学习Cocos2D-X游戏开发的权威指南

Cocos2D-X技术专家鼎力推荐

Game Design and Develop

游戏设计与开发技术丛书

Cocos2D-X 游戏开发技术精解

刘剑卓 著

人民邮电出版社
POSTS & TELECOM PRESS

Cocos2D-X 游戏开发技术精解

刘剑卓 著

人民邮电出版社
北京

图书在版编目（CIP）数据

Cocos2D-X游戏开发技术精解 / 刘剑卓著. -- 北京
：人民邮电出版社，2013.6
（游戏设计与开发技术丛书）
ISBN 978-7-115-31484-0

Ⅰ. ①C… Ⅱ. ①刘… Ⅲ. ①移动电话机－游戏程序
－程序设计②便携式计算机－游戏程序－程序设计 Ⅳ.
①TN929.53②TP368.32

中国版本图书馆CIP数据核字(2013)第065978号

内 容 提 要

Cocos2D-X 是一款支持多平台的 2D 手机游戏引擎，支持 iOS、Android、BlackBerry 等众多平台。当前，很多移动平台流行的游戏，都是基于 Cocos2D-X 开发的。

本书详细介绍如何使用 Cocos2D-X 引擎开发自己的移动平台游戏。全书共 15 章，主要内容包括：Cocos2D-X 引擎简介；如何建立跨平台的开发环境；引擎的核心模块——渲染框架；如何实现动态画面和用户交互；二维游戏中背景的实现方法和技术；Box2D 物理引擎；如何掌握声音引擎的用法；Cocos2D-X 引擎的文件操作模块和内存管理机制；各种各样的粒子效果；如何掌握利用 Lua 脚本制作游戏的能力；Cocos2D-HTML5 引擎版本；引擎的附加功能等。最后，本书和读者一起展望了 Cocos2D-X 引擎的未来。

作为 Cocos2D-X 的权威指南，本书得到了 Cocos2D-X 引擎开发者的建议以及指导。本书适合对 Cocos2D-X 感兴趣的以及有志于学习和从事移动平台游戏开发的读者阅读参考。

◆ 著　　刘剑卓
责任编辑　陈冀康
责任印制　程彦红　焦志炜

◆ 人民邮电出版社出版发行　　北京市崇文区夕照寺街 14 号
邮编　100061　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京艺辉印刷有限公司印刷

◆ 开本：800×1000　1/16
印张：26
字数：747 千字　　2013 年 6 月第 1 版
印数：1－3 000 册　　2013 年 6 月北京第 1 次印刷

定价：69.00 元（附光盘）

读者服务热线：(010)67132692　印装质量热线：(010)67129223
反盗版热线：(010)67171154

前　言

将此书献于那些想要制作快乐的人！

凭借 Cocos2D-X 强大丰富的功能、简单易用的特点，读者成为一个优秀的游戏开发者将是轻而易举的事情。同时，网上商店也为开发者提供了面向全球用户的开放市场。此时，正是读者尽显才华、影响世界的机会。所以无需等待，尽快开始神奇而愉快的游戏开发之旅吧！接下来，为了方便读者对本书中的内容有一个全面的认识，这里将按照章节的顺序进行概要介绍。

本书的章节安排

本书作为 Cocos2D-X 的权威指南，得到了引擎开发者的建议以及指导。虽然 Cocos2D-X 是来源自 Cocos2D-iphone 的改编版本，但是其源代码是完全诞生自中国人之手。Cocos2D-X 还是一款免费开源的引擎。

Cocos2D-X 引擎支持多达 6 个平台，其中包含有 iOS、Android、Web、PC、Bada、BlackBerry 等。因为考虑到技术国际化的原因，所以在书籍和技术文档方面很少有中文版本。所以本书存在的价值就是为了满足中文开发者的需求，将移动跨平台中最好的二维游戏引擎 Cocos2D-X 介绍给大家，让更多的人进入到游戏制作领域，立志于游戏产业的发展。

第 1 章　Cocos2D-X 引擎的介绍

作为本书的第一章，将会为读者介绍游戏开发的基础知识。引擎在游戏开发中为什么不可替代？一款引擎需要具备哪些功能？Cocos2D-X 游戏引擎的特点是什么？相信读者在熟悉本章的内容后，也会认同本书的决定，而选择 Cocos2D-X。另外，本章的内容还会涉及引擎的安装、组成以及成功游戏的展示。

第 2 章　Cocos2D-X 引擎的开发环境

本章将会指导读者建立跨平台的开发环境，以 Android、iOS、PC 平台为主，其他平台为辅，通过详尽的文字以及图示逐步指导开发者搭建开发环境。本章最后为读者展示一些引擎中的实例项目。示例项目中的各种功能，将会在后续各章中逐个介绍。

第 3 章　引擎的核心——渲染框架

本章将会带领读者深入引擎内部。渲染框架作为引擎的核心模块，这是每个开发者必须要掌握的核心功能。要想做好游戏，本章所介绍的内容将是关键。作为重点知识，本章的内容并不少。为了让读者熟悉每一个知识点，本章中除了文字描述，还展示了源代码、图示以及示例程序。

第 4 章　动作功能

本章将会按照从简单到复杂的安排，为读者介绍动作功能。读者不仅会知道每个动

作的用法，还会领略到其类的组成方式。动画几乎是二维游戏的画面主题。在本章的末尾，读者还将会接触到几款与动画配套的编辑工具。

第 5 章　用户交互

本章中，读者将会掌握引擎中所提供的用户交互功能。首先，我们需要了解移动平台所能接受的玩家操作有哪些。然后，明白引擎是如何接受用户操作事件并及时地反馈给开发者。

第 6 章　游戏背景

在本章中，读者将会接触到二维游戏中背景的实现方法。首先，我们将介绍游戏产品中常见的背景效果以及它们的实现技术。然后，通过一款知名的 2D 背景编辑器，制作一张游戏的背景地图。最后会将制作的背景地图，放置到示例程序进行展示。

第 7 章　物理模拟与碰撞检测

Cocos2D-X 引擎中包含了两款优秀的物理引擎：chipmunk 和 Box2D。它们都是来自于第三方开发者的开源分享。本章将着重为大家介绍 Box2D 物理引擎。读者可以用它模拟现实的真实世界，让玩家体会到变化非凡、不可预知的感觉。

第 8 章　游戏中的声音

在本章中，读者将会接触到引擎中的声音引擎。读者将会通过示例项目来掌握声音引擎的用法。利用背景音乐和音效将会提升游戏的品质，吸引更多的玩家。无论怎样，一款游戏产品是绝对少不了声音的配合的。

第 9 章　文件操作模块

文件存储与读取，也是大多数游戏需要的功能。Cocos2D-X 的跨平台特性，很好地解决了不同平台之间的差异。开发者无需关心各个平台的实现细节。通过简单易用的程序接口，就可以实现多平台的数据读取与保存。

第 10 章　内存管理机制

内存管理机制常常被开发者称为是一种高级技巧。本章就是要让读者深入浅出地理解 Cocos2D-X 引擎的内存管理机制。本章中将会涉及几个内存管理的技术，比如引用计数的原理以及垃圾回收机制。

第 11 章　粒子系统

粒子系统能够为游戏产品带来变化多端、绚丽多彩的画面效果。它已经成为了如今游戏开发中必不可少的环节。粒子效果常常在游戏产品中起到了画龙点睛的作用。在本章开始，将介绍粒子系统的来历以及实现效果。

第 12 章　Lua 脚本语言

脚本语言通常是引擎功能完善之后，为开发者准备的一种更为简单的编程方式。Lua 脚本语言是游戏制作中非常流行的一种脚本语言。脚本习惯地被开发人作为引擎编码的“副手”，有时会被提供给设计人员来使用。

第 13 章　Cocos2D-HTML5 引擎版本

本章的内容有些特别，它并不属于 Cocos2D-X 引擎。Cocos2D-HTML5 是与 Cocos2D-X 等同的一个版本。正是因为 HTML5 的火爆，引擎开发团队才推出了此版本。HTML5 被许多游戏业内人士誉为了明日之星。它代表了未来游戏发展的趋势。

第 14 章　引擎之外的附加功能

本章将会为读者介绍一些引擎之外的附加功能。首先是游戏的联网功能，主要因为 Cocos2D-X 引擎更倾向于单机类游戏产品。虽然也有开发者用其制作网络游戏，但是网络游戏制作的重点更在于服务器端。所以网络功能成为了引擎中附带的功能。

第 15 章　Cocos2D-X 引擎的未来

到本书最后的一章，读者应该已经领略了 Cocos2D-X 引擎的魅力。在逐步学习和实践的过程中，或多或少地大家都会遇到一些不顺手的地方。这些宝贵的经验，正是引擎开发者们希望看到的改善建议。

本书适合的读者

- ✧ 喜欢游戏并怀揣梦想的有志青年，本书将会为你提供影响世界的机会。
- ✧ 想成为移动平台游戏开发者的人，本书中的游戏制作技术将会帮助你找到一条捷径。
- ✧ 想抓住移动互联网机遇成就一番事业的人，书中的内容就是你步向成功的台阶。
- ✧ 具备其他平台游戏开发经验的人，阅读本书之后将能轻松地跨越各个移动平台，谋取更多的机会。
- ✧ 智能手机设备的用户，想制作一款自娱自乐的游戏产品。
- ✧ 移动领域的游戏开发者，本书将会为你介绍一款功能丰富、简单易用的跨平台游戏引擎。

致谢

像以往一样，在此我由衷地感谢许多人慷慨相助！

最需要感谢的就是 Cocos2D-X 引擎的开发团队，尤其是王哲（Walzer）给予的帮助和指导。没有免费开源的精神，我们是不会享受到如此一款优秀的游戏引擎。当然，也要感谢那些来自全球各地的开发者，是他们善于分享、乐于解答的共同努力，才造就了引擎的繁荣场面。

感谢父母，谢谢您们的养育之恩，让我能够有今天的成就。希望您们身体健康、心情愉悦地度过今后的每一天。

还要特别感谢的就是刘霞，她是我身边的督促者。正是因为她严谨求实的敬业精神，成为了本人写书过程中的源动力。没有她的协助，接下来的内容显然不会存在。另一方面，她是我生活中的朋友，一个热爱生活、向往美好的女孩。

本书由刘剑卓组织编写，同时参与资料收集、代码测试和文字整理的还有胡奇峰、张国强、张昊、李峥、孙飞、王朋章、刘波、栗菊民、文奇、吴琪、席国庆、臧勇、罗思红、郭玉敏、贺道权、胡斯登、江成海、王新平、利建昌、王石磊、张金霞、张昆、尹继平、项宇峰，在此一并表示感谢！

刘剑卓

2013 年 1 月 10 日

目　录

第 1 章　Cocos2D-X 引擎的介绍

如果你梦想着创造充满了价值和理念的世界，那么本书将会介绍一个帮你实现梦想的绝佳途径。

游戏正在改变世界，改变人们的生活。它甚至被赋予了神圣的使命——重塑人类积极的未来。在游戏当中，人们可以感觉到平等、充实和愉悦。游戏让人们的交际更加真实、深入和多元。游戏让娱乐业有更大的发展空间，有更多的经济收益，有更具想象力的挑战。通过本书的学习，读者将会掌握制作游戏的本领。制作游戏的过程，充满了兴奋和喜悦。相信阅读本书的读者中，每一个人都是喜欢游戏的，也喜欢创造和编写游戏。

本章作为开篇的第一章，将会为读者介绍游戏引擎的概念。作为游戏制作的核心，引擎发挥着极其重要的作用。在众多游戏引擎当中，我们选择了一款跨平台支持的 2D 游戏引擎——Cocos2D-X。它绝对是移动平台最火热的游戏引擎。通过介绍其形成的历史、背景以及组成结构，让大家对游戏引擎，尤其是 Cocos2D-X 有整体的认识，知道一款游戏引擎需要具备哪些功能，Cocos2D-X 游戏引擎的特点在于哪里，还会涉及一些引擎的安装与使用指南。最后，必须要说明的是，Cocos2D-X 是一款免费开源的游戏引擎。

1.1　何为游戏引擎

什么是游戏引擎？提到“引擎”一词，很容易想到汽车、轮船以及飞机。这些由机器驱动的交通工具，其能量的来源就是引擎。游戏引擎和汽车引擎在概念上是一样的，都是驱动整体运转的核心部件。游戏产品的核心就是引擎，它是每款游戏的运行基础。引擎的好坏直接影响着游戏的品质。

1.1.1　游戏的核心——引擎

引擎的概念应该是来自机器制造。它通常作为机器设备的动力核心，所以有人将引擎称作发动机。引擎最大的作用，就是能够为依附于它的部件提供能量。引擎对于机器的重要性不言而喻。如果以人体为比喻的话，引擎就好比心脏。试想一下没有了心脏，人就无法存活。同样，汽车没有了引擎，也只能依靠人推马拉。不过，就像心脏可以移植，引擎也是可以更换的。一辆好车，一定要配有一个强劲的引擎。图 1-1 所示的就是一款跑车的引擎。单看其十缸的结构就能知道这是一款性能卓越的引擎。

图 1-1 所示的引擎，是人类工业化艺术的最好呈现。现在汽车引擎的鼻祖是工业革命时人类发明的内燃机。如今的汽车引擎都是以燃油为主，也有使用混合动力的。由内燃机发展至汽车引擎，人们大约用了二十年的时间，在二十年间经历了无数次的技术革新。其实每一次技术的变革都不是突然发生、毫无根据

的，而是制作行业发展到一定规模才出现的。因此引擎并不是上帝赐给人们的礼物，也不是来源于自然的神奇生命，而是人们知识的结晶。

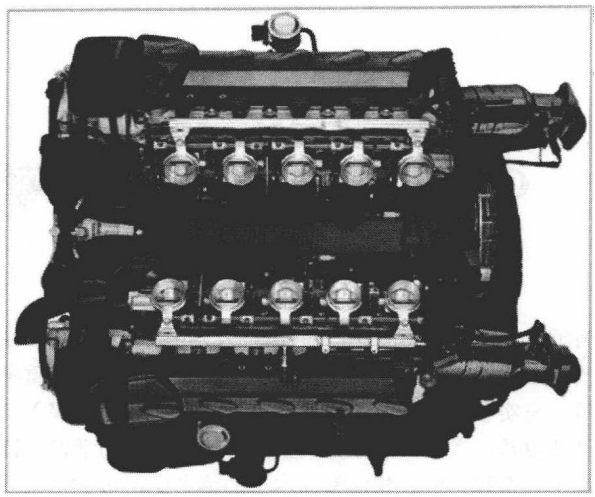

图 1-1　汽车引擎

对于游戏引擎，也是同样的道理。引擎也是一款游戏的核心，它为游戏中其他部分提供了功能服务。没有了强劲引擎的支持，游戏将会逊色许多，甚至遭到玩家的遗弃。许多老旧游戏产品的命运就是如此。从世界上第一款游戏算起，至今也有三十年的时间了。在这期间游戏引擎的发展，也是逐渐形成、不断地推陈出新的过程。

说明：在电子游戏初期，并没有核心引擎。引擎是在游戏技术发展了一段时间后产生的。

综上所述，无论是汽车引擎，还是游戏引擎，其制作技术仍然在不断地推陈出新。从早期简单、局限、低效，到如今丰富、开放、高效，经过了许许多多开发者的辛勤工作。更少的损耗、更高的性能，这正是引擎设计与制作一直追求的目标。

虽然引擎是一辆汽车的重要部件，但它并不是整辆汽车。事实是引擎是一个独立的部件，是能够从汽车里面取出的发动机。工人可以建造另外一个外壳，再将引擎安装入内，这就产生了一辆新的汽车。所以说引擎作为核心部件，是可以更换，也可以重复使用的。

游戏中的引擎也是如此。作为游戏的核心，它并不是游戏的全部。一款优秀的游戏引擎经常被用来制作很多游戏产品。这些产品在市场上将单独出售。因此为了保证引擎的通用性和标准化，引擎需要具备让游戏运行的基本功能，但不能含有游戏特有的功能。所以一款好的游戏引擎应该可以轻易地更新换代，同时又可以重复利用。

说明：有些引擎为了宣传效果，会使用和游戏相同的名字，比如 Quake，致使许多人会混淆游戏引擎和游戏本身。

1.1.2　引擎的特点

在前面的介绍中，读者知道了引擎的作用。在游戏和汽车中引擎有很多类似的概念。但是游戏引擎也有很多自身的特点。下面我们通过游戏引擎产生的过程，来了解它的特点。

正如前面所说，游戏引擎并不是一开始就有的，也是通过了技术发展与革新的。游戏引擎的概念首次出现在 1990 年 id Software 发行的《Doom》游戏当中。为了更好地展现游戏魅力，《Doom》开发者约翰·卡马克将其建立在一个性能优秀的内核之上。当时的游戏内核具备三个主要功能：游戏中画面的渲染、物体之间的碰撞以及音乐音效的播放。

说明：正是因为上述的原因，约翰·卡马克被大家习惯地称为游戏引擎之父。

卡马克意识到内核所提供的功能完全可以脱离游戏而独立存在，内核可以重复利用，作为今后其他游戏产品的内核。于是，他就把游戏中给玩家带来直观感觉的内容剥离掉，其中包括：图片数据、逻辑运算、游戏规则等，那么余下的内容就是重复利用的核心部分。这个核心部分后来就被定义为“引擎”。当然，具体工作可不是两句话这么简单。从那之后，由于引擎的诞生，游戏制作领域就进入了一个崭新的时代。开发者开创了一种全新的游戏开发模式：游戏“引擎”开发。

引擎开发方式逐渐取代了原本一体化的开发方式。早期游戏制作时，一款游戏由许多的模块构成，它们紧密地结合很难单独拆分出来。这就导致原本的游戏产品除了升级优化，推出后续版本之外，别无他用了。当需要准备制作新的游戏时，并没有多少可以再次利用的功能。而引擎化的开发方式，则克服了低利用率的缺陷，充分利用开发者的工作成果。这种模块化、封装化以及良好扩展性的游戏开发方式，逐渐成为了主流。

如今开发者几乎都会使用引擎来制作游戏，很少有使用一体化开发游戏方式的了。那么选择一款优秀的游戏引擎，就成为游戏制作之前的首要目的了。怎样才算优秀的游戏引擎呢？作为第一次接触游戏开发的读者来说，很难作出决定。下面来介绍一些游戏引擎的特性作为参考。

说明：有些独立游戏开发者为了自娱自乐，仍然热衷于一体化开发方式。

（1）稳定压倒一切！游戏引擎首要标准就是稳定，必须能够保证游戏产品的顺畅运行。引擎为玩家提供稳定连续的游玩体验算是最本质的要求。如果游戏中经常出现中断、崩溃或者画面错乱的问题，对于玩家的体验感损失非常大。

另外，从游戏开发的角度来看。通常引擎的开发者与游戏制作者不是同一人，游戏制作者并不熟悉引擎的编码。由于引擎大多进行封装，隐藏了内部的逻辑与调试信息，如果一个问题是由引擎内部产生的话，接下来的修补工作将是很难进行的。所以，稳定性是游戏引擎制作的首要标准。就算性能再强劲的引擎，如果问题重重的话，那也是无人问津的。

（2）性能是引擎好坏的关键。引擎性能一方面是指游戏运行时的流畅度，实际的技术参数就是指每秒屏幕的刷新率。另一方面，就是指引擎能够承载的运算量，比如是否可以进行复杂的物理模拟运算，能够支持多少个画面层次。它通常用来衡量一个引擎的好坏。这就好比汽车引擎的功率，游戏配备了性能强劲的引擎，就能够表现给玩家更多、更丰富的内容。如果把引擎比作心脏的话，一款好的游戏必须要拥有强劲、节奏均匀的心跳。性能好坏，通常是开发者选用引擎的标准。

（3）好的引擎就要有丰富完善的功能。随着玩家对游戏的需求和硬件设备性能的日渐提高，现在的游戏引擎不再是一个简单的渲染引擎。它需要涵盖许多丰富的功能，比如二维图形绘制、三维图形绘制、多声道处理能力、人工智能、物理碰撞、文件存取、社区分享、UI 界面、动画视频等。丰富的功能为开发者提供了更多的选择，更能发挥游戏制作者的创造力。游戏引擎就像是一把多用途的瑞士军刀，或者是机器猫的口袋，开发者有了它能够应对各种各样的问题，满足玩家千奇百怪的口味。

（4）简单易用才是引擎发展的王道。引擎是被用来制作游戏的工具，那么必然要易于使用。钻木取火是非常麻烦的一件事，但是有了打火机后只需要轻轻一按，就能获得光明和温暖了。引擎也要发挥同样的

作用。因为开发者使用引擎时，也要编写部分的代码，所以那些具备了简洁高效的程序接口、完善的技术支持文档的引擎将会更讨人青睐。

引擎要具备丰富的功能，同时还要简单易用。这算是所有消费者的心理需求。现代科技就是把一些复杂的事情简单化，游戏引擎也是如此。就好比按一下开关，洗衣机就会自动进水、洗涤、甩干等。对开发者来说，也许只需一行简单的命令，就可以让游戏中的人物完成跳跃、奔跑、站立的动作。引擎会把复杂的图像算法、物理模拟等功能封装在模块内部，对外提供的则是简洁高效的程序接口。这样有助于游戏开发人员迅速上手，这一点就如各种编程语言一样，越高级的语言功能越丰富，越容易使用。

另外，引擎也会为非编程人员提供可视化的编辑器或者第三方插件。实际开发过程中，只依靠引擎制作游戏是不够的，制作人员还需要各类工具来提高开发速度。所以引擎需要具备可视化编辑器，包括场景编辑、模型编辑、动画编辑、精灵编辑等。编辑器提供了所见即所得方式，不仅会加快制作的速度，也能保证游戏的品质，减少开发人员的错误。这些编辑器或者工具，不仅仅是为编程人员准备的，而是所有游戏参与人员都有可能使用它们。

说明：有些引擎本身就是一个集所有功能于一身的编辑器。

第三方插件则是另一种形式。它们通常是一些软件的辅助工具。例如游戏开发中美术人员经常会用到的第三方软件 3DS Max、Maya、Photoshop 等等绘图工具。引擎中提供了与其对接的插件。在第三方插件中，美术人员所制作的游戏资源，不再需要其他工具的辅助，直接可将资源转换为引擎需要的格式。当美术人员用编辑器调整人物动画时，可发挥的余地就更大，做出的效果也越多。这样就节省了开发人员的学习成本。

（5）跨平台特性是引擎的趋势。随着越来越多的电子设备融入人们的生活，为了给用户更加全面的体验，游戏引擎跨平台特性也逐渐被开发者重视。引擎能够帮助开发者实现游戏产品跨越不同平台带来的差异。开发者只需编写一套代码，就可以在多个平台运行。这无疑会节省游戏开发的成本，缩短周期。另外，更多平台支持，就意味着更多用户选择，也会为开发者带来更多的收益。

经过上面的介绍，读者对引擎的好坏也有评判的标准了。不过，世界上没有完美的事物，所以也没有最好的游戏引擎。对于开发者来说，需要做的是选择一款最适合的引擎。接下来，为读者介绍一些知名的引擎。虽然本书的主题已经确认，将会以 Cocos2D-X 引擎为主，不过，可以扩宽知识内容，有对比的学习，也未尝不是一件好事。

1.1.3　知名的引擎介绍

从第一款引擎的出现到现在已经过去了二十多年。这期间出现很多值得称赞的引擎。这些引擎各具特色。开发者使用它们也制作了许多为玩家所喜爱的经典游戏。限于篇幅，很难逐个地为读者详细的介绍。但是读者也不能作为井底之蛙，只看到眼前一个游戏引擎，毕竟 Cocos2D-X 引擎与那些老牌游戏引擎比起来，只能算是新起之秀。单就游戏技术而言，移动平台依旧是在追随其他游戏平台已经实现的内容。

说明：最领先的游戏技术大多应用在个人电脑以及家用机平台。

从游戏发展的历史来看，移动平台的游戏产品在全球游戏产业中只能算是冰山一角。但是，它作为新型平台，蕴涵爆发的能量，其快速扩张的程度已经占据了一定的市场份额。几乎所有的知名游戏引擎厂商都十分注重甚至已经进入了移动游戏市场。短短几年的发展，iOS 设备全球扩张的速度，足以威胁到了原本的游戏产业。我们都知道 iOS 设备推出之初，并不是为游戏打造的。但是根据 AppStore 的数据显示，人们最热衷的依然是游戏类的产品。据欧美国家统计，今年的圣诞节孩子们最希望的礼物，已经从家用游戏

机转到 iPad 了。

表 1-1 列出了一些知名的游戏引擎。它们是众多游戏引擎中的一部分。按照粗略的估算，在游戏产业中至少存在上百款游戏引擎。正是这些引擎推动了今天游戏产业的繁荣。大体上游戏引擎可以看作两类：2D 和 3D。这只是根据其产出的游戏产品来划分的。按照比例来看，3D 游戏引擎占据了主流地位。这主要是因为 3D 技术的复杂度，导致一般开发者很难独立制作，所以会借助成熟的游戏引擎。而 2D 引擎因为技术比较成熟、容易实现、制作周期短，所以行业内的成熟产品不多。很多的 2D 引擎是开发团队或者公司内部使用的。

表 1-1 知名游戏引擎

引擎名称	类型	技术	适用平台
虚幻	授权	3D	多平台
Unity	授权	3D/2D	多平台
Cry Engine	授权	3D	PC 和家用机
Cocos2D-X	开源	2D	手持设备
极品飞车系列	自主	3D	多平台
IW	自主	3D	多平台

注意：有些 2D 引擎也是使用 3D 技术来渲染画面的，大部分的 3D 引擎都可以用来制作 2D 游戏。

在表 1-1 中，引擎按照类型来区分，大致可以分为三类：授权、开源和自主。这三类涵盖了市场上所有引擎的种类。不同的开发者将会有各自选择的类型。下面就按照分类，依次为读者介绍每种引擎的特点。

1.1.4 引擎的分类

授权类的引擎，是指具备深厚技术实力的引擎厂商开发，通过授权的方式出售给游戏制造商的引擎。授权类的引擎主要以 3D 的类型为主。比如虚幻、Unity3D、Torque、CryEngine 等都是非常不错的授权 3D 引擎。这些引擎原本都是针对个人电脑以及家用游戏机平台的，随着移动平台的全球火爆，一些引擎也纷纷推出的移动平台的版本。因为移动平台的游戏规模小，所以其价格也相对优惠。例如虚幻引擎（UDK）针对开发者前期免费使用，只有在游戏产品售出超过一定规模后才需支付费用。Unity3D 引擎的 iOS 平台版本只是销售 1500 美元而已。

3D 引擎的制作难度以及开发周期都要远远高于 2D 引擎，所以对于一般开发者来说直接购买成熟引擎要比自主开发更适合一些。授权引擎中几乎包含了游戏开发的所需全部内容：渲染、编辑工具、物理体系、人工智能、网络通讯等。引擎框架严密而稳定，功能丰富易用，相关的配套工具以及技术文档也十分全面。毕竟开发者支付了费用，所以能够享受到周全的服务。况且，使用授权引擎能够降低开发失败的风险，因为大多数的授权引擎，已经被用来开发了很多游戏产品。

注意：在购买授权引擎时，需要明确购买的内容。很多的授权引擎是按照开发者购买的内容来计算价格的。

虽然授权引擎提供了完善而丰富的游戏功能，但是因为其需要支付一定的费用，所以对于资金有限的开发者来说还是不太合适。除了需要支付费用之外，授权引擎还有下面三点不足。

（1）授权的引擎中的源代码和工具通常是不开放的。当开发者遇到一些致命问题时，自己是很难解决的。唯一可行的办法，就是依靠引擎提供方的技术支持。这可是一项耗时耗力的沟通工作啊！尤其是对于国内的开发者来说，除了语言上的交流困难之外，还会因为距离、时差等因素导致沟通周期很长。

（2）由于引擎提供的功能是早已定制好的，虽然可以扩展相关的模块，但也是在限定范围内，如果开发者需要为自己的游戏加入新的特有功能，这也将会是一件难事。同时，游戏引擎的最新版本是掌握在厂商手中的，开发者很难及时地将一些新的技术应用在游戏项目之中。

（3）最后，由于引擎并不属于游戏制作商，导致游戏产品本身缺乏核心的价值。再加上，同一商业引擎可能已经被应用到许多的游戏当中。这就导致游戏之间同质化的现象是十分明显的。游戏的特色就不能依靠引擎来体现。

总之，如果开发者没有时间以及能力开发一个新的引擎，就不要介意购买第三方的引擎！读者需要明确，购买一个商业引擎并不意味着在游戏制作的过程中不再需要技术人员了。虽然引擎完成了许多的编码工作，但是游戏中的逻辑或者规则，还是需要编写代码。编码可能是某种脚本语言，也可能是某种编程语言。授权引擎适合那些想要一个成功的、现成的解决方案的开发者。这样的话，他们就能够把精力集中在游戏可玩性和创造内容上。授权引擎几乎用最优化的方法解决了游戏开发中的技术问题。它可以快速帮助开发者完成游戏产品。

注意：每一款商业引擎都存在局限性，可能是游戏类型的限制，也可能是游戏平台的限制，开发者在购买之前需要仔细考虑。

自主研发类的引擎通常是一些游戏制作公司出于制作游戏的需求而开发的引擎。其实授权引擎的前身大都是自主研发类的引擎。引擎制作商最初的引擎都是用来制作自己发行的游戏，在游戏产生良好市场效益并获得了开发者和玩家认可后，才转为商业引擎的。这些引擎多半是因为一款成功游戏而被人们所熟知。自主研发引擎是最传统的游戏开发方式，也是使用最为广泛的。比如 id Soft 知名的 Doom 引擎，就是用来制作 Doom 游戏的引擎，在技术成熟和产品成功后，就成为了授权的商业引擎。

说明：Doom 引擎现在已经是免费的引擎，曾用于《Doom》和《Doom II》。Doom 3 引擎还在销售中。

自主研发引擎的方式，适合具备游戏开发经验，有一定技术实力的开发者团队或者公司。自主研发引擎需要时间、人力和资金投入，这对于一些独立游戏开发团队是一笔不小的前期开支。不过，自主研发引擎获得成功，这将是一个游戏团队最核心的价值。游戏开发者可以按照自己的意图去构建引擎来制作游戏。引擎相关的制作工具，也可按照开发团队的工作流程以及人员配备来制作。自主研发的方式可以让开发者充分发挥自由，量身定做属于自己的游戏引擎，不仅在技术具有自主的产权，也可以打造出独具特色的游戏，令他人无法模仿。

自主研发的方式，意味着游戏制作要从最基础的地方开始。这是一个从“无”到“有”的制作过程。毫无疑问，自主研发的方式是游戏开发周期最长的，失败的风险也是最大的。作为自主研发引擎的开发者，需要有足够的耐心和不屈的毅力才能坚持到最后环节。只有将游戏产品推向市场，获得玩家的认可才有可能看到回报。所以如果开发者选择这样的方式时，就要有足够的信心和毅力，做好准备，因为在漫长的游戏开发阶段是没有利益回报的，期待一举成名的美好夙愿是行不通的。

注意：开发者不妨选择逐步去完善游戏引擎，先推出初期版本而不是一气呵成地去打造完美无缺的引擎。

开源引擎是通过将引擎的源代码在网络分享，以不谋取任何商业利益为基础，以公开发布的方式免费