

ICS

点击此处添加中国标准文献分类号

T/

# 广州市番禺动漫游艺行业协会团体标准

T/××× ××××—××××

## 商用动漫游戏机 软件系统通用技术要求

点击此处添加标准英文译名

点击此处添加与国际标准一致性程度的标识

(征求意见稿)

××××-××-××发布

××××-××-××实施

广州市番禺动漫游艺行业协会 发布

## 前 言

本标准按照GB/T 1.1-2020给出的规则起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

本标准由广东省商用动漫游戏机标准化技术委员会提出并归口。

本标准起草单位：XXXX

本标准主要起草人：XXX

# 商用动漫游戏机 软件系统通用技术要求

## 1 范围

本标准规定了商用动漫游戏机的术语和定义、策划、系统开发要求、系统引擎、设计、和调试。本标准适用于广东省范围内商用动漫游戏机产品。

## 2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB 28022—2011 玩具适用年龄判定指南

GB 8408—2008 游乐设施安全规范

DB44/T 1454—2014 商用动漫游戏机 术语

## 3 基本要求

商用动漫游戏机的软件系统通用技术要求遵守国家《关于进一步加强游艺娱乐场所管理的通知》一文市发[2009]4号和其他相关标准要求。

一般来说，通过满足本部分中规定各项相关要求来实现上述准则。

确定商用动漫游戏机的合适年龄，应该着重考虑下列的特征：感官元素、因果效应、特许性、所需要的肌肉动作技能。

## 4 策划

### 4.1 适用人群

商用动漫游戏机推荐适用人群如下：

示例：游戏机适用人群

| 游戏特性        | (3岁-6岁)                    | (7岁-11岁)                                       | (12岁-16岁)                                      | (16岁以上)  |
|-------------|----------------------------|--|--|--|
| 目的性         | 认知、智力开发、表达能力               | 科普、运动能力、想象力                                    | 发展技能、逻辑思维                                      | 娱乐、趣味、放松   |
| 适合游戏类别/游戏设计 | 音乐类游戏机<br>益智类游戏机<br>礼品类游戏机 | 音乐类游戏机<br>运动类游戏机<br>竞速类游戏机<br>益智类游戏机<br>礼品类游戏机 | 音乐类游戏机<br>运动类游戏机<br>竞速类游戏机<br>益智类游戏机<br>礼品类游戏机 | 音乐类游戏机<br>运动类游戏机<br>竞速类游戏机<br>益智类游戏机<br>礼品类游戏机<br>格斗类游戏机<br>射击类游戏机 |

表1 游戏机适用人群（续）

| 游戏特性    | (3岁-6岁)  | (7岁-11岁)   | (12岁-16岁)  | (16岁以上)   |
|---------|--|--|--|---|
| 相关游戏/行为 | 宜设计低复杂度程度的游戏；<br>能应付游戏，包括目标的权衡；<br>慢速电动交通工具；<br>不包含灵活元素；<br>宜设计难度小的游戏； | 宜设计中低程度复杂度的游戏；<br>宜设计初级游戏规则、游戏策略；<br>宜设计多人游戏；<br>宜设计网络游戏；<br>宜设计体育游戏、科幻游戏和冒险游戏、社会性的游戏；<br>宜设计难度一般的游戏     | 略微复杂程度的游戏<br>宜设计中等进度游戏；<br>宜设计社会性游戏、多人游戏；<br>可设置游戏中的隐藏功能；<br>宜设计复杂的为时较长的游戏；<br>精细动作技能进一步接近成人；<br>宜设计难度中等的游戏； | 宜设计高程度复杂度的游戏；<br>游戏持续时间长；<br>宜设计挑战性游戏；<br>宜设计自由度、趣味度及真实性、剧情生动，动作华丽的游戏；<br>宜设计有中国元素的画面和文化元素的游戏；<br>宜设计设计细致、人物刻画细腻的画面；<br>宜设计玩法多样的游戏；<br>宜设计的难度大游戏  |
| 感官元素    | 宜设计音乐或会发声的游戏；<br>宜设置大字体；<br>如软件的感官因素十分丰富、活泼，例如配有音乐、运动形象、闪光灯、演讲和图像；     | 宜设计具有明亮光线的游戏；<br>能感知辅音组合等细微的音位区分；<br>宜设计标准/普通版字体；  | 宜设置流行音乐；<br>宜设置动漫元素；   | 宜设计画面、音效强的游戏；<br>宜设计立体感强的游戏；<br>宜设计视觉冲击强的游戏<br>宜设计可视角度广的游戏；   |
| 因果效应    | 中等复杂的因果效应功能（如：推动产生声音、光线）；<br>点式和敲击式的游戏；<br>清晰的因果效应；                    | 高等复杂的因果效应功能（如：多功能性，能多方向旋转的身体部件，能产生多种光线、声音、噪声、通过按钮、游戏操纵杆做出动作反应，智能芯片附件）；<br>对点式和敲击式游戏的兴趣减少；<br>能理解迟滞思的因与果； | 高度复杂的因果效应；<br>个性化；   | 宜设计挑战性强的游戏；<br>宜设计交流性强的游戏；<br>宜设计趣味性强的游戏；<br>宜设计知识性强的游戏；<br>宜设计动漫元素多的游戏；<br>宜设计有特效的游戏；<br>游戏类型；<br>游戏题材；<br>游戏操作；<br>宜设计情景精妙的游戏；<br>宜设计个性、立体强的游戏；<br>完善的世界观，丰富的剧情任务；<br>操作简单、界面直观、画面精美； |

表1 游戏机适用人群（续）

| 游戏特性      | (3岁-6岁)  | (7岁-11岁)   | (12岁-16岁)  | (16岁以上)  |
|-----------|--|--|--|--|
| 特许性       | 宜用不同媒体上流行的有特许的交通工具和形象；<br>宜用流行形象和卡通形象；                             | 宜用特许的交通工具在游戏中；<br>宜用基于电影和主题类套装；<br>宜用动作人物形象和体育明星感兴趣；                         | 宜用大众媒体的流行元素；<br>宜设计实践性强的游戏；                        | 宜用动漫故事的情节、人物、宜设计虚幻性的情节；                        |
| 所需的肌肉动作技能 | 中等肌肉精细动作调节灵巧性与控制能力；<br>容易操纵配件、松散件；<br>视觉辨识力和肌肉精细动作技能提高；<br>可设计脚踏板； | 躯干及手臂的协调；<br>中高等程度灵巧性、力量、肌肉粗大动作技能；<br>可设计手闸                                  | 高程度灵巧性、力量、肌肉粗大动作技能；<br>可设计双手能同时使用的游戏；<br>可设计换挡和手闸； |  |
| 控制方式      | 可设计按键和遥控杆；<br>可设计简单连接装置；<br>可设计控制杆和鼠标等输入装置；                        | 可设计大部分的遥控装置；<br>可设计中等复杂的连接装置；<br>可设计输入装置；<br>可设计地图引导浏览游戏；<br>可设计阅读的、遵守复杂的指令； | 可设计手柄控制；<br>可设计按键控制；<br>可设计触屏控制；<br>可设计较为复杂的输入方式   | 可设计手柄控制；<br>可设计按键控制；<br>可设计触屏控制；<br>可设计复杂的输入方式 |
| 游戏举例      | 简单多媒体软件；<br>较复杂目标明确的游戏；<br>简单的控制台游戏机和控制台游戏；                        | 媒体处理软件；<br>冒险游戏；<br>体育游戏；<br>复杂的控制台游戏机和控制台游戏                                 | 互动游戏；<br>模拟游戏；                                     | 射击类游戏；<br>复杂、逻辑型游戏；                            |
| 系统开发要求    | 游戏开发的系统结构。   |  |  |  |
| 系统引擎      | 为制作游戏需要而事先编制好的一系列同游戏初始化、图形处理、角色控制、碰撞检测等有关的 APU 函数                  |  |  |  |
| 设计        | 立体游戏设计、平面游戏设计、立体渲染技术   |  |  |  |
| 调试        | 根据不同阶段，设置的软件调试   |  |  |  |

## 4.2 游戏设计

游戏设计如下参考：

### a) 动作类

玩家控制游戏人物用各种武器消灭敌人以过关的游戏，不追求故事情节。

### b) 冒险类

由玩家控制游戏人物进行虚拟冒险的游戏。故事情节往往是以完成一个任务或解开某些谜题的形式出现的，而且在游戏过程中刻意强调谜题的重要性。

T/xxx xxx—xxx

c) 解谜类

通过对游戏中出现的信息或情节进行分析和处理，发掘线索解决各种谜题。

d) 策略类

玩家运用策略与电脑或其它玩家较量，以取得各种形式胜利的游戏，或统一全国，或开拓外星殖民地。

e) 竞速类

通过模拟汽车、摩托车等交通工具进行速度比赛，以实现人机互动的游戏。

f) 格斗类

由玩家操纵各种角色与电脑或另一玩家所控制的角色进行格斗的游戏。

g) RPG 角色扮演类

由玩家扮演游戏中的一个或数个角色，有完整故事情节的游戏。强调的是剧情发展和个人体验。

### 4.3 控制方式

a) 手柄控制

通过操纵其按钮等，实现对游戏机模拟角色等的控制。

b) 按键控制

通过键盘对游戏机的内容进行控制。

c) 触屏控制

用手指或其它物体触摸安装在显示器前端的触摸屏，然后系统根据手指触摸的图标或菜单位置来定位选择信息输入。触摸屏由触摸检测部件和触摸屏控制器组成。

### 4.4 控制系统安全

控制系统的安全防护应符合GB 8408-2008中6.4的规定。

## 5 系统开发要求

### 5.1 游戏系统结构

游戏系统结构如图 1 所示。

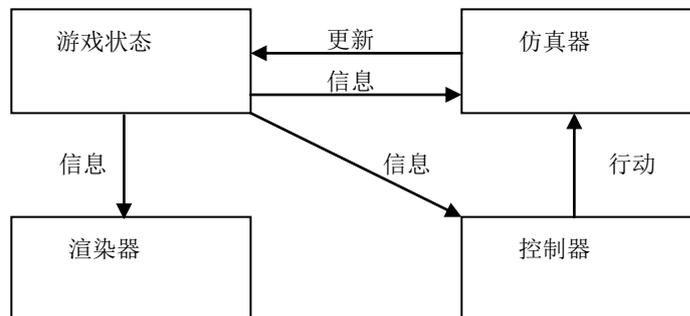


图1 游戏系统结构

注1：游戏状态（Game-State）：游戏状态代表当前游戏世界的状态，描述游戏世界中所有物体及其状态信息。

T/xxx xxx—xxx

注2: 仿真器 (Simulator): 仿真器制定游戏状态如何改变的规则, 即游戏物理学 (Game Physics)。具体说, 游戏角色控制器所选择的所有行动都是通过仿真器调动相应的动画实现的。

注3: 渲染器 (Renderer): 渲染器利用游戏中物体的几何关系及纹理提供对游戏状态的描述, 其输出结果通常包括图形和音响。

注4: 控制器 (Controllers): 每一个游戏角色都至少有一个控制器, 控制器负责角色的行动。

## 5.2 游戏框架系统

- a) 生存体系
- b) 升级体系
- c) 地图系统
- d) 斗系统
- e) 务系统
- f) 作系统
- e) 面系统
- f) NPC 设计
- g) AI 设计

## 6 系统引擎

### 6.1 构架

#### 6.1.9 结构化构架

将游戏引擎内部的模块以结构化的形式组合, 并以API的形式提供接口。

#### 6.1.10 基类根形式构架

基类根形式构架是以基本的几个抽象服务提供接口类为基础, 并以接口与实现相分离为原则, 进行引擎实现, 提供的实现类在引擎内部创建, 而将接口供给用户使用。以4个基本服务抽象类作为整个引擎底层提供服务的基础如表2所示。在基本类的基础上派生出引擎使用的基本类结构见表3。

表 2 基本服务抽象类及功能说明

| 类名                | 功能说明      |
|-------------------|-----------|
| IRefCount         | 提供引用计数功能  |
| ITypeInfoomation  | 提供类型识别功能  |
| IObjectInfomation | 提供对象识别功能  |
| IPersistence      | 提供对象持久化功能 |

表 3 基本类结构继承关系

| 类名        | 父类                                      |
|-----------|---|
| IBase     | ITypeInfomation                         |
| IResource | IBase、IObjectInfomation、ITypeInfomation |

注：IBase 类是引擎中绝大部分接口类的基类，它实现了的类型识别虚方法。IResource 类是引擎中资源的基类，它在 IBase 基础上实现了引用计数和对象信息功能。

### 6.1.11 组件形式构架

组件形式构架是把不同功能的模块做成互相独立的系统，模块内部可以使用任何构架方式，只需要提供相应的接口。

### 6.1.12 CORA 立体引擎架构

CORA 立体引擎架构如下图。

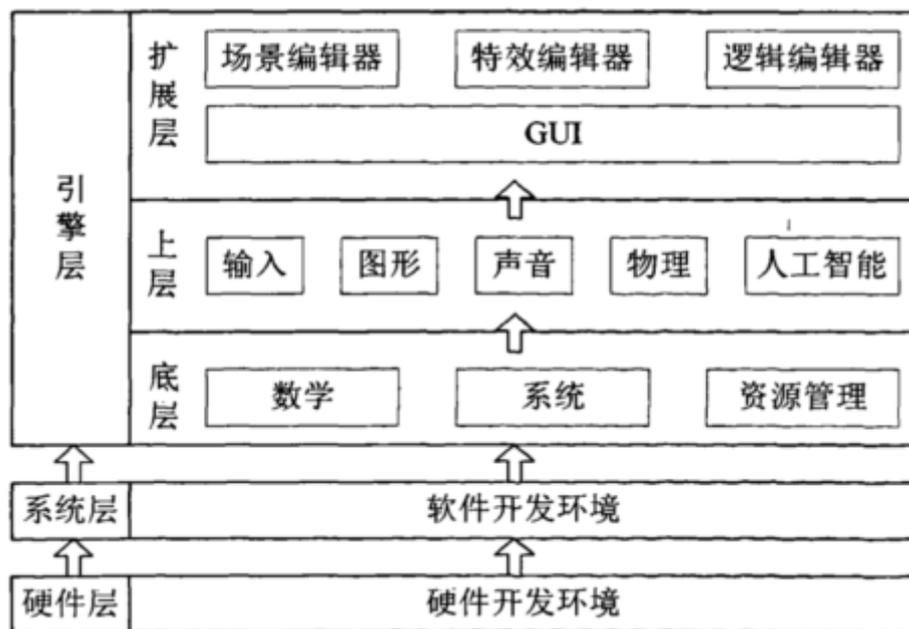


图2 CORA 引擎架构

引擎的底层由以下三个模块组成：

a) 系统模块

实现平台的基本公共功能。主要包括：内存分配模块、异常错误检测模块、智能指针模块、CPU检测模块、编译环境检测模块和日志模块等。

b) 数学模块

实现基本的矩阵、向量的几何代数操作。包括2D向量、3D向量、四元数、3D平面、4x4矩阵、轴对称包围盒和基本几何体等。

c) 资源管理模块

场景在运行时需要载入各种各样的资源，包括三维模型、纹理贴图、骨骼资源、地形资源、特效资料等。

引擎的上层由以下五个模块组成：

d) 输入模块

负责用户与电脑之间的沟通。处理来自键盘、鼠标、摇杆和其他外设的信号。该模块支持两种获取输入信息的模式，一种是立即模式，另一种是缓冲模式。

e) 声音模块

实现虚拟世界中音乐的播放，主要包含音效、语音和背景音乐等。是声卡等硬件设备和声音资源文件沟通的桥梁。

f) 人工智能模块

人工智能模块为游戏中的非玩家控制角色的行为和决策提供智能支持。

g) 物理系统

物理系统用来模拟虚拟世界中物体的运动或物体之间的碰撞以及作用力情况。物理系统包括动力学模块、运动学模块和碰撞检测模块等三个模块。动力学模块为整个场景定义了一个物理模型，如整个场景的重力加速度。运动学模块主要是模拟了牛顿运动学方程。碰撞检测模块提供场景中物体之间碰撞的模拟。

h) 图形系统

图形系统是引擎的核心模块，负责处理3D世界的数据结构以及将3D世界以绚丽的画面快速的渲染出来。该模块主要包含场景(室内或室外)的管理与渲染，角色的动作管理，特效的管理与渲染(如粒子系统、水、植物等的模拟)，光照和材质的处理，LOD(Level ofDetail)的管理等。图形系统在引擎的所有模块中是最复杂的，它的强大与否直接决定着最终输出画面的质量。

i) 图形用户界面 (GUI)

图形用户界面是人与虚拟空间交流信息的窗口和渠道。

j) 场景编辑器

场景编辑器用来创建三维场景，它支持所见即所得，用户在将所有的三维模型导入到场景当中进行编辑完成后，可以将整个场景保存下来供程序调用。

k) 特效编辑器

特效编辑器专门负责往场景中加入绚丽的图形效果，包括粒子特效、天空体、光晕等，不同的特效编辑器提供的功能不尽相同。

l) 逻辑编辑器

逻辑编辑器是一个能支持UML语言的编辑器，通过它，用户在开发的时候能快速的生成游戏逻辑。经过逻辑编辑器的编辑后，场景中的元素能够按照编辑器中的要求进行相应的动作，从而减少程序开发的工作量。

## 7 设计

### 7.1 立体

#### 7.1.1 基于多边形面片的立体渲染原理

用多边形面片来建立物体的三维模型在游戏三维图形渲染中被广泛采用(通常是三角形面片)。从模型的顶点送入渲染管道到最后形成二维图像的过程如下图3所示：

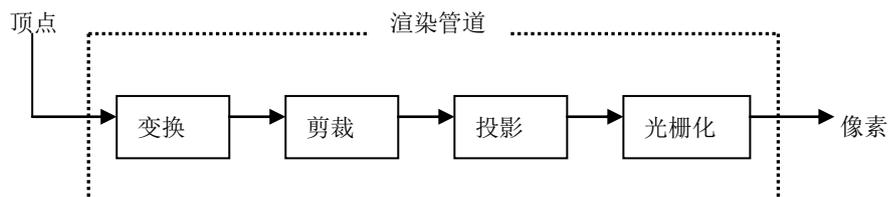


图3 渲染管道中的流程

### 7.1.2 变换

渲染管道中大部分工作是把对象模型的顶点在一个坐标系中的表示转化为另一坐标系中的表示，一般需要经过模型变换和相机变换两个过程。

### 7.1.3 剪裁

应用程序定义额外的平面对场景进行剪裁。视域剪裁通常由底层图形AP自动完成。

### 7.1.4 光栅化

从顶点组成的几何模型变换到像素的过程称为光栅化。机理与得名来源于CRT显示器的电子枪发射方式。光栅化可分为四个子阶段，即消隐、逐像素光照明计算、纹理映射和颜色融合。

### 7.1.5 消隐

消隐的目的是解决场景的可见性问题。可见性计算是指计算物体投射到投影平面有交叠的投影点。

### 7.1.6 光照计算

光照明计算技术是在物体建模时指定每个顶点的颜色和纹理坐标，在绘制时直接利用颜色和纹理映射融合为最终颜色。

### 7.1.7 纹理映射

将预生成的图像直接贴在物体表面，模拟物体表面外观。

### 7.1.8 颜色融合

颜色值将根据各自的不透明度融合出最终结果。

### 7.1.9 系统模块

- a) 系统模块。
- b) 输入子系统模块。
- c) 声音子系统模块。
- d) 计时器子系统模块。
- e) 配置子系统模块。

### 7.1.10 底层渲染模块

- a) 可见剪裁。
- b) 碰撞检测与反馈。
- c) 摄像机。
- d) 静态几何体。
- e) 粒子系统。
- f) 布告板。
- g) 网格。
- h) 天空体。
- i) 灯光。
- j) 雾。
- k) 节点阴影。

1) 输出。

## 7.2 平面

一次性渲染是在任何一个下列条件被满足之时进行的：

- a) 2D 场景结束时。
- b) 图元类型变化。
- c) 纹理变化。
- d) 渲染状态变化。
- e) 缓冲满。
- f) 更换渲染目标。

## 7.3 场景设定

### 7.3.9 一般要求

场景设定应遵循以下原则：

- a) 主体同基调之间的平衡。
- b) 注重良好的氛围的营造。
- c) 注重场景空间的表现和造型形式。

### 7.3.10 写实风格

造型样式：考虑历史的真实、时代的要求和地域的特色，还要符合一定的科学定律。

自然材质：场景中所涉及的自然属性的材料和质地宜遵循一定的自然法则，并符合常规视觉感知。

光影关系：符合科学和自然的光学规律，且要符合自然中物体被光照射后所产生的投影效果和投影角度。

色彩规律：符合光色条件下物象色调的色彩形式。

### 7.3.11 装饰风格

秩序性：将生活中的随意形体，经过删减、概括、归纳、夸张去取精，得到具有一定秩序感的形体。

主观装饰性：对场景中的装饰因素概括并强化。

示例：变形、变位、变色。

## 8 调试

### 8.1 Alpha 版本阶段

内部测试：主要是测试和完善主要功能以及有没有重大缺陷。

通过在不同点检查程序的状态，确定实际的状态是否与预期的状态一致。

### 8.2 除错调优

在 alpha 版本的基础上修改缺陷以投入外部测试。

### 8.3 Beta 版本阶段

外部测试：进一步测试和完善各项功能并预备发行，同时设计宣传品。

将程序所有可能的输入数据（有效的和无效的）划分成若干个等价类。然后从每个部分中选取具有代表性的数据当做测试用例进行合理的分类。

## 8.4 Release 版本

游戏发行：项目完成阶段，开始正式的发行游戏。

## 8.5 Gold Release 版本阶段

开发补丁：开发游戏的补丁包、升级版本，以及各种官方插件等。

## 8.6 人工智能

### 8.6.9 可视化调试

利用从游戏运行过程中获得的有关当前系统状态的可视化信息来调试程序。游戏运行时，通过输出AI角色的信息来发现问题，这些信息的输出形式包括相关AI角色的当前状态、意图、移动方向和思考过程的文本，甚至可以是游戏中影响图的变化情况。

#### 8.6.9.1 提供各种信息

可视化调试包括向游戏界面和其他可视化辅助程序输出文本。宜用线标出每个AI角色感兴趣的目标，或者突出显示导航跟踪的方法来发现导航系统中的错误。

#### 8.6.9.2 监视状态转变

当采用基于状态的决策结构时，根据AI系统输出到游戏中的可视化状态信息来监控奇怪状态的变化。使游戏把状态信息直接输出在角色附近或者它的上方，就能把数据和角色联系起来。

## 8.7 音效设定

游戏的声音准确，符合原定设计方式。

---