



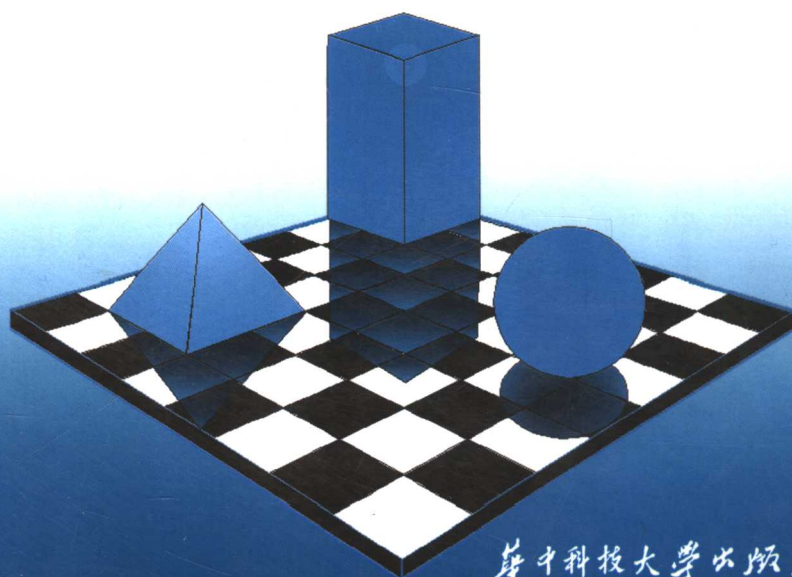
21世纪高等学校虚拟仿真技术系列教材



Vega

实时三维视景仿真技术

王 乘 李利军 周均清 陈大炜 编著



华中科技大学出版社
<http://www.hustp.com>

21 世纪高等学校虚拟仿真技术系列教材

Vega 实时三维视景仿真技术

王乘 李利军 周均清 陈大炜 编著

华中科技大学出版社

图书在版编目(CIP)数据

Vega 实时三维视景仿真技术/王乘 李利军 周均清 陈大炜 编著
武汉:华中科技大学出版社,2005 年 12 月
ISBN 7-5609-3440-4

I. V...

II. ①王... ②李... ③周... ④陈

III. Vega 技术-系统仿真

IV. TP391.9

Vega 实时三维视景仿真技术

王乘 李利军 周均清 陈大炜 编著

责任编辑:万亚军

封面设计:潘 群

责任校对:刘 飞

责任监印:张正林

出版发行:华中科技大学出版社

武昌喻家山 邮编:430074 电话:(027)87557437

录 排:华中科技大学惠友文印中心

印 刷:华中科技大学印刷厂

开本:787×1092 1/16

印张:25.75

字数:614 000

版次:2005 年 12 月第 1 版

印次:2005 年 12 月第 1 次印刷

定价:36.00 元

ISBN 7-5609-3440-4/TP·574

(本书若有印装质量问题,请向出版社发行部调换)

内 容 简 介

本书根据 Multigen-Paradigm 公司专门针对可视化仿真行业应用特点开发的实时可视化三维视景仿真软件——Multigen Vega 系列软件的应用经验编著而成。本书系统地讲述了 Vega 仿真技术的基本原理和编程应用实例,并结合应用实例阐述了 Vega 的开发函数、核心功能模块和扩展模块,最后介绍了 Vega 视景仿真综合应用技术,以及基于此基础上的高级应用技术。全书共分 6 章,主要内容包括 Vega 仿真技术概述、使用 LynX 快速定制 Vega 应用程序、Vega API 编程指南、Vega 扩展模块简介及应用实例、Vega 视景仿真综合应用实例和 Vega API 的高级应用。

本书内容循序渐进,深入浅出,对 Vega 的基本原理结合应用实例加以阐述,便于读者巩固所学知识。本书可作为高等学校虚拟现实、视景仿真相关专业本科高年级学生及研究生教材,对从事虚拟仿真技术应用研究的科技人员也有参考价值。

前 言

当前,信息技术飞速发展,三维可视化视景仿真技术在科研、生产、生活中的应用越来越广泛,各种三维视景仿真软件不断涌现。由 Multigen-Paradigm 公司专门针对可视化仿真行业应用特点开发的实时可视化三维视景仿真软件系统——Multigen Vega 系列软件以其强大的视景仿真功能得到众多用户的青睐,越来越多的科研单位和科技人员应用该软件作为三维可视化仿真应用和开发的基础。现在, Vega 已经成功应用于城市规划仿真、建筑设计漫游、飞行仿真、海洋仿真、地面战争模拟、车辆驾驶仿真、三维游戏开发等方面,并不断地向新的领域扩展。

Vega 是一套完整地用于开发交互式、实时可视化仿真应用的软件平台,其最基本的功能是驱动、控制、管理虚拟场景并支持快速复杂的视觉仿真程序,快速创建各种实时交互的三维环境,快速建立大型沉浸式或非沉浸式的虚拟现实系统。从整体上而言, Vega 具有如下特点:易用性、高效性、集成性、可扩展性、跨平台性。

针对上述情况,本书将三维视景仿真的理论与方法与 Vega 软件使用功能融为一体,并结合实例深入浅出地加以阐述。全书共分 6 章,包括 Vega 仿真技术概述、使用 LynX 快速定制 Vega 应用程序、Vega API 编程指南、Vega 扩展模块简介及应用实例、Vega 视景仿真综合应用实例、Vega API 的高级应用等内容。

三维视景仿真技术、虚拟现实技术的发展异常迅速,新的概念、方法和研究成果不断出现,各种视景仿真软件系统也不断推陈出新。为此,虽然作者在编写过程中作了很大努力,书中仍难免出现缺点和错误,敬请广大读者批评指正。

编 者

2004 年 7 月于华中科技大学

目 录

第 1 章 Vega 仿真技术概述	(1)
1.1 Vega 综述	(1)
1.2 LynX 概览	(6)
1.3 Vega API 简介	(7)
第 2 章 使用 LynX 快速定制 Vega 应用程序	(15)
2.1 小镇(Town)视景仿真实践	(15)
2.2 多通道窗口配置实现“画中画”效果	(27)
2.3 构成场景的模型对象模块	(29)
2.4 Vega 高级功能的深度使用	(58)
2.5 程序定义文件(ADF)	(78)
第 3 章 Vega API 编程指南	(87)
3.1 Vega 常用函数库	(87)
3.2 Vega API 功能函数的使用	(96)
3.3 最简单的 Vega 应用程序	(106)
3.4 基于控制台的 Vega API 编程	(112)
3.5 常用 Vega 开发函数	(140)
第 4 章 Vega 扩展模块简介及应用实例	(183)
4.1 Vega 扩展模块简介	(183)
4.2 Vega 特殊效果模拟模块应用实例	(199)
4.3 大规模数据库管理模块应用实例	(218)
第 5 章 Vega 视景仿真综合应用实例	(236)
5.1 视景仿真应用的设计与规划	(236)
5.2 使用 LynX 配置应用程序定义文件	(241)
5.3 使用 Vega API 实现高级仿真交互功能	(283)
第 6 章 Vega API 的高级应用	(316)
6.1 基于 Windows API 的 Vega 应用程序	(316)
6.2 基于 MFC 的 Vega 应用程序	(331)
6.3 基于 X-Window/Motif 的 Vega 应用程序	(389)
后记	(403)

第 1 章 Vega 仿真技术概述

本章简要回顾了 Vega 的发展历程,从整体上介绍了 Vega 的基本功能和特点,以及 Vega 的应用范围和应用前景。通过阅读本章,读者还能熟悉 Vega 图形界面——LynX 的基本使用方法,认识应用程序定义文件(ADF)的概念,了解 Vega API 的基本构成。本章的主要内容包括“Vega 综述”、“LynX 概览”、“Vega API 简介”。

1.1 Vega 综 述

1.1.1 Vega 的发展历程

Vega 的英文原意是织女星的意思。成立于 1990 年的 Paradigm Simulation Inc.首先在 SGI Performer 软件的基础上开发出了一套完整的可用于实时仿真应用的渲染引擎,随后不久发布了 Vega 的第一个商业化版本。

1998 年 9 月,Paradigm 公司与开发了 Creator 系列等著名实时三维模型制作工具的 Multigen Inc.合并,组建了现在的 Multigen-Paradigm,Inc.。合并后的公司凭借其在实时三维可视化方面强大的技术优势,很快成为世界上在虚拟现实和视景仿真及其他相关领域最成功的系统解决方案提供商之一。Vega 作为 Multigen-Paradigm 公司的旗舰产品之一,也逐渐成为可视化仿真领域的世界领先级应用软件环境。

目前,Vega 的最新版本为 3.7,又增加了许多新特性,提供了新的可选模块,并进一步提高了系统的稳定性和执行效率。同时,Vega 针对不同的用户需求,又分为多种不同版本的产品:对于不同的应用平台,Vega 分为 Vega for NT(用于 MS Windows 平台)和 Vega for IRIX(用于 SGI IRIX 平台);就 Vega 本身运行机制而言,Vega 又分为 VegaSP(仅支持单个处理器)和 VegaMP(支持多个处理器);从开发和应用的角度来说,Vega 又可分为 Vega Development(包含完整的开发包)和 Vega Runtime(仅包括运行库)两种版本。

1.1.2 Vega 的基本功能和特点

Vega 是一套完整地用于开发交互式、可视化仿真应用的软件平台和工具集,它最基本的功能是驱动、控制、管理虚拟场景并能够方便地实现大量特殊视觉和声音效果。具体而言,Vega 的核心功能模块如表 1-1 所示。

表 1-1 Vega 的核心功能模块

模块英文名	模块中文名	模块功能简介
Vega System	系统配置模块	支持多重处理,能够充分发挥运行平台的硬件资源;支持应用程序线程的处理器以及优先级的独立设置与分配;支持各种方式的帧频控制;支持非同步的数据库管理和相交检测
Vega Graphics States	图形状态模块	支持多重图形状态配置,提供了多种图形渲染选项;支持细节层次的淡入淡出效果;支持细节层次的动态比例缩放
Vega Windows	管道窗口模块	支持多窗口、多管道配置;支持动态窗口缩放和定位调整;支持分布式窗口显示;支持鼠标、键盘输入/输出事件响应;支持多种帧缓存的位面配置
Vega Channels	渲染通道模块	支持多通道渲染;支持对称、非对称的视景平载体;支持水平、垂直视野控制以及远、近裁剪平面管理;支持动态视频分辨率(DVR);支持通道倾斜变换
Vega Objects	模型对象模块	支持多种格式的三维模型数据库;支持模型数据库的优化加载;支持多种模型对象转换选项
Vega Observers	观察者模块	支持视点运动、跟踪控制;支持渲染通道共享分组;支持场景加载管理;支持立体显示模式
Vega Motion Models	运动模式模块	支持多种繁简不一的运动模式,包括 UFO(飞碟模式)、Drive(驾驶模式)、Fly(飞行模式)、Warp(沙盘模式)、Spin(旋转模式)、Missile(导弹模式)、Flight Simulator(飞行模拟器模式)等;支持用户自定义运动模式和多输入设备控制;支持运动贴地控制
Vega Paths	文件目录模块	支持文件系统的绝对目录路径和相对目录路径搜索
Vega Scenes	虚拟场景模块	支持多模型对象加载;支持虚拟场景的优化加载
Vega Database Manager	数据库管理模块	支持三种参考坐标系——平面型地球坐标系、圆球型地球坐标系和椭球型地球坐标系;支持数据库参考原点设置
Vega Players	角色对象模块	支持多角色对象控制;支持角色对象运动模式选择;支持角色对象运动参考坐标系选择
Vega Environments	场景环境模块	支持多种环境雾模型;支持一日时间控制;支持天空颜色、光源和环境效果的选择
Vega Lights	场景光照模块	支持五种光源类型,包括无限光源、局部光源、聚光光源、太阳光源和月亮光源;支持环境光和漫射光颜色设置;支持光源定位和光线衰减
Vega Environment Effects	环境效果模块	支持两种云层模型;支持天体时间控制;支持地表雾效果;支持暴风雨效果;支持动态云层效果

续表

模块英文名	模块中文名	模块功能简介
Vega Isectors	相交矢量模块	支持八种不同类型的相交测试方法和十二种相交测试计算结果;支持两种相交测试更新模式;支持世界坐标系和局部坐标系的相交测试结果输出;支持相交测试掩码设置
Vega Volumes	包容边界体模块	支持包括点集、线段和平面在内的八种类型的包容边界体
Vega Color Tables	颜色表模块	支持两种不同的颜色表插入使用模式;支持最高 64 K 大小的颜色表
Vega Input Devices	输入设备模块	支持包括数据手套、头盔显示器、轨迹球、操纵杆、跟踪控制器等在内的二十多种标准虚拟现实外部设备;支持同时使用多个输入控制设备;支持串口设备的自动检测和通信控制
Vega Pathing	路径定义模块	支持线性和样条曲线运动路径定义;支持动态和静态路径原点;支持相对路径定义
Vega Navigator	路径导航模块	支持路径控制点动作设置和控制;支持多种运动路径导航方式
Vega Custom Statistics	统计数据模块	支持多种应用程序实时运行阶段统计数据的实时显示;支持统计数据的显示效果控制;支持多种统计数据的配置

此外, Vega 还提供了多种方便实用的可视化编辑设置工具, 常用的基本 Vega 工具包括:

- Object Viewer ——模型对象查看器;
- Object Property Editor ——模型对象属性编辑器;
- Scene Viewer ——虚拟场景观测器;
- Input Device Tool ——输入设备调试工具;
- Path Tool——运动路径设置工具。

Vega 可以很好地支持多处理器、多通道渲染、多格式数据块调入和其他的附加可选模块以及第三方提供的模块产品, 支持快速复杂的视觉仿真程序, 可以快速创建各种实时交互的三维环境。Vega 还可以直接支持立体显示、Cave 系统(虚拟洞穴)、iDesk 系统(沉浸式虚拟桌面)、球面/柱面屏幕, 可以快速建立大型沉浸式或非沉浸式的虚拟现实系统。

Vega 包含了一整套可以提供最充分的软件控制同时又具有最大灵活性的 C 语言应用程序接口(API), 一个可以大大简化应用程序开发过程的图形环境用户界面——LynX, 多种方便实用的可视化编辑工具, 以及丰富的实用库函数、大量的可供使用和参考的源程序, 详尽的帮助文档资料, 等等。它把简单的工具和先进的仿真功能有机地结合在一起, 所以无论对于专业的程序员还是普通的仿真爱好者, 都可以通过使用 Vega 快速创建、编辑和运行

复杂的仿真应用程序，甚至不需要编写任何代码。

图 1-1 所示为一个典型的 Vega 应用环境示例。

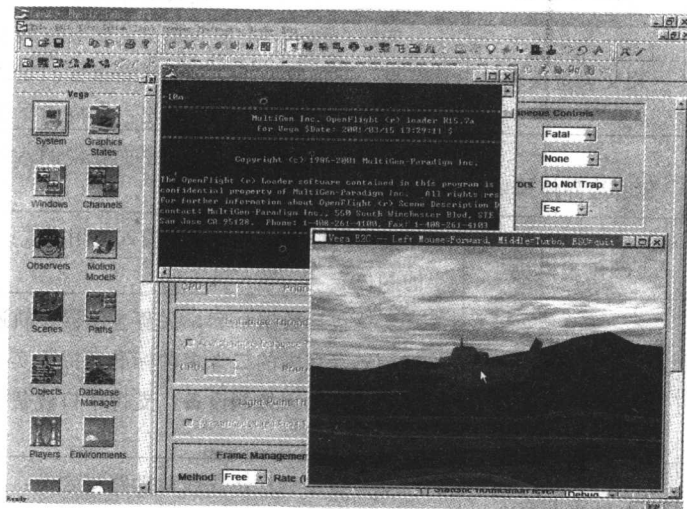


图 1-1 Vega 应用环境示例

从整体上而言，Vega 包括如下特点。

(1) 易用性：使用 LynX 图形界面可以方便快捷地设定和预览 Vega 应用程序。LynX 是一种基于 X-Motif 的点击式图形环境，用户只需利用鼠标就可以快速而显著地改变应用程序的性能和表现方式，可实时调整通道、窗口、视点、观察者等的状态，随时改变时间设定、系统配置，加入特殊效果、模型数据库等。

(2) 高效性：Vega 与仿真业界的标准文件格式 OpenFlight 无缝结合，可以对虚拟三维场景中的模型进行精确而有效的控制；在 IRIX 环境下，Vega 跟 OpenGL Performer 紧密配合，可以充分发挥 SGI 图形硬件的能力；使用 LynX 的动态预览功能，用户可以及时地看到完成操作后所产生的效果；Vega 中的统计数据模块可以实时地在终端上看到系统各部分的执行情况，以便更有效地进行系统配置。实践证明，Vega 可以显著提高工作效率，并大幅度减少源代码的开发时间。

(3) 集成性：Vega 通过精巧的设计把实时仿真应用的许多复杂烦琐的步骤清晰、紧密、高效地集成在一个框架下，使得系统集成者可以在预算内完成预定的功能和效果，并能更好地维护和支持应用系统；Vega 支持多种格式数据的调入，允许多种不同格式数据的综合显示，还提供了高效的 CAD 数据转换工具，从而把开发人员、工程师、设计师、模型制作者和管理人员有机联系起来。

(4) 可扩展性：Vega 采用了扩展性极好的模块机制来不断完善和补充自身的功能，常用的附加模块包括特殊效果模块、音响环境模块、人体动作模拟模块、面板仪表模拟模块、大地形数据库管理模块等。同时，用户也可以自己开发满足一定标准的特殊模块或使用第三方提供的专用模块，而且还能够方便地与原有的系统集成在一起使用。

(5) 跨平台性: Vega 所有的基本模块及其大多数可选模块均同时支持 SGI IRIX 平台和 Windows NT 平台, 在不同平台下开发的程序也具有相当高的兼容性。另外, 为了适应图形工作站的不同配置, Vega 分为多处理器和单处理器两种发行版本。VegaMP, 即多处理器版本, 通过在多个处理器上逻辑分配进程和系统任务, 以最大限度地利用多处理器环境, 用户也可以针对应用程序的要求分别对处理器进行自定义设定, 以满足特别的需要。VegaSP 是 Multigen-Paradigm 公司为了满足那些既需要 Vega 全部特性又只配备了单独处理器计算机的用户而特别推出的单处理器版本, 具有很高的性价比。

1.1.3 Vega 的应用范围和前景

Vega 主要用于虚拟现实、实时可视化仿真和普通的视觉模拟应用等领域。

除了上面提到的几个常用可选模块外, Multigen-Paradigm 公司还提供了和 Vega 紧密结合的特殊应用模块, 包括海浪模拟模块、红外传感器模拟模块、引航导向光源模拟模块、面板仪表模拟模块、分布式交互仿真模块等。这些附加模块可以使 Vega 很容易满足诸如航空、航海、意外事故、红外线雷达效果、高级照明系统以及人物动作等多种特殊模拟的要求。

现在, Vega 已经成功应用于建筑设计漫游、城市规划仿真、飞行仿真、海洋仿真、传感器仿真、地面战争模拟、车辆驾驶仿真、虚拟训练模拟、三维游戏开发等方面, 并不断向新的领域扩展, 如图 1-2 所示。

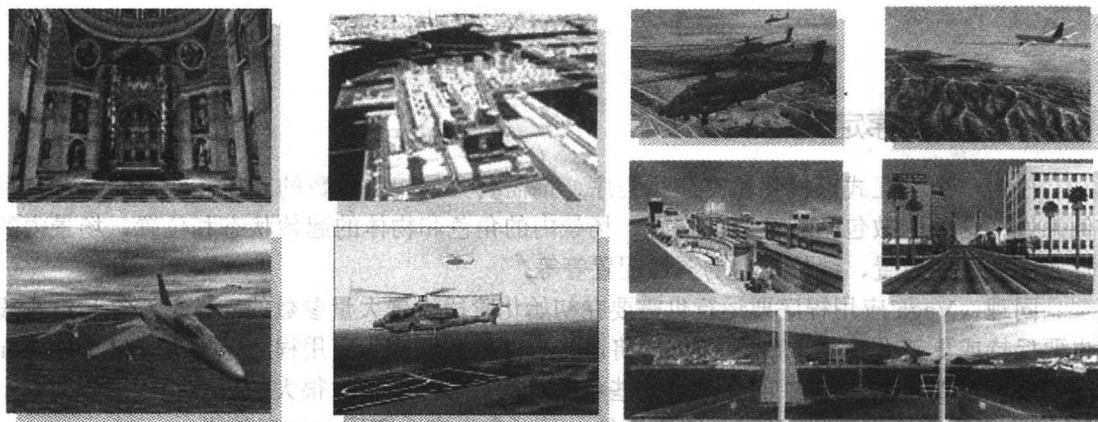


图 1-2 Vega 在各个领域的应用实例

随着虚拟仿真应用不断的大型化、复杂化和普及化, Multigen-Paradigm 公司新近开发出了新一代的仿真应用环境平台 Vega Prime。Vega Prime 虽然与 Vega 一脉相承, 但它并不是 Vega 的简单升级, 而是一种全新的软件环境: 它不是基于 SGI Performer 平台, 而是直接建立在 Multigen-Paradigm 公司自己的跨平台场景渲染引擎——VSG 之上, 并集成了全新的应用程序设置图形界面——LynX Prime, 如图 1-3 所示。

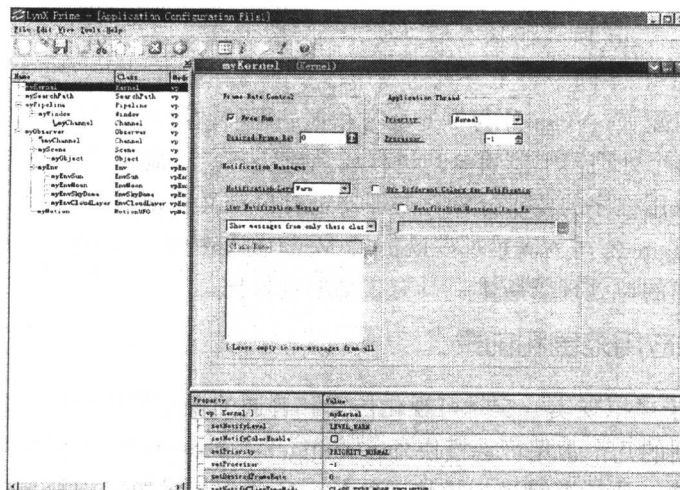


图 1-3 全新的 LynX Prime 界面

Vega Prime 可以快速开发出更加精确、更加复杂的仿真应用程序, 提供更高的稳定性和兼容性, 而且 Vega 用户可以很方便地过渡到使用 Vega Prime。由于 Vega Prime 是刚刚推出不久的产品, 所以还有相当的功能和模块需要完善。但可以预见, Vega 系列产品的用户将会涉及更多更新的领域, 应用前景也将会更加广阔。

1.2 LynX 概览

1.2.1 应用程序定义文件

任何一个交互式可视化仿真应用程序, 都需要大量相关参数的支持才能可靠运行。举例来说, 这些参数包括观察者的位置、场景内的角色和物体的起始状态和位置、场景内的灯光和环境的设置、硬件平台的属性配置等等。

同理, Vega 应用程序要运行也需要在初始化阶段设置大量参数的初始值, 在运行阶段也要保持或更新大量的参数值。Vega 将这些数据信息储存为应用程序定义文件(Application Definition File, ADF)格式的文件。这些参数对应用程序会产生很大的影响, 以至于同一个 Vega 应用程序通过解释不同的 ADF 文件, 可以产生完全不同的仿真效果。

从本质上讲, LynX 是一个功能强大的创建和编辑 ADF 文件的编辑器, 这还不包括它的应用程序动态预览功能。实际上, 用户可以用任何一种文本编辑器来查看、修改和保存 ADF 文件, 当然, 使用 LynX 操作是最直观和方便的。

1.2.2 LynX 图形界面

LynX 是 Vega 提供的一个界面友好、使用方便的点击式应用程序定义配置和动态预览工具。它的最主要功能是定义虚拟场景中的元素属性及其相互关系, 并可以实时预览参数设置的效果, 最后生成用于 Vega 程序的应用程序定义文件——ADF 文件。

LynX 的用户界面遵循标准的窗口用户界面规则, 非常直观和灵活。所以, 只要是熟悉窗口操作的用户, 都可以很快掌握它的使用方法。图 1-4 所示为 LynX 在 Windows NT 平台下的典型布局。

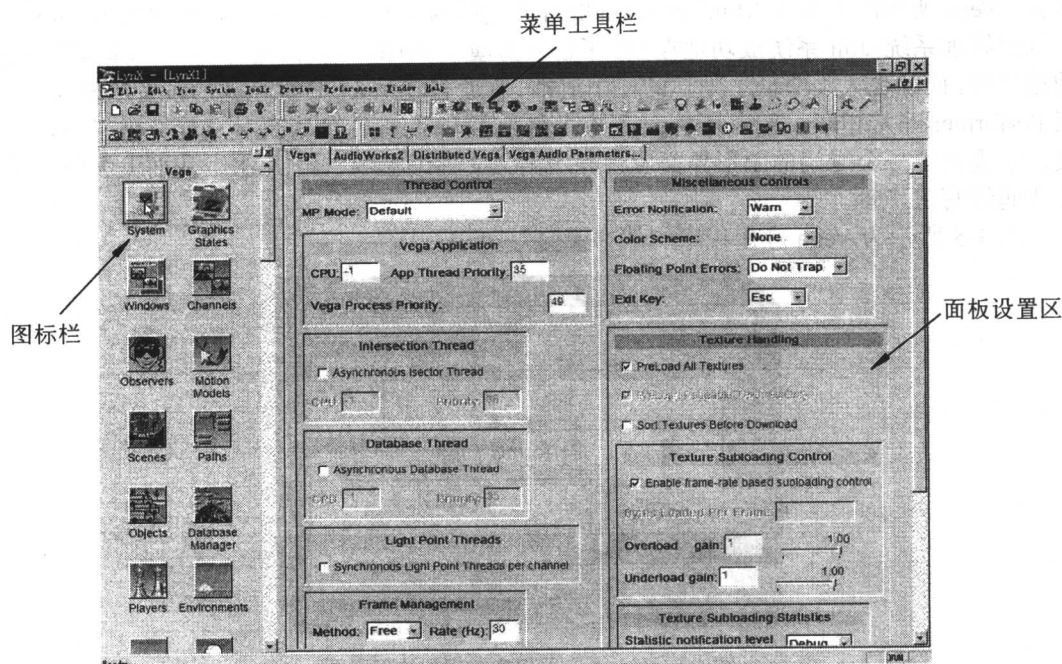


图 1-4 LynX 用户界面

LynX 用户界面大致可分为三个部分: 菜单工具栏、图标栏和面板设置区。菜单工具栏位于 LynX 窗口的顶部, 注意, 菜单项会根据当前的操作对象有所变化。在默认状态下图标栏位于 LynX 窗口的左侧, 图标栏实际上是各模块所提供的类的图标化列表。面板设置区占据了 LynX 窗口的绝大部分位置, 用于显示和调整与当前类相关的所有参数集。

LynX 在 SGI IRIX 平台上的界面基本上是类似的, 不同的是窗口样式遵循的是 X Window 标准。

1.3 Vega API 简介

1.3.1 Vega 应用程序的实现架构

从 Vega 软件系统的底层实现框架来看, Vega 实际上是建立在特定的场景图(Scene Graph)之上的, 而场景图管理系统本身又建立在诸如 OpenGL 这样的标准图形库之上。所谓的场景图, 是从数据结构的角度来看场景数据的, 这里的“图”是指队列、树、图等数据结构术语中的图, 具体而言是一种有向无环图(Directed Acyclic Graph, DAG)。这个有向无环图就被称为“场景图”。场景图从整体上描述了虚拟场景中各个元素之间的相对位置及其相互关系。场景图由各种节点组成, 一般来说, 场景中的模型对象用场景图中最下层的叶节点描述, 而

这些叶节点上层的各种节点则负责完成包括定位、显示或隐藏模型对象、对节点分组等在内的各种功能,实际上对虚拟场景实时仿真的过程就是对场景图节点进行特定方式的遍历绘制。

在 SGI 平台上, Vega 所依附的场景图管理系统就是 Performer, 而对于 Windows 平台而言, Vega 所依附的是由 Multigen-Paradigm 专门针对 NT 平台开发的一套被称为“Jolt”的场景图管理系统。Jolt 系统成功地在 PC 平台下扮演了 Performer 的角色,这不仅提高了 Vega 的通用性,而且也大大降低了用户的使用成本。有过 Performer 开发经验的读者会发现, Vega 跟 Performer 非常相似,实际上 Vega 基本上继承了 Performer 的大部分功能并做了一定的扩展,它提供了一个专门的图形界面应用程序定义文件编辑器——LynX,从而大大降低了程序代码编写工作量。

图 1-5 所示为 Vega 应用程序的基本架构。

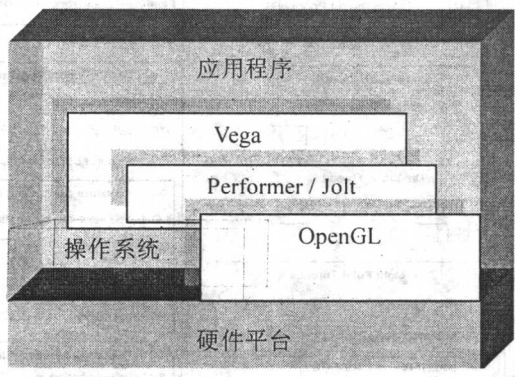


图 1-5 Vega 应用程序实现架构示意图

1.3.2 OpenGL 应用程序与 Performer 函数库

1. OpenGL

OpenGL 不是一种计算机编程语言,而是一个功能强大的图形库(Graphics Library),本质上就是计算机图形硬件的软件接口。OpenGL 图形库中包含了大量功能强大的图形函数,集成了所有曲面造型、图形变换、光照、材质、纹理、像素操作、融合、反选择、雾化等复杂的计算机图形学算法。对于计算机图形应用开发人员来说,OpenGL 就是一套独立于硬件设备、窗口系统和操作系统的应用程序编程接口(API)。

图 1-6 所示为一个典型的基于 OpenGL 的应用程序系统架构。

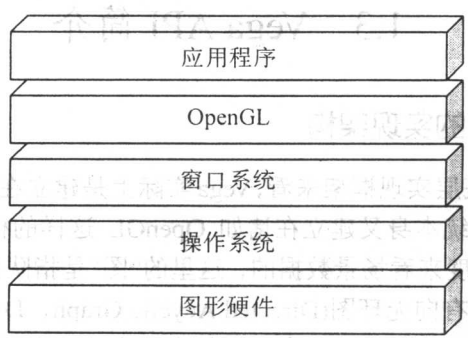


图 1-6 OpenGL 应用框架

图形标准或图形应用程序编程接口在图形应用领域有着重要的地位,它不仅加速了图形应用程序尤其是三维图形应用程序的开发,而且还使得图形应用程序具有更好的移植性;历史上曾出现的图形标准(或 API)包括 Core、GKS、PHIGS、PEX、GL、Dore、RenderMan、Hoops、OpenGL 等等。经过竞争与淘汰,OpenGL 成为国际上公认的 3D 图形工业标准,在 Unix 与 PC 平台得到广泛应用。OpenGL 的前身是 SGI 公司为其图形工作站设计的一个图形开发软件库 IRIS GL。凭借其优越的图形性能,IRIS GL 受到了计算机图形业界的一致推崇。

但由于 IRIS GL 不是一个开放的标准,它只能在 SGI 公司的图形工作站上运行,所以限制了其进一步发展和应用。SGI 公司有针对性地对 GL 进行了改进,特别是扩展了 GL 的可移植性,使之成为一个跨平台的开放式图形编程接口,从而演变成现在的 OpenGL。目前,已有相当多的计算机行业软硬件厂商和相关研究机构都支持 OpenGL 作为业界标准的图形软件接口,并且由一个包括 SGI 公司和 Microsoft、Intel、IBM、Compaq、DEC 等在内的软、硬件制造商成立的 OpenGL 体系结构评审委员会(OpenGL Architecture Review Board, OARB)负责管理和完善 OpenGL 规范,从而充分保证了 OpenGL 的独立性、开放性、前瞻性、可靠性、稳定性、兼容性和可扩展性。

OpenGL 实现的主要功能包括以下几种。

(1) 几何建模: OpenGL 提供绘制点、线、多边形等几何图元的函数,通过这些基本函数可以绘制出任意的二维和三维物体,同时 OpenGL 图形库还提供了诸如球体、锥体、多面体,甚至茶壶等复杂三维物体模型以及 Bezier、NURBS 等复杂曲线和曲面的绘制函数。

(2) 坐标变换: OpenGL 提供了基本的模型平移、旋转、缩放、镜像变换操作函数,以及用于投影变换的正射投影和透视投影变换函数,通过这些函数来放置三维场景中的物体,并实现从三维场景到二维图像的变换。

(3) 颜色模式: OpenGL 支持 RGBA 模式和颜色索引(Color Index)两种不同的颜色模式,两种颜色模式有各自的运用范围,具体采用哪种颜色模式,可以根据具体的应用需求和运行平台的图形硬件配置来选择。

(4) 光照模拟: OpenGL 支持环境光(Ambient Light)、漫反射光(Diffuse Light)、辐射光(Emitted Light)、镜面光(Specular Light)四种常见光源以及多种光照模型,可以逼真地模拟出各种真实的光照效果。

(5) 材质模拟: OpenGL 用光的反射率来表示物体的材质,材质和光照一起决定了人眼所观测到的物体的颜色,物体的材质特性直接影响三维物体的最终绘制效果。

(6) 纹理映射: OpenGL 的纹理映射功能允许把真实图像映射到三维物体的表面来表现物体的细节,通过多种不同的纹理映射方式,可以逼真地绘制出足以乱真的三维物体。

(7) 位图绘制: OpenGL 提供对位图和图像的绘制渲染函数,可以方便地生成各种字符、字体。另外,图像不仅可以直接被绘制到屏幕上,还可以将其作为纹理映射到物体表面上。

(8) 图像增强: OpenGL 除了支持基本的图像显示和绘制功能外,还特别支持融合(Blending)、反走样(Antialiasing)、雾化效果(Fog)等多种图像增强功能,能够使绘制的三维图像产生真实感极强的视觉效果。

(9) 平滑动画: OpenGL 可利用各种帧缓存技术,尤其是双缓存技术,从而能够生成平滑的连续动画效果,即把帧缓存分为前台缓存和后台缓存,分别负责绘制图像和显示图像,并能在动态渲染的过程中互相交换。

图 1-7 所示为 OpenGL 的基本工作流程。

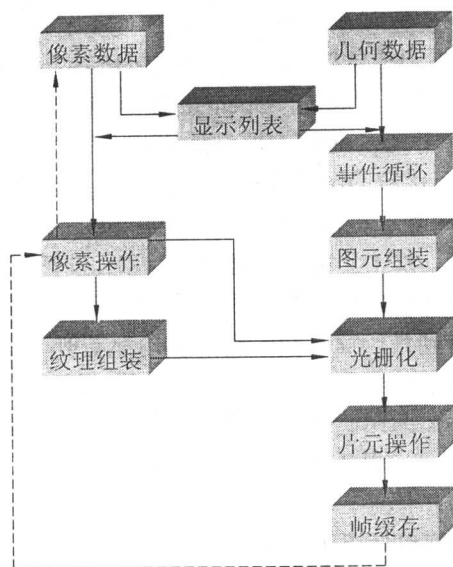


图 1-7 OpenGL 的基本流程

作为标准的图形软件接口，OpenGL 提供了数百个库函数，可以方便地绘制具有真实感的二维和三维图形；但由于 OpenGL 与窗口系统无关，不提供任何交互手段，所以必须由程序员自己编写所有的交互功能。另一方面，OpenGL 的编程接口是低级的 C 函数，不提供可复用的对象库或者应用程序框架，所以直接使用 OpenGL 开发交互式应用程序的工作效率并不高。

许多软件厂商鉴于 OpenGL 强大的图形处理能力，在不同的应用领域中相继开发出众多基于 OpenGL 的图形应用软件产品，其中比较著名的软件产品包括：用于三维动画制作的软件——Maya、SoftImage 3D；用于视景仿真建模的软件——Multigen Creator；用于可视化仿真驱动的软件——World Tool Kit、VTK；用于计算机辅助设计和制造的软件——Pro Engineer、Alias Studio Tools；用于地理信息系统的软件——Arc/Info 等等。同时，SGI 公司也不断推出以 OpenGL 为基础的高级开发工具包，以满足对图形工具性能日益增长的需求，这些高级图形开发工具包包括 Cosmo3D、Open Inventor、OpenGL Performer(可视化仿真驱动软件)、OpenGL Optimizer(高端图形应用工具箱)等等。

2. Performer

OpenGL Performer 最初是 SGI 公司开发的一个可扩展的高性能实时三维视景开发软件包，它提供了一套完整的、用 C/C++ 语言写成的应用程序接口，并且可以与 IRIS GL 和 OpenGL 相连接，可以运行于所有的 SGI 图形计算机使用的 IRIX 操作系统之上。OpenGL Performer 根植于图形化的 IRIX 操作系统，能够充分发挥 SGI 图形硬件的强大图形功能，因而具有超强的视景生成能力，并广泛应用于可视化仿真、虚拟现实、视频广播、游戏娱乐以及计算机辅助设计等领域。

图 1-8 所示为 OpenGL Performer 的一个简单应用示例。

