2016 －2017 学年第 二 学期

组号：G15



XP碎片开发

总体设计说明书

****

实验课程名称 软件工程导论

专 业 班 级 软件工程1501

学 号 31501317 31501305 31501299

学 生 姓 名 郑丁公 谢正树 嵇德宏

实验指导老师 杨枨

目录

[1 引言 3](#_Toc481177002)

[1.1 编写目的 3](#_Toc481177003)

[1.2 范围 3](#_Toc481177004)

[1.3 定义 3](#_Toc481177005)

[1.4 参考资料 4](#_Toc481177006)

[2 总体设计 4](#_Toc481177007)

[2.1 需求规定 4](#_Toc481177008)

[2.2 运行环境 6](#_Toc481177009)

[2.2.1.最低系统配置 6](#_Toc481177010)

[2.2.2推荐系统配置 6](#_Toc481177011)

[2.3 基本设计概念和处理流程 7](#_Toc481177012)

[2.4 结构 7](#_Toc481177013)

[2.5 功能需求与程序的关系 10](#_Toc481177014)

[2.6人工处理过程 10](#_Toc481177015)

[2.7 尚未解决的问题 10](#_Toc481177016)

[3 接口设计 11](#_Toc481177017)

[3.1 用户接口 11](#_Toc481177018)

[3.2 外部接口 11](#_Toc481177019)

[3.3 内部接口 12](#_Toc481177020)

[4 运行设计 12](#_Toc481177021)

[4.1 运行模块组合 12](#_Toc481177022)

[4.2 运行控制 13](#_Toc481177023)

[4.3 运行时间 13](#_Toc481177024)

[5 系统数据结构设计 13](#_Toc481177025)

[5.1 逻辑结构设计要点 13](#_Toc481177026)

[5.2 物理结构设计要点 16](#_Toc481177027)

[5.3 数据结构与程序的关系 17](#_Toc481177028)

[6 系统出错处理设计 19](#_Toc481177029)

[6.1 出错信息 19](#_Toc481177030)

[6.2 补救措施 19](#_Toc481177031)

[6.3 系统维护设计 19](#_Toc481177032)

[7 界面原型设计 20](#_Toc481177033)

# 1 引言

## 1.1 编写目的

系统设计是开发人员进行的工作，他们将系统设计阶段得到的目标系统的逻辑模型转换为目标系统的物理模型，该阶段得到工作成果――系统设计说明书是下一个阶段系统实施的工作依据。预期的读者是软件开发人员，由于这只是课程设计的一部分，因此是在小组里是所有的小组人员。

## 1.2 范围

**项目名称**：RPG游戏开发计划

**任务提出者**：郑丁公

**用户：**14-30岁的人群 ****

**开发者**：嵇德宏，谢正树，郑丁公

项目提出的背景：目前，人们的生活水平提高的同时，压力也在一步步提升，再加上电脑游戏热潮席卷全球，游戏软件的开发和维护正在成长为一个新兴的产业。RPG游戏（角色扮演类游戏）无疑是目前市场上最受欢迎的游戏类型之一。

**该软件系统同其他系统或其他机构的基本的相互来往关系:**该系统主要运行在WINDOWS 98以上的WINDOWS系列操作系统上。 因此需要有WINDOWS操作系统的支持，底层的所有工作包括声音依靠WINDOWS的声音处理API，图像处理全部依靠WINDOWS的GDI。该系统属于应用软件系统，因此处于应用层，在有这些操作系统层的支持下，不需要其它额外的支持。由操作系统负责与该系统用到的输入输出打交道。

**目标**：文案第一章部分必须全部完成，之后部分待定

## 1.3 定义

RPG:role-play-game—角色扮演游戏

RMXP:RPGMakerXP

PS:Adbode PhotoShop CS6(仅限于本项目)

PR:Adbode Premiere CS5(仅限于本项目)

VOCALOID3:一款音频软件

GIT: 最先进的分布式版本控制系统

## 1.4 参考资料

软件工程导论（第6版）张海藩 牟永敏著

游戏概论

计算机网络（第三版）吴功宜著

操作系统精髓与设计原理（第七版）William Stallings 著

# 2 总体设计

## 2.1 需求规定

**精度**

计算数据为双精度，其它的为整型或字符型。

**时间特性要求**

a． 响应时间:30ms

b． 更新处理时间:50ms；

c． 数据的转换和传送时间:10ms内；

**灵活性**

a． 操作方式上的变化：键盘操作改变的话基本代码改变很小，如果需要鼠标的操作改

动就大一点，大概需要1%的改动。

b． 运行环境的变化：如果不是WINDOWS操作系统则无法运行，如果没在系统要求下则

运行缓慢或显示不清，这个几乎无法提高；

c． 同其他软件的接口的变化：只要操作系统符合要求即可，若硬件驱动有问题可能也

无法运行，系统无法自动适应，必须外界提供好；

d． 精度和有效时限的变化：精度变大的话很难适应，变小的话可以适应，只是部分数

据不够精确，有效时限短了可能无法全部完成，只能删减功能；

e． 计划的变化或改进：如果改进则效率和效果都会提高，否则能按要求完成。

**输入输出要求**

**输入：**

输入是键盘上的操作，是键盘ASCII码,用到字符型，整形，布尔型，字符串以及数组等。

**输出：**

输出是保存的存档文件，或者显示在屏幕上的游戏信息、图片。

**数据管理能力要求（针对软件系统）**

　大概要有管理1519个文件，106个文件夹。总大小294MB的能力。但都不是数据库文件，全部为自定义的格式文件或标准格式文件。可预见30%的增长存储要求。

## 2.2 运行环境

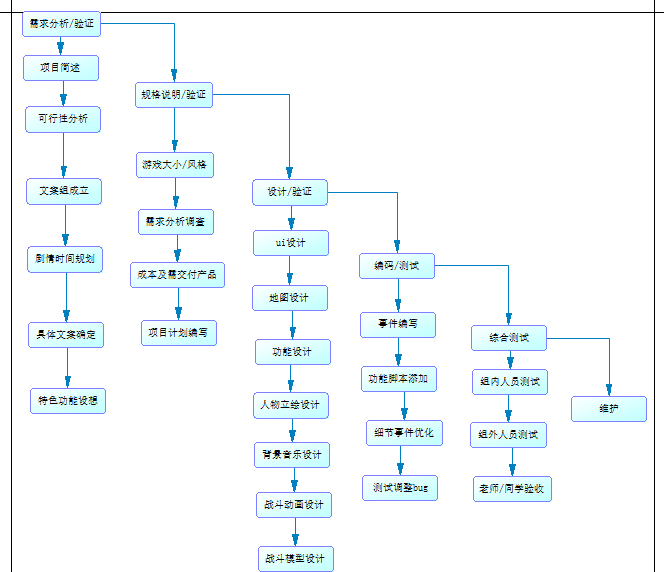
### 2.2.1.最低系统配置

|  |  |
| --- | --- |
| **系统** | Windows 98 / 98SE / Me / 2000 / XP 中文版 |
| **CPU** | Intel PentiumIII 800 MHz 以上 |
| **内存** | 128 MB 以上 |
| **显卡** | 分辨率 1024×768 以上高彩 |
| **声卡** | 兼容 DirectSound 声卡 |
| **硬盘** | 可用空间 100 MB 以上 |

### 2.2.2推荐系统配置

|  |  |
| --- | --- |
| **系统** | Windows XP 中文版 |
| **CPU** | Intel Pentium4 1.5 GHz 以上 |
| **内存** | 256 MB 以上 |
| **显卡** | 分辨率 1024×768 以上真彩 |
| **声卡** | 兼容 DirectSound 声卡 |
| **硬盘** | 可用空间 500 MB 以上 |

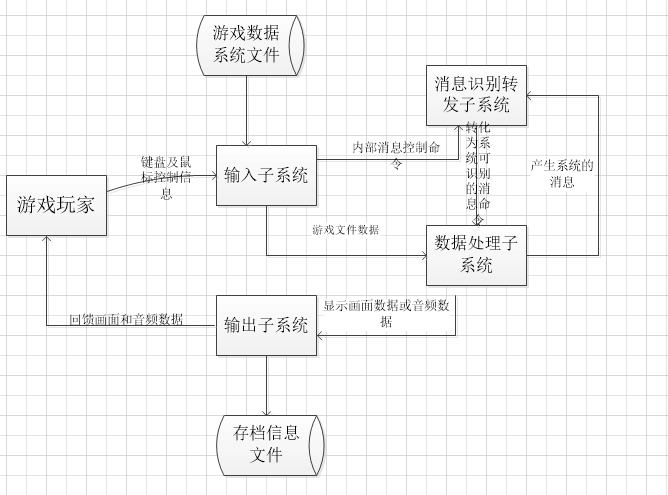
## 2.3 基本设计概念和处理流程



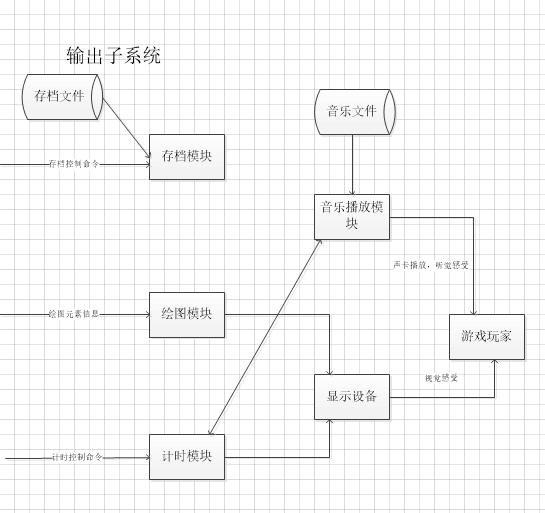
## 2.4 结构

**总流程结构**

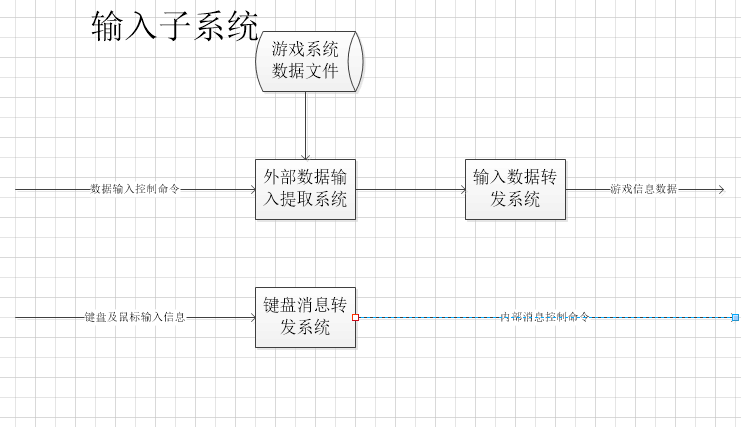
**![V$V]ZO](F[V~EGJ6P0VKW7I](data:image/jpeg;base64,)**

****

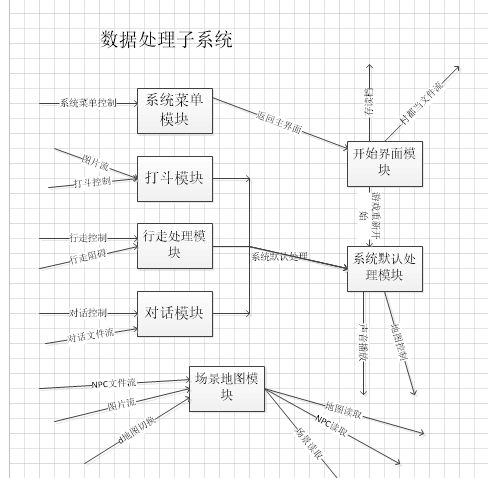
**输出子系统结构**

****

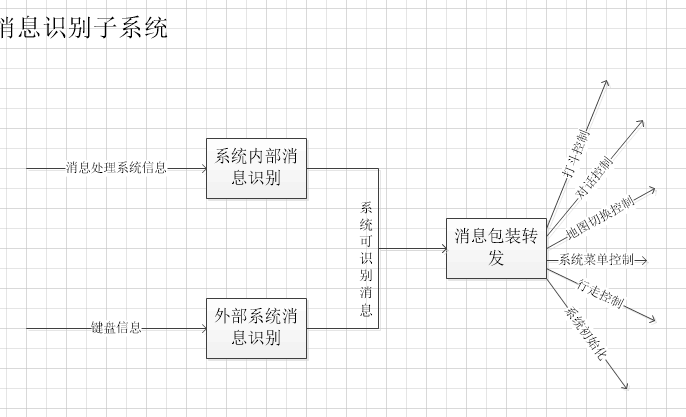
**输入子系统结构**

****

**数据处理子系统结构**

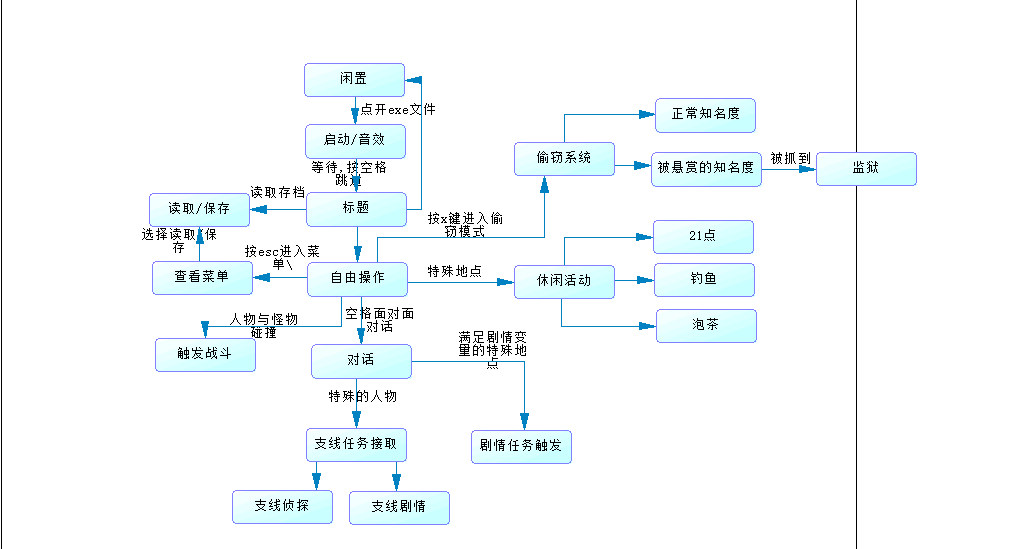
****

**消息识别子系统结构**

****

## 2.5 功能需求与程序的关系

本条用一张如下的矩阵图说明各项功能需求的实现同各块程序的分配关系：



## 2.6人工处理过程

【说明在本软件系统的工作过程中不得不包含的人工处理过程（如果有的话）。】

## 2.7 尚未解决的问题

【说明在概要设计过程中尚未解决而设计者认为在系统完成之前必须解决的各个问题。】

1）了解整个软件的需求分析。

2）对于软件的数据结构和整体关系进行全面了解。

# 3 接口设计

## 3.1 用户接口

本程序与用户的接口为GUI接口，即图形接口。目前我们设计的用户接口主要通过键盘与系统交互。程序主要捕获系统将ASCII码转换后的虚拟键值码来与用户交互。而用户的键盘输入是根据程序的提示而选择的，目前的有效按键只有：方向键、回车、ESC、A、 Alt、Ctrl随着我们的设计深入可能会有增加，但均会提示用户。

软件的回答信息即为图形信息和文字信息，其接口是用户的显示输出设备。即通过显卡输出到显示器上，展现给用户。 （例如：点击进入游戏界面时，当用户输入开始，则运行游戏；用户输入结束，则退出游戏；在游戏运行过程中，当用户输入行走，对话，战斗，事件之中的其中一个命令时，系统亦会做出相应的操作，当用户输入行走命令时，系统发出行走的操作，并同时把下一个动作的命令显示给用户，让用户选择是否继续或是另输入其他的命令操作。）

## 3.2 外部接口

本系统不直接与硬件接触，全部与硬件接触都是通过操作系统进行的。而与用户接触的输入设备是键盘，输出设备是显示器。

而本系统是运行在Windows98以上操作系统的应用程序，且只能运行在Windows系列操作系统。通过调用操作系统的API函数来处理数据及间接操作硬件来与用户交互。

不需其它软件的支持。只需有Windows操作系统的动态链接库。 初步统计目前本程序用到的DLL文件：

而以上程序接口均由编译器在生成本系统的PE文件时链接好，在本系统运行时将直接与这些接口交互。

本游戏画面风格为欧洲中世纪风情，画面制作精良。在游戏中，主要有以下几个界面 

行走界面：玩家通过键盘方向键控制角色的行动，还可以在本界面通过空格键完成拾取物品，与NPC对话触发剧情，进入战斗等操作 

属性物品界面：在本界面中，玩家可以查看角色的属性，使用物品增加属性值。  打斗界面：触发战斗后将进入本界面，在本界面，玩家操作角色如敌人进行回合制战斗

## 3.3 内部接口

内部接口方面，各模块之间采用函数调用、参数传递、返回值的方式进行信息传递。接口传递的信息将是以数据结构封装了的数据，以参数传递或返回值的形式在各模块间传输

# 4 运行设计

## 4.1 运行模块组合

【说明对系统施加不同的外界运行控制时所引起的各种不同的运行模块组合，说明每种运行所历经的内部模块和支持软件。】

游戏在运行时通过主页面和各模块之间的调用，读入操作进行格式化。在主模块得到充分的数据后，将调用各个子模块，显示子模块对应的操作和信息。子模块返回主模块后，可继续进行上述操作。

## 4.2 运行控制

【说明每一种外界的运行控制的方式方法和操作步骤。】

运行控制将严格按照各模块间函数调用关系来实现。在各事务中心模块中，需对运行控制进行正确的判断，选择正确的运行控制路径。

## 4.3 运行时间

【说明每种运行模块组合将占用各种资源的时间。】

# 5 系统数据结构设计

## 5.1 逻辑结构设计要点

**事件页信息表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **事件页信息表** | | | | | |
| 列名 | 类型 | 是否为空 | 是否主键 | 是否外键 | 说明 |
| condition | Condition | 否 | 否 | 否 | 条件（RPG::Event::Page::Condition）。 |
| graphic | Graphic | 否 | 否 | 否 | 图像（RPG::Event::Page::Graphic） |
| move\_type | Fixnum | 否 | 否 | 否 | 移动类型。（0：固定，1：随机，2：接近，3：自定义） |
| move\_speed | Fixnum | 否 | 否 | 否 | 移动速度。（1：Slowest，2：Slower，3：Slow，4：Fast，5：Faster，6：Fastest）。 |
| move\_frequency | Fixnum | 否 | 否 | 否 | 移动频度。（1：Lowest，2：Lower，3：Low，4：High，5：Higher，6：Highest）。 |
| move\_route | MoveRoute | 否 | 否 | 否 | 移动路线（RPG::MoveRoute）。仅限于移动类型为自定义时调用 |
| walk\_anime | true/false | 否 | 否 | 否 | 选项「移动时动画」的真伪值 |
| step\_anime | true/false | 否 | 否 | 否 | 选项「停止时动画」的真伪值 |
| direction\_fix | true/false | 否 | 否 | 否 | 选项「固定朝向」的真伪值 |
| through | true/false | 否 | 否 | 否 | 选项「充许穿透」的真伪值 |
| always\_on\_top | true/false | 否 | 否 | 否 | 选项「在最前面显示」的真伪值 |
| trigger | Fixnum | 否 | 否 | 否 | 触发条件（0：决定键，1：与主角接触，2：与事件接触，3：自动执行，4：并行处理）。 |
| list | Array | 否 | 否 | 否 | 执行内容。RPG::EventCommand 的数组 |

**事件页[条件]信息表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **事件页[条件]信息表** | | | | | |
| 列名 | 类型 | 是否为空 | 是否主键 | 是否外键 | 说明 |
| switch1\_valid | true/false | 否 | 否 | 否 | 表示条件「开关」（第一个）是否有效的真伪值 |
| switch2\_valid | true/false | 否 | 否 | 否 | 表示条件「开关」（第二个）是否有效的真伪值 |
| variable\_valid | true/false | 否 | 否 | 否 | 表示条件「变量」是否有效的真伪值 |
| self\_switch\_valid | true/false | 否 | 否 | 否 | 表示条件「独立开关」是否有效的真伪值 |
| switch1\_id | Fixnum | 否 | 否 | 是 | 条件「开关」（第一个）有效时，其开关 ID |
| switch2\_id | Fixnum | 否 | 否 | 是 | 条件「开关」（第二个）有效时，其开关 ID |
| variable\_id | Fixnum | 否 | 否 | 是 | 条件「变量」有效时，其变量 ID |
| variable\_value | Fixnum | 否 | 否 | 否 | 条件「变量」有效时，其变量的标准值（x 以上） |
| self\_switch\_ch | String | 否 | 否 | 否 | 条件「独立开关」有效时，其文字（"A".."B"） |

**事件页[图像]信息表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **事件页[图像]信息表** | | | | | |
| 列名 | 类型 | 是否为空 | 是否主键 | 是否外键 | 说明 |
| tile\_id | Fixnum | 否 | 是 | 否 | 地图元件 ID。指定图像非地图元件的话为 0 |
| character\_name | String | 否 | 否 | 否 | 角色图像的文件名 |
| character\_hue | Fixnum | 否 | 否 | 否 | 角色图像的色相变化值（0..360） |
| direction | Fixnum | 否 | 否 | 否 | 角色的朝向（2：下，4：左，6：右，8：上） |
| pattern | Fixnum | 否 | 否 | 否 | 角色的模式（0..3） |
| opacity | Fixnum | 否 | 否 | 否 | 不透明度 |
| blend\_type | Fixnum | 否 | 否 | 否 | 合成方法 |

**事件指令信息表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **事件指令信息表** | | | | | |
| 列名 | 类型 | 是否为空 | 是否主键 | 是否外键 | 说明 |
| code | Fixnum | 否 | 是 | 否 | 事件代码 |
| indent | Fixnum | 否 | 否 | 否 | 缩进的深度。通常为 0，「条件分歧」等指令每加深一层其值 +1 |
| parameters | Array | 否 | 否 | 否 | 包含了事件指令参数的数组。其内容每个指令都不同 |

**移动路线信息表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **移动路线信息表** | | | | | |
| 列名 | 类型 | 是否为空 | 是否主键 | 是否外键 | 说明 |
| repeat | true/false | 否 | 否 | 否 | 选项「重复运作」的真伪值 |
| skippable | true/false | 否 | 否 | 否 | 选项「忽略不能移动的场合」的真伪值 |
| list | MoveCommand | 否 | 否 | 否 | 执行内容。RPG::MoveCommand 的数组 |

**移动指令信息表**

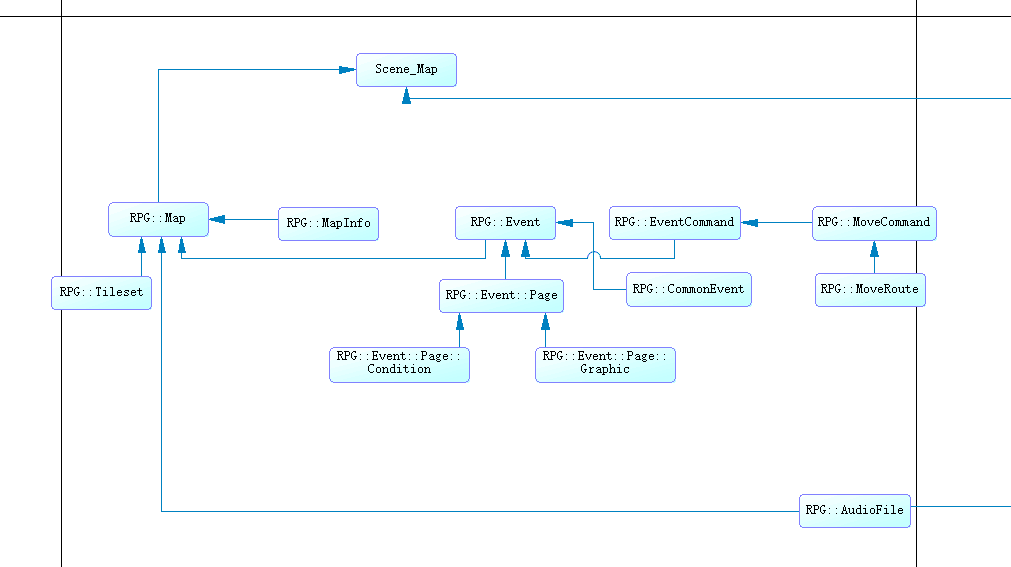
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **移动指令信息表** | | | | | |
| 列名 | 类型 | 是否为空 | 是否主键 | 是否外键 | 说明 |
| code | Fixnum | 否 | 否 | 否 | 移动指令代码 |
| parameters | array | 否 | 否 | 否 | 包含了移动指令参数的数组。其内容每个指令都不同 |

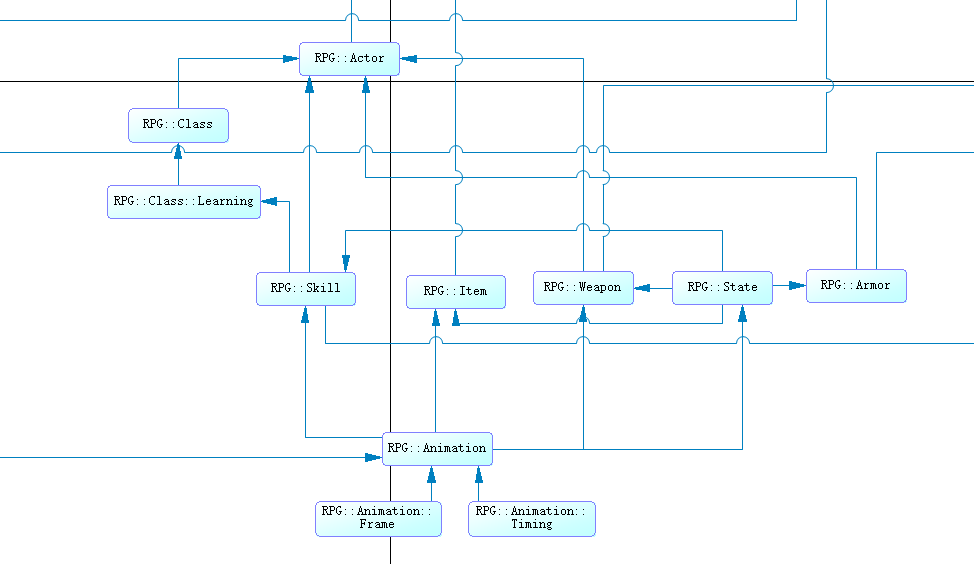
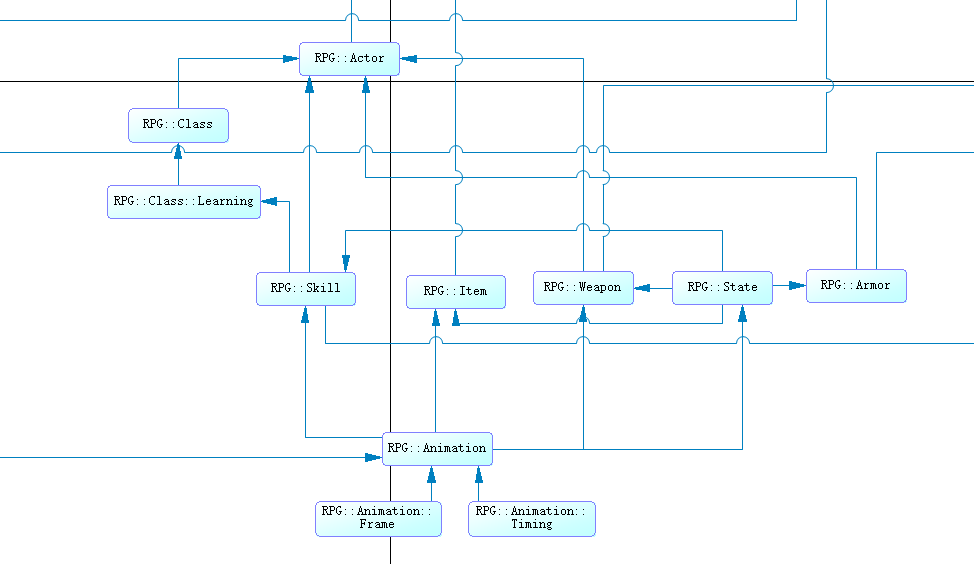
## 5.2 物理结构设计要点

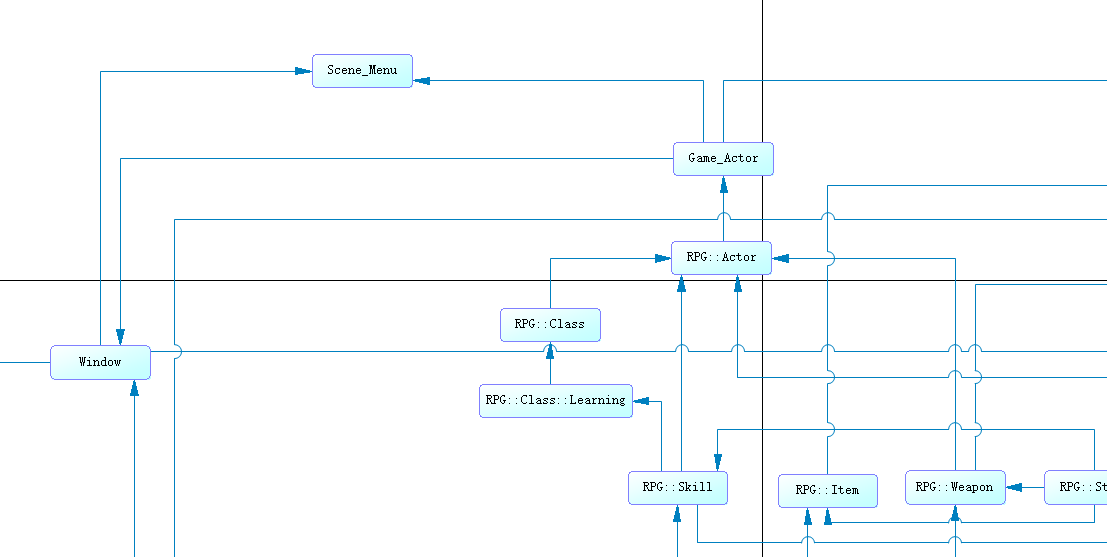
【给出本系统内所使用的每个数据结构中的每个数据项的存储要求，访问方法、存取单位、存取的物理关系（索引、设备、存储区域）、设计考虑和保密条件。】

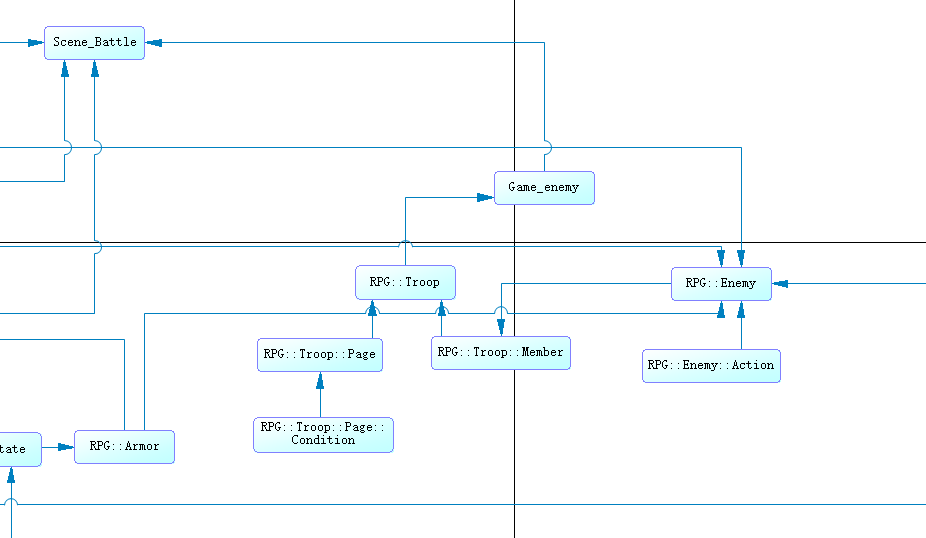
## 5.3 数据结构与程序的关系

说明各个数据结构与访问这些数据结构的各个程序之间的对应关系，可采用如下的矩阵图的形式：









# 6 系统出错处理设计

## 6.1 出错信息

【用一览表的方式说明每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。】

|  |  |  |  |
| --- | --- | --- | --- |
| 软件故障 | 假死现象 | 不能运行 | 存档读档不正常 |
| 原因 | 可能在地图切换时，调用的声音播放API需要缓冲 | 缺少完成的程序文件 | 没有正确存档和读档，只允许所要求的个数，缺少存档文件 |
| 故障处理要求 | 等几秒 | 拷贝完整的程序文件 | 重新存档 |
| 后果 | 有延迟 | 不能运行 | 存档读档文件损坏 |
| 硬件故障 | 无显示 | 无显示 | 显示不清晰 |
| 原因 | 驱动有问题，音响系统有问题 | 显示驱动有问题，显示器有问题 | 硬件显示不能符合要求 |
| 故障处理要求 | 重装驱动或者检音响系统 | 重装驱动或检修显示器 | 找到更高档次的硬件系统 |

## 6.2 补救措施

1. 若在安装期间出现错误导致程序无法执行，可以检查机器的性能进行再次安装。

2. 在硬件方面要保证机器能够稳定的运行。

3. 当出现大的程序问题重新打开游戏；如果无法在重新打开游戏后解决问题则只能关闭游戏进行抢修。

## 6.3 系统维护设计

【说明为了系统维护的方便而在程序内部设计中做出的安排，包括在程序中专门安排用于系统的检查与维护的检测点和专用模块。】

系统的维护和管理也是一个系统的问题，也包括两个层面：

1）技术层面 包括：（1）使用较新的开发和维护技术； （2）专职维护人员维护硬件

2）人员层面 包括：

（1）维护协议； （2）思想认识； （3）法律法规教育；

（4）工作流程及方式方法。

# 7 界面原型设计

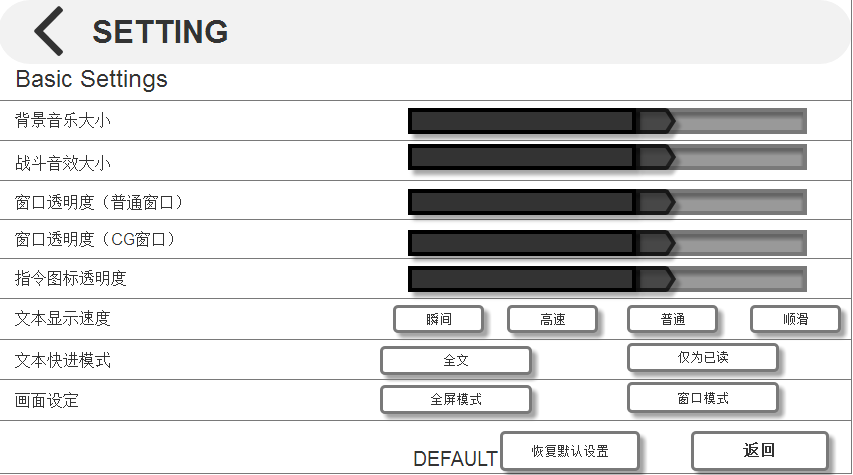
**标题界面**

****

**加载界面**



**设置界面**

****

**读档界面**

****