.x 软件，对应下载地址为 <http://www.unity3d.com>。

对于第 4 章所讨论的多种方案，如果读者计划构建自己的图像文件，则需要使用到相应的图像编辑软件，例如 Adobe Photoshop，读者可访问 <http://www.photoshop.com> 进行下载；除此之外，读者还可使用 GIMP，其下载地址为 <http://www.gimp.org>。

本书适用范围

本书适用于打算深入学习 Unity 基本机制、多媒体特性以及涉及诸多游戏特性的既定方案的相关读者。相应地，程序员可关注 Unity 中的多媒体特性，而多媒体开发人员则可着重阅读脚本方面的内容。

同时，本书也适用于中、高级读者，其中不乏设计人员以及程序员，甚至包括开发团队的全体成员。

本书要求读者具备 Unity 的基本应用知识，并初步了解 C# 语言的程序设计方法。

相关标识

本书中加入了多种标识，进而可明晰对应方案的实现步骤，其中包括：

准备工作

该部分内容显示了对应方案的前期工作，描述了软件的设置方式，或者当前方案所需的基本设置项。

实现方式

该部分内容包含了方案作序的操作步骤。

工作方式

该部分内容通常包含了实现过程中的细节内容。

更多内容

该部分内容包含了某些附加信息，以使读者针对当前方案了解更多信息。

相关资源

针对相应方案，该部分内容提供了辅助链接以及其他有用的信息。

本书约定

文本中的代码、数据库表名称、文件名、文件扩展名、路径名、伪 URL、用户输入以及推特用户名采用如下方式表示：

“针对当前方案，对应字体位于 1362_01_01 文件夹中名为 Font 的文件夹内”。

URL 则采用如下方式描述：

“读者可访问 <http://docs.unity3d.com/Manual/UISystem.html> 以获取与 Unity UI 相关的更多信息”。

代码块则通过下列方式设置：

```
void Start () {  
    textClock = GetComponent<Text>();  
}  
  
void Update () {  
    DateTime time = DateTime.Now;  
    string hour = LeadingZero( time.Hour );  
    string minute = LeadingZero( time.Minute );  
    string second = LeadingZero( time.Second );
```



```
textClock.text = hour + ":" + minute + ":" +  
second;  
}
```

 图标表示较为重要的概念，而  图标则表示提示或相关操作技巧。

读者反馈和客户支持

欢迎读者对本书进行建议或意见反馈，以便我们进一步解读读者的阅读喜好。反馈意见对于我们来说十分重要，以便改进我们日后的工作。

对此，读者可向 feedback@packtpub.com 发送邮件，并以书名作为邮件标题。

若读者针对某项技术具有专家级的见解，抑或计划撰写书籍或完善某部著作的出版工作，则可阅读 www.packtpub.com/authors 中的 author guide 一栏。

我们将竭诚为每一名读者服务。

资源下载

读者可访问 <https://github.com/dr-matt-smith/unity-5-cookbook-codes> 下载完整的解决方案。

在下载的代码中，包含了详细的注释内容，以及各章中所提及的 Unity 项目。另外，其中还提供了本书中的截图/图表的彩色图像，以帮助读者进一步理解输出结果中的变化。

勘误表

尽管我们在最大程度上做到尽善尽美，但错误依然在所难免。如果读者发现谬误之处，无论是文字错误抑或是代码错误，还望不吝赐教。对于其他读者，以及本书的再版工作，这将具有十分重要的意义。对此，读者可访问 <http://www.packtpub.com/submit-errata>，选取对应书籍，单击 ErrataSubmissionForm 超链接，并输入相关问题的详细内容。经确认后，填写内容将被提交至网站，或添加至现有勘误表中（位于该书籍的 Errata 部分）。

另外，读者还可访问 <http://www.packtpub.com/books/content/support> 查看之前的勘误表。在搜索框中输入书名后，所需信息将显示于 Errata 项中。

版权须知

一直以来，互联网上的版权问题从未间断，Packt 出版社对此类问题异常重视。若读者在互联网上发现本书任意形式的副本，请告知网络地址或网站名称，我们将对此予以处理。

关于盗版问题，读者可发送邮件至 copyright@packtpub.com。

对于读者对笔者的爱护，我们表示衷心的感谢，并于日后向读者呈现更为精彩的作品。

问题解答

若读者对本书有任何疑问，均可发送邮件至 questions@packtpub.com，我们将竭诚为您服务。

目 录

第 1 章 核心 UI——消息、菜单、积分榜和计数器.....	1
1.1 简介	1
1.2 显示“Hello World”文本消息	5
1.3 显示数字时钟	8
1.4 显示倒计时数字计数器	11
1.5 生成渐变消息	13
1.6 显示具有透视效果的 3D 文本消息.....	15
1.7 显示一幅图像	18
1.8 创建 UI 按钮并在场景间切换	20
1.9 在面板中组织图像并通过按钮调整面板深度.....	25
1.10 显示交互式 UI 滑块的数据值	29
1.11 基于 UI Slider 的图形化倒计时计数器.....	33
1.12 表示对象相对位置的显示雷达	36
1.13 利用 Fungus 开源对话系统创建 UI	46
1.14 设置自定义鼠标指针图案	48
1.15 用于文本输入的 Input Field.....	51
1.16 基于 Toggle Groups 的切换按钮和单选按钮	55
1.17 本章小结	58
第 2 章 道具栏 UI.....	60
2.1 简介	60
2.2 创建简单的小型 2D 游戏——SpaceGirl.....	61
2.3 利用文本显示单一对象的拾取	65
2.4 利用图标显示单一对象的拾取道具	72
2.5 利用文本显示同一对象的多个拾取道具.....	75
2.6 利用多个图标显示同一对象的多个拾取道具.....	77
2.7 调整贴图尺寸并显示多个对象拾取道具图标.....	82
2.8 显示不同对象的多个拾取道具	84

2.9	使用 Dictionary<>和枚举类型	89
2.10	利用 UI Grid Layout Group 生成多个图标显示	94
2.11	本章小结	109
第 3 章	2D 动画	110
3.1	简介	110
3.2	水平方向上的精灵对象翻转	111
3.3	角色运动事件中的动画效果	112
3.4	创建 3 帧动画剪辑并生成连续动画	118
3.5	平台的跌落效果	120
3.6	利用精灵板序列创建动画剪辑	124
3.7	本章小结	127
第 4 章	创建贴图和材质	128
4.1	简介	128
4.1.1	创建并保存纹理贴图	129
4.1.2	概述	129
4.2	利用 Standard Shader 创建基本材质 (Specular 设置)	132
4.3	利用 Specular 和 Metallic 调整材质	140
4.4	向材质添加法线贴图	144
4.5	向材质添加透明度和自发射贴图	148
4.6	鼠标指针悬停时材质的高亮效果	153
4.7	向材质中添加细节贴图	156
4.8	材质透明度的淡出效果	160
4.9	在场景中播放视频	163
4.10	本章小结	166
第 5 章	相机	168
5.1	简介	168
5.2	生成画中画效果	169
5.3	在多部相机之间进行切换	174
5.4	根据屏幕内容生成纹理	178
5.5	相机的放大效果	182
5.6	显示小型地图	186

5.7	创建游戏中的监控相机	192
5.8	本章小结	196
第 6 章	光照效果	197
6.1	简介	197
6.2	使用光照和 cookie 纹理模拟阴天环境	200
6.3	向场景添加自定义反射贴图	205
6.4	生成激光瞄准器	208
6.5	利用反射探头实现周围物体的反射效果	214
6.6	利用 Procedural Skybox 和 Directional Light 设置环境	221
6.7	利用光照贴图和光探头照亮简单场景	224
6.8	本章小结	235
第 7 章	控制 3D 动画	236
7.1	简介	236
7.2	配置角色以及空闲动画	237
7.3	利用脚部运动和混合树移动角色	242
7.4	利用层和遮罩混合动画	249
7.5	将状态整合至子状态机中	255
7.6	通过脚本转换角色控制器	260
7.7	向动画角色中添加内容	265
7.8	通过动画事件投掷物体	268
7.9	向角色添加 Ragdoll 物理行为	272
7.10	角色的旋转和瞄准动作	277
7.11	本章小结	281
第 8 章	位置、运动和导航	282
8.1	简介	282
8.2	2D 对象的控制	283
8.3	3D 对象控制	287
8.4	计算最近（随机）生成点	292
8.5	在最近途径检测点重新生成对象	298
8.6	躲避障碍物时的 NPC NavMeshAgent	300
8.7	基于序列路点的路径	307

8.8	利用集群方式控制对象组的运动	315
8.9	本章小结	320
第 9 章	音频.....	322
9.1	简介	322
9.2	音调与动画速度之间的匹配	323
9.3	利用 Reverb Zones 模拟声学环境	328
9.4	避免音频剪辑重新播放	330
9.5	在对象自动销毁前终止音频播放	332
9.6	利用 Audio Mixer 添加音量控制器.....	335
9.7	利用 Snapshots 生成动态音轨	342
9.8	利用 Ducking 实现音频的平衡效果.....	350
9.9	本章小结	356
第 10 章	与外部文件和设备协同工作	357
10.1	简介	357
10.2	通过 Unity Default Resources 加载外部资源文件	359
10.3	通过互联网下载方式加载外部资源文件.....	362
10.4	通过手动存储文件加载外部资源文件.....	365
10.5	利用静态属性保存和加载玩家数据	367
10.6	利用 PlayerPrefs 保存和加载玩家数据	371
10.7	保存游戏中的屏幕截图	373
10.8	利用 PHP/MySQL 设置排行榜.....	376
10.9	从文本文件中加载游戏数据	380
10.10	利用 Git 版本控制和 GitHub 托管管理 Unity 项目	385
10.11	通过 Unity Cloud 在多台设备上发布.....	392
10.12	本章小结	396
第 11 章	利用附加特性和优化操作改善游戏	397
11.1	简介	397
11.2	暂停游戏	399
11.3	实现慢动作效果	403
11.4	避免游戏运行于未知服务器上	407
11.5	状态驱动行为	409

11.6	通过状态设计模式实现事件驱动行为.....	416
11.7	销毁死亡对象以减少对象的数量	421
11.8	通过禁用对象减少处于开启状态的对象数量.....	424
11.9	令对象处于非活动状态	428
11.10	利用委托机制和事件改善效率	431
11.11	利用协同程序定期执行某些方法	436
11.12	利用协同程序将较为耗时的计算分布于多个帧中.....	438
11.13	计算最小和最大帧速率并估算性能	441
11.14	Unity 性能分析器	443
11.15	DIY 性能分析机制	446
11.16	缓存 GameObject 和组件的引用	448
11.17	利用 LOD 改善性能	458
11.18	减少绘制调用的次数	462
11.19	本章小结	464
第 12 章	编辑器扩展	465
12.1	简介	465
12.2	通过自定义 Inspector UI 修改道具类型（以及参数）	465
12.3	利用单次单击操作添加 100 个随机位置处的副本.....	478
12.4	通过进度条显示完成比率	482
12.5	实例化道具对象	483
12.6	本章小结	488

第 1 章 核心 UI——消息、菜单、积分榜和计数器

本章主要包含以下内容：

- ❑ 显示“Hello World”UI 文本消息。
- ❑ 显示数字时钟。
- ❑ 显示数字计数器。
- ❑ 生成渐变消息。
- ❑ 显示 3D 透视文本消息。
- ❑ 显示一幅图像。
- ❑ 创建 UI 按钮，并在不同场景间进行切换。
- ❑ 在面板内组织多幅图像，通过按钮调整面板深度。
- ❑ 显示交互式 UI 滑块的具体数据值。
- ❑ 利用 UI 滑块显示图形化计数器。
- ❑ 对象相对位置的雷达显示。
- ❑ 利用 Fungus 开源对话框系统创建 UI。
- ❑ 设置自定义鼠标指针图案。
- ❑ 文本项的输入框组件。
- ❑ 基于 Toggle Group 的选择按钮。

1.1 简介

大多数游戏的娱乐性关键元素主要体现在视觉体验效果上，其重要组成部分则是用户界面（UI）。针对用户而言，UI 元素涉及多种方式，进而可与游戏进行交互，例如按钮、鼠标指针、文本框等，同时还包含了针对用户的实时显示信息，例如时间剩余、当前生命值、积分榜、生命值的剩余量以及敌方角色的位置。本章主要介绍 UI 方案，并展示了大量的游戏 UI 示例及其设计理念。

游戏的种类千变万化，但本章主要讨论两个关键目标。第一个目标是阐述如何创建各种 Unity 5 UI 元素，并在合适的地方与代码中的变量进行关联。第二个目标是分析 UI

元素的应用方式，进而探讨如何使 Unity 5 UI 控件传递特定的视觉效果和交互功能。

基本的 UI 元素可提供静态图像和文本，以使屏幕内容看起来更加有趣。通过使用脚本，可适当调整此类图像和文本对象的内容，进而更新玩家的积分榜，或者显示相关图像以通知玩家的生命值，等等。相比较而言，某些 UI 元素可具有交互特征，玩家可据此单击按钮，选择选项，输入文本内容，等等。同时，更为高级的 UI 元素还可进一步选取并计算与游戏相关的数据（例如剩余时间的百分比、敌方角色的消耗值、场景中关键 `GameObject` 的位置和类型，以及玩家位置和方向间的关系），随后可通过较为自然的图形方式显示这一类内容（例如进度条或雷达屏幕）。

具体而言，与 UI 开发相关的核心 `GameObject`、组件以及相关概念介绍如下。

- ❑ **Canvas**: 各种 UI 元素均为 `Canvas` 的子对象。在某一场景中可存在多个 `Canvas GameObject`。如果 `Canvas` 尚未存在，则在创建新的 UI `GameObject` 时，仅对其予以自动生成。此时，UI 对象为最新创建的 `Canvas GameObject` 的子对象。
- ❑ **EventSystem**: `EventSystem GameObject` 用于管理基于 UI 控件的交互事件。对于首个 UI 元素，`EventSystem` 通常会自动生成。
- ❑ **Panel**: 利用 UI `Panel`，UI 对象可采用分组方式予以显示（通过逻辑方式或者物理方式）。相应地，`Panel` 可扮演多种角色，例如，针对关联的控件组，可在 `Hierarchy` 中提供对应的 `GameObject` 父对象。这一类对象可针对屏幕上的图形关联控件显示可视化背景图像。另外，还可实现缩放和拖曳等交互行为。
- ❑ **Visual UI 控件**: 可视化 UI 控件自身包含了 `Button`、`Image`、`Text` 以及 `Toggle` 等。
- ❑ **交互式 UI 控件**: 作为不可见控件，需要添加至 `GameObject` 中，例如 `Input Field` 和 `Toggle Group`。
- ❑ **Rect Transform 组件**: UI `GameObject` 可存在于 2D 或 3D 这一类不同的空间中，进而实现相机的渲染行为。因此，UI `GameObject` 均设置了特定的 `Rect Transform` 组件，并针对场景的 `GameObject Transform` 组件提供了不同的属性（例如较为直观的 X/Y/Z 位置、旋转以及缩放属性）。与 `Rect Transform` 关系紧密的则是轴点和锚点，稍后将对此予以介绍。
- ❑ **Sibling Depth**: 对于某一 UI 元素，其自上而下的显示顺序在初始状态下可通过 `Hierarchy` 中的顺序加以确定。在设计时，还可采用手动方式进行设置，也就是说，可将 `GameObject` 拖曳至 `Hierarchy` 中的期望位置处。在运行期内，通过游戏或用户交互命令，则可向 `GameObject` 的 `Rect Transform` 传递消息，进而以动态方式调整其 `Hierarchy` 位置（即显示位置），稍后将对此加以讨论。

图 1.1 显示了 UI 控件的 4 种分类方式，且分别位于某一 `Canvas GameObject` 中，并