

游戏编程大师系列

Visual C++ 角色扮演 游戏程序设计



本书内附
范例光盘

坂本千寻 著 博硕文化 译

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

北京市版权局著作合同登记号：01-2003-5302 号

版 权 声 明

本书中文简体字版经 Japan UNI Agency, Inc., 由 Softbank Publishing, Inc. 授权中国铁道出版社出版 (2003), 任何单位或个人未经出版者书面允许不得以任何手段复制或抄袭本书内容。

ロールプレイングゲームプログラミング

Copyright©2003 by 坂本千寻

Simplified Chinese translation rights arranged with Softbank Publishing, Inc.

Through Japan UNI Agency, Inc., Tokyo

图书在版编目 (CIP) 数据

Visual C++ 角色扮演游戏程序设计 / (日) 坂本千寻著; 博硕文化译. —北京: 中国铁道出版社, 2004.9

(游戏编程大师系列)

ISBN 7-113-06154-0

I. V... II. ①坂... ②博... III. 游戏-C语言-程序设计 IV. TP312

中国版本图书馆CIP数据核字 (2004) 第094316号

书 名: Visual C++ 角色扮演游戏程序设计

作 者: 坂本千寻

出版发行: 中国铁道出版社 (100054, 北京市宣武区右安门西街8号)

策划编辑: 严晓舟 郭毅鹏

责任编辑: 苏茜 赵汶 魏巨兵

封面设计: 清风书坊 江涛

印 刷: 北京鑫正大印刷有限公司

开 本: 787×960 1/18 印张: 21.75 字数: 430千

版 本: 2004年11月第1版 2004年11月第1次印刷

印 数: 1~5000册

书 号: ISBN 7-113-06154-0/TP·1308

定 价: 43.00元

版权所有 侵权必究

凡购买铁道版的图书, 如有缺页、倒页、脱页者, 请与本社计算机图书批销部调换。

出版说明

本书作者特别针对游戏程序设计所需要的特有思维与技巧，结合本身经验将这些零散的数据整理成了本书，希望能对游戏开发的初学者有所帮助。本书以 VC++ 配合 MFC 类库进行专业游戏设计，从制作游戏雏形开始由浅入深进行介绍，并介绍图型的特效处理等。适用于想要学习开发游戏的读者，相信读者可以通过本书从中学到游戏的设计思维与技巧。

随书所附光盘中文文件夹的内容为书中所涉及的相关范例。

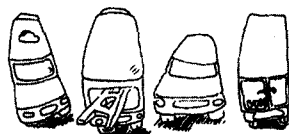
本书由Softbank Publishing,Inc.通过Japan UNI Agency,Inc.提供版权，并由博硕文化翻译，由中国铁道出版社计算机图书中心审选。费广正、韩红雷、湛永松、房雅丁、孙庆杰等同志完成本书的整稿工作。

编 者

2004 年 8 月

目 录

Chapter 1 写一个角色扮演游戏	1
何谓角色扮演游戏 (RPG)	2
● 电脑游戏的RPG	2
RPG 分类	3
● 故事性	3
● 移动	5
● 战斗	6
● 系统概要	7
写 RPG 时的软硬件需求	9
● 程序设计的软硬件需求	9
● C/C++编译器	9
最新版的 Service Pack	10
● 硬件环境	10
● 其他硬件需求	11
● 收集数据	12
● RPG设计相关工具软件, 如地图编辑器 (Map Editor) 等	12
Chapter 2 游戏系统的设计	13
RPG 系统	14
● 外观 (画面显示方法)	14
● 故事铺陈	15
● 移动 (非战斗状态)	16
● 战斗	17
● 单次移动距离 (战斗状态时)	17
● 窗口大小	18
● 游戏系统内容总整理	19
RPG 的必备功能和设计	19
● RPG部分	19
地图显示	19



CONTENTS 目录

移动	22
战斗	24
CPU 端的动作循环	24
参数	25
● 事件部分	26
游戏的发展	27
Chapter 3 Windows程序设计	29
游戏的必备功能	30
● 画面显示	30
● 其他功能	31
开启窗口	32
● 结合两端的桥梁部分	32
● Windows程序的流程	32
● 基本类	33
应用软件	33
● 产生窗口与消息处理	39
窗口	40
消息处理	51
显示 CG	52
● BMP格式的概要	52
● DIB的架构	53
显示 C G 的应用软件	62
● 设定窗口样式	67
● 产生DIB及读入CG图文件	69
● 显示DIB	70
CG/Sprite 的重叠显示	71
● 制作合成用的类	71
复制用的函数	75
合成用的函数	75
● DIBSection	76

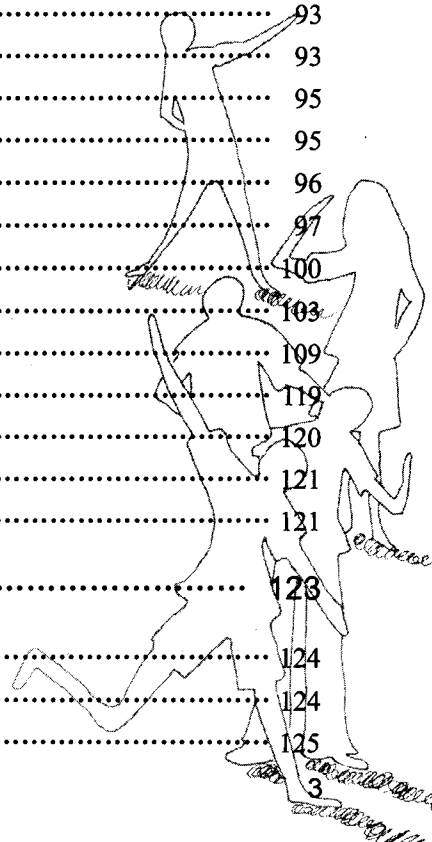
● DIBSection的绘制	81
贴图零件 (sprite) 显示	82

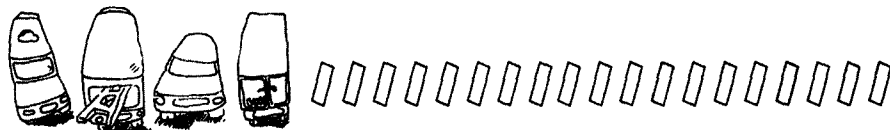
Chapter 4 显示地图和人物 87

地图坐标系	88
● 从地图坐标转换成画面坐标	88
● 从画面坐标转换成地图坐标	90
利用渐变色地图	90
坐标转换的计算式	91
解法(1): 实数运算	92
解法(2): 应用移位 (shift)	92
解法(3): 加上 offset (位移偏差)	92
解法(4): 分类处理	93
Sprite 显示	93
● Sprite的显示顺序	93
显示的程序实例	95
● 设置结构 (config)	95
● 坐标转换	96
● 扩充Sprite类	97
● 输入操作	100
● 窗口操作	103
● 战斗类	109
鼠标事件	119
● 执行程序实例	120
动作验证	121
故意的 bug 也是一种方法	121

Chapter 5 人物移动 123

移动的思维方式	124
● 判断可移动的位置	124
● 确认移动距离	125





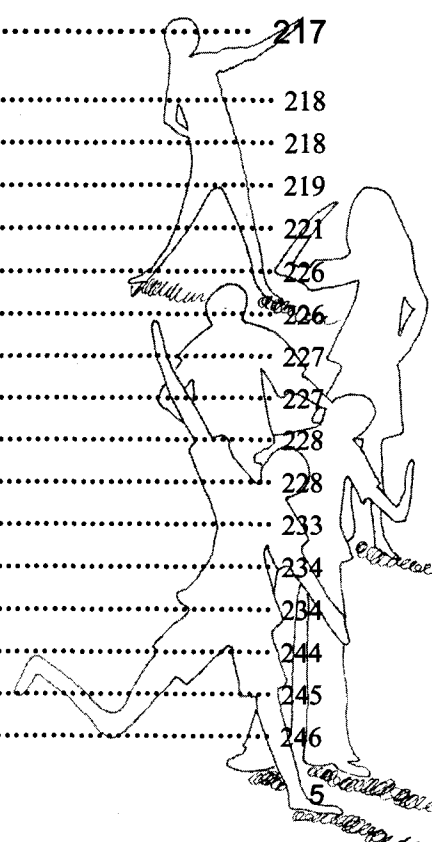
CONTENTS 目录

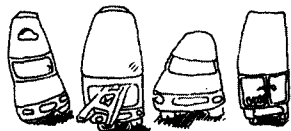
验证移动判断	127
● 移动到指定位置	130
移动动画	131
● 动画格式	131
● 登录动画	132
移动的程序实例	135
● 扩充战斗类别	135
● 加上输入处理	141
● 执行程序实例	144
Chapter 6 战斗	147
读入地图数据	148
● 写配置文件	148
● 读入文本文件	150
读入文本文件	150
词法分析 (lexical analyze)	154
● 战斗类加入读入配置文件的功能	162
战斗规则	178
● 决定判断例程 (Routine)	178
● 计算时所需的参数	179
● 计算破坏力	179
● 随机数	180
随机数的临界爆发攻击	181
● 求出射程距离	181
攻击时的动画	182
● 动画模式	183
● 攻击处理的步骤	184
魔法攻击的处理	193
● 增加视觉特效	193
● 其他魔法	195
CPU 端的思维模式	196

● 选择攻击对象	196
● 攻击顺序	197
CPU 端人物的攻击	198
● 搜寻候选的移动位置	198
● 决定移动位置	198
● 攻击游戏玩家所选的人物	199
程序代码解释	203
● 驱动CPU端	203
等待消息的处理	203
执行程序实例	206
● 显示状态	206
设计弹出式状态说明	208
● 输入命令	211

Chapter 7 事件 217

事件的执行方法	218
● 设定故事大纲	218
● 决定命令	219
● 命令	221
● 主菜单用的命令	226
命令的执行处理	226
● 解释型？编译型？	227
● “语法分析”和“词法分析”	227
● 读入script及执行命令	228
执行 script	228
触发事件时的输入处理	233
● 处理的内容	234
命令的处理	234
● 文字内容显示	244
● 菜单显示	245
● 绘制	246





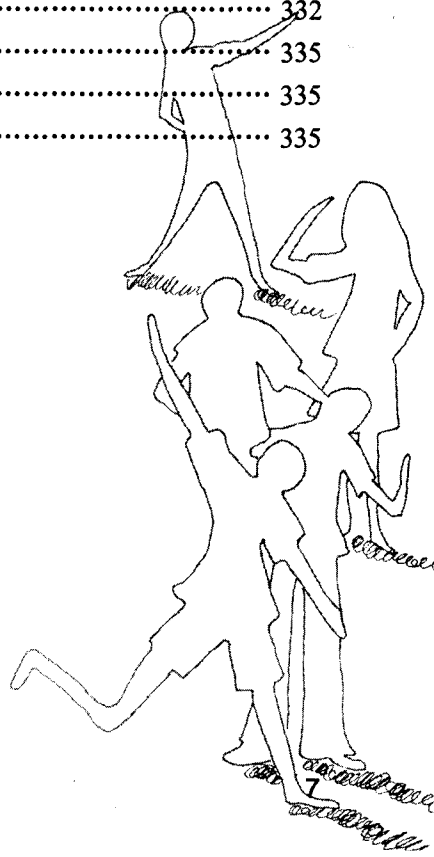
CONTENTS 目录

效果	248
● 计时器处理	248
● 擦入	250
● 渐淡	251
● 合成图形	253
● 执行程序实例	254
Chapter 8 编辑参数	257
参数	258
● 利用参数调整游戏平衡	258
参数编辑器	258
● 雏形	259
● 对话框	261
● 编写程序代码	262
读取 / 写入参数文件	262
参数的编辑处理	267
● 执行程序实例	275
利用 dummy	275
Chapter 9 大功告成	277
增补其他功能	278
从外部文件读入参数	278
● 修改源程序代码	278
存储参数	279
结合战斗部分和故事大纲	283
● 战斗开始	283
● 战斗结束	284
判断结束	285
播放音乐	287
● MCI的使用	287

● 追加标头和函数库	300
存储 / 装入	301
播放音效	308
提升等级	311
● 提升等级的适当时机	311
● 提升等级的处理	312
直接跳跃	316
● 直接跳跃的命令	317
故事大纲实例	320
● 故事大纲说明	330

Chapter 10 浅谈即时系统 331

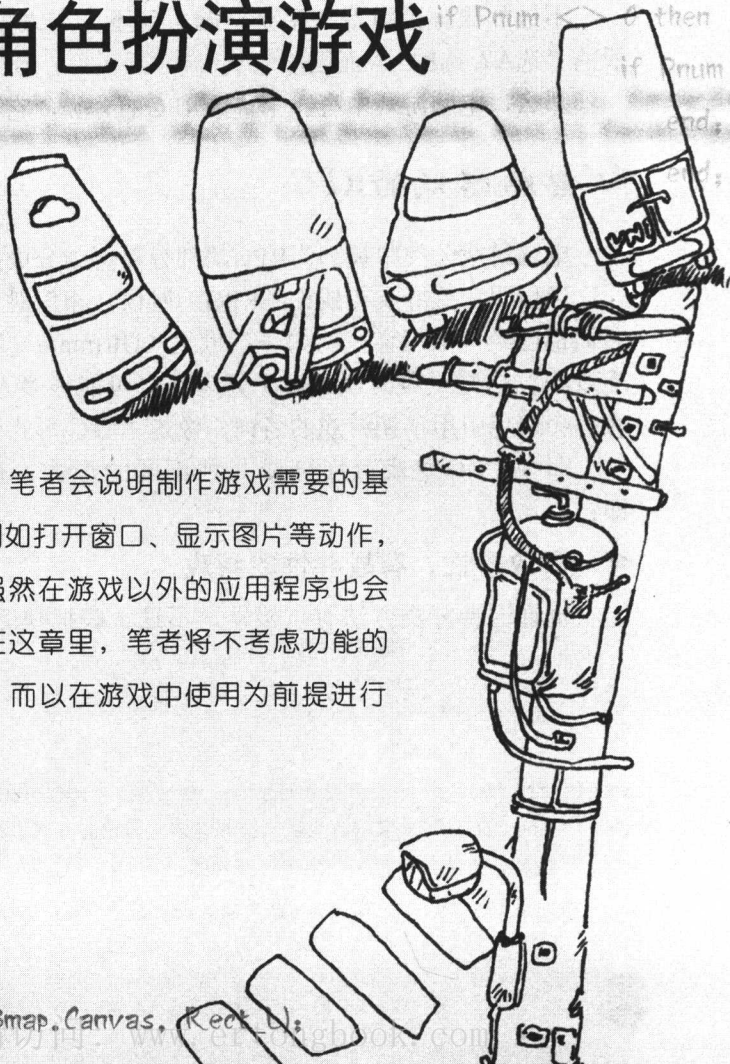
战略型即时系统的基本概念	332
● 游戏玩家所选人物的移动	332
更自由的移动系统	335
● 敌方人物的移动	335
● 战斗	335



CHAPTER

1

写一个角色扮演游戏



在这章里，笔者会说明制作游戏需要的基本内容。例如打开窗口、显示图片等动作，这些动作虽然在游戏以外的应用程序也会使用，但在这章里，笔者将不考虑功能的“泛用性”，而以在游戏中使用为前提进行说明。

何谓角色扮演游戏 (RPG)

角色扮演游戏 (RPG, Role Playing Game的缩写) 究竟是指什么样的游戏内容?

如果游戏过程没有电脑参与的话 (如table talk^{※注1-1}、版图游戏 (board game) 等), 角色扮演 (Role Playing) 就如同字面上的意义, 它其实是一种由游戏玩家自己扮演角色的游戏。在遵守既定规则的前提下, 游戏玩家摇身一变成为某个角色 (即若选择勇者的角色, 自己就是勇者), 自己决定采取哪些行动、让游戏发展下去。因此, 这类游戏的可能发展层面很广, 可任由游戏玩家自由发挥。

不过如果你的游戏对手是电脑, 就不会有太大的选择空间, 顶多也只是死板的“选A? 选B?”而已。

那么, 电脑游戏中有一个称为RPG的分类, 它又代表什么意思?

● 电脑游戏的RPG

电脑游戏分类中被列为RPG的部分可再细分成几个种类。

当你跟三五好友在谈论到RPG的时候, 相信对一些从游戏代表作流传出来的名词都已经耳熟能详, 例如“创世纪 (Ultimate)”、“巫术 (Wizardry)”。

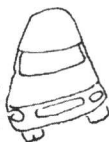
不过毕竟这些经典游戏都属于元老级, 可能很多人没有玩过, 所以本书在形容时会避免引用早期游戏的名词, 像是“○○形式”、“××风格”等。

即使RPG游戏可细分成几种不同的形式, 但它们还是有共同的部分。即:

● 有故事性、有战斗性的游戏

也许这种分类方法有点极端, 不过一般把缺乏故事性的游戏划为“战略 /

注1-1 tabletalk 一种采取以游戏玩家彼此对谈、讨论的方式进行的。每个游戏者都是自己选择的特定人物 (角色), 由担任裁判的游戏主持 (Game Master) 审查游戏玩家的行动。这种游戏仅预先设定游戏规则, 故事内容则可自行决定, 所以游戏的发展空间较广。



战争模拟游戏”，而不着重在战斗攻击方面的游戏则称为“冒险游戏”或“模拟系列（visual novel）”。其实市面上的游戏杂志也多半采取同样的分类方式。

因此，本书也是设定为“RPG是指有战斗、而且有故事性的电脑游戏”。

RPG分类

Visual Novel
角色扮演

由于“创作RPG”如果只用“有故事性、有战斗的游戏”来定义，显得太过笼统不够严谨，所以还要再进一步分成几个小类。

角色扮演游戏可根据下列几项来细分：

故事性

移动

战斗

下面就各项分别说明。

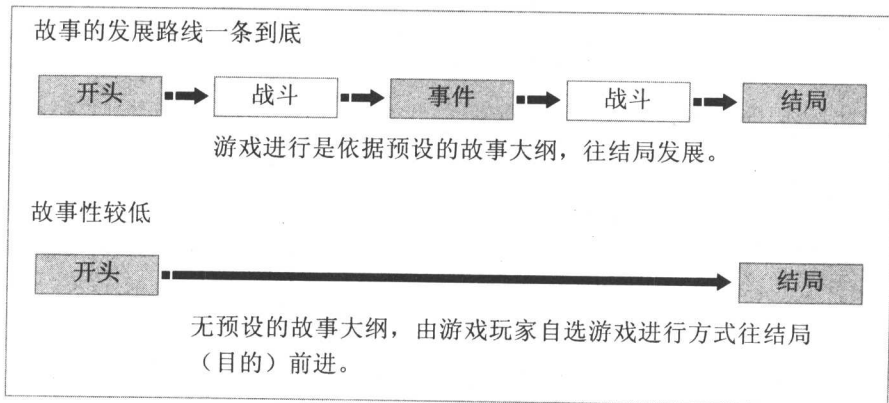
● 故事性

故事的设定方式不限，可以是只确定故事背景和最后结局，其他部分则不着墨太多，当然也可以是整个流程（故事）几乎都固定下来，发展路线就是一条到底。我们先从故事内容的预设比例来分类RPG。

给游戏玩家的选择空间越大，故事性就相对偏低。

以类似“勇者斗恶龙”的故事为例，游戏一开始就会发生一件意外要你去消灭恶龙，当恶龙被消灭后就是大结局。在前往杀死恶龙的路上会出现各种事件和战斗场面。

游戏是设计成根据游戏玩家的行动的改变而改变路上所发生的事件，还是所有事件不受影响只会依序发生，当然也会产生不同的游戏内容，不过基本上这类游戏只要有开头（opening）和结局（ending）就是一个故事，如图1-1所示。

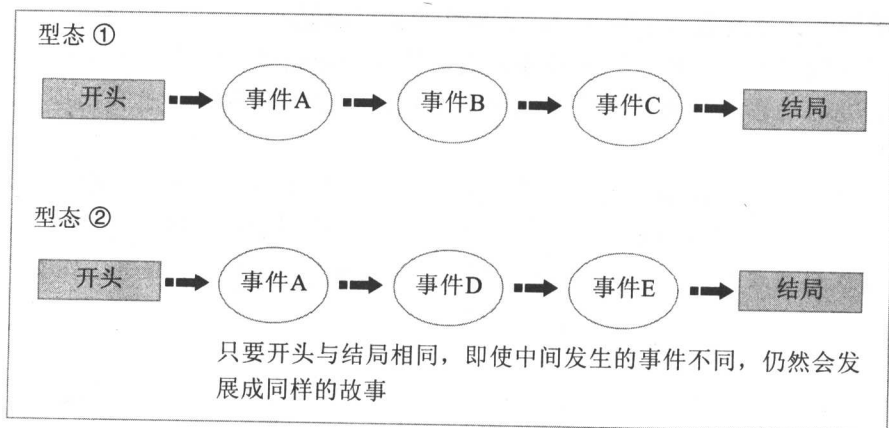


【图1-1】故事性

例如：

“勇者奉国王之命去除掉恶龙。经过无数次的冒险犯难后，终于杀死恶龙，国家也恢复和平”。

要是像这样光看故事“骨架”的话，其实不管路上“无数次的冒险犯难”是哪些、发生顺序如何，都可以算是同样的故事结构。不过话说回来，就算故事内容相同，事件的内容安排如果巧妙得当就能增加游戏的趣味性，反之则会让游戏索然无味，如图1-2所示。



【图1-2】事件和故事



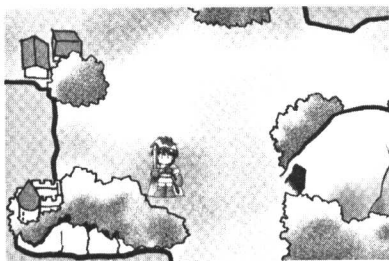
虽然说让事件多些变化可以增加游戏玩家的兴趣，不过以现阶段电脑游戏的程度来看，要像table talk那样能配合游戏进行，让游戏玩家任意“产生事件”还是心有余而力不足，所以目前的做法仍然是根据游戏玩家的行动内容，选择几个预设的事件。

● 移动

移动的方法有能在整个地图上游走自如的移动方式，也有只限从几个固定地点中择一移动过去的方式。当然，有些移动动作甚至不允许游戏玩家自行选择移动位置。

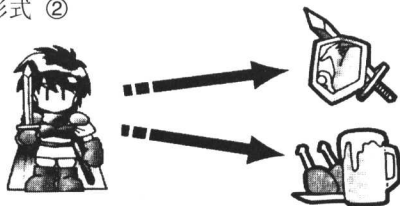
利用移动的方法也可以来分类RPG，如图1-3所示。

形式 ①



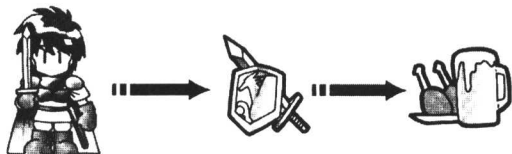
游戏玩家可在画面（地图）上自由移动

形式 ②



游戏玩家可自行选择地点移动

形式 ③



随着游戏进行，人物会自动移动到固定的地点

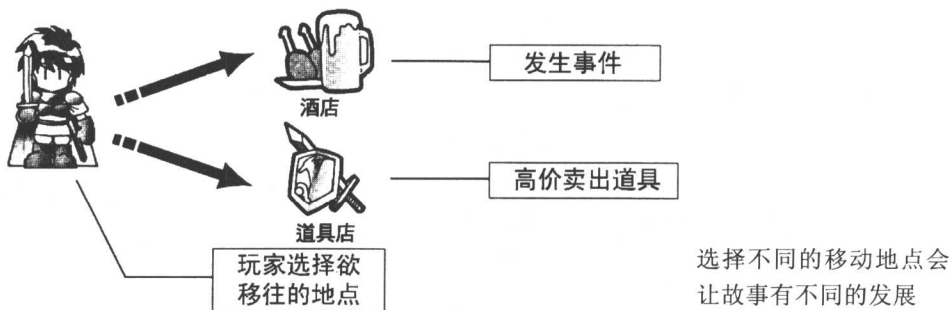
【图1-3】 根据移动而分类

通常多半是在要发生事件的时候就会有移动。

你可以要求游戏玩家选择要移动到A地还是B地，再要求他选择移动结果所引发的事件。还是举个例子来说明好了。

假设当人物手上有一个道具时，就有“到酒店，发生事件”和“到道具店，高价卖出”这两种选择。如果发生事件，就不会拿到钱；但如果卖掉道具，就不会发生事件。这里所发生的事件不仅是这个时间点的行动而已，它还会影响到后面的发展，所以选择移动位置这个动作乍看之下似乎无关紧要，其实会让整个故事发展有很大的变化，如图1-4所示。

游戏玩家手上如果没有说明这个岔路的信息^{※注1-2}，“先到哪里”的决定就会影响后面的故事发展。也有可能是游戏玩家没有注意到自己做了选择（有时没有注意到会让故事更有趣，反过来也可能让自己找不到路走出来）。要是事先就有这类信息，如果有“现在需要增加财富”的问题，就可以选择适当的行动。至于哪种设计方法比较好，就得看游戏的重点在哪里了。



【图1-4】 故事因移动位置而有不同的发展

有些移动当然跟故事发展没有关系（例如到武器店、道具店），不过移动的主要目的还是在抉择故事发展方向或推动游戏向前发展。

● 战斗

战斗次数、所需时间和战斗方法等也是RPG分类的重点。

如果游戏的主要目的是移动和冒险，单次战斗通常都会被简化，很快就结束。如果整个游戏的战斗次数不多，则多半是需要花时间思考的策略型战斗。

注1-2 事前消息这里的消息可能是得到道具时的说明，如“这可以高价卖出”、“○○会想要这个道具”等等，“攻略本的消息”当然也是一种消息。



简单版的战斗方式很少会在战斗中移动，大部分是“面对面互相攻击”的形式。而花时间思考的战斗方式通常有点类似版图游戏，会有特定人物的移动或攻击（本书的程序实例就是采用版图式的战斗形式），如图1-5所示。

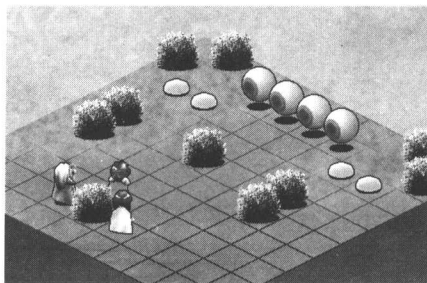
另外，有些游在战斗时会切换到战斗专用画面，有些则无。

通常是自己跟敌人轮流攻击



简单版的战斗画面

版图式的战斗画面（程序实例）



版图式战斗系统所采取的行动模式是先移动到敌人旁边再攻击或进行其他动作

【图1-5】 战斗画面

单次战斗时间 × 总战斗次数 + 事件等所用时间 = 总游戏时间

由于上面的计算式成立，所以战斗次数越多当然就越偏向简单版，战斗次数越少则选择边思考边玩的游戏方法。

● 系统概要

根据故事性、移动和战斗这三个要素的比例组合，即可决定RPG的系统概要。而若再加上系统和游戏视觉效果（2D或3D等）的差异，则可再细分下去。

游戏视觉效果有由上往下正视的方形图示法、45°斜视角（俯瞰）或利用3D任意转动视角等几种差别，如图1-6所示。如果游戏设计有配合不同的时地采