



林 锐 蔡文立 著

Scientific Visualization System Design

**微机科学**

**可视化系统设计**

西安电子科技大学出版社

# 微机科学可视化系统设计

林 锐 蔡文立 著

西安电子科技大学出版社

1996

(陕)新登字 010 号

## 内 容 简 介

科学可视化(SV)是近几年国际上迅速发展起来的研究领域,它已在科学计算与工程设计中取得了卓著成效。在微机上研究 SV 软件系统设计的一般性问题是富有开创性的工作,它无疑将推进 SV 的普及。

结合计算机辅助软件工程的研究成果,本书全面论述了在微机平台上有效实现“大幅度提高科学可视化应用软件生产率与质量”的方法学与关键技术。全书分 4 部分。第一部分讲述基本概念与方法学;第二部分结合随书赠送的软件“微机科学可视化软件开发工具 2.0”详细介绍 DOS 图形用户界面设计、DOS 与 Windows 环境可视化算法设计、程序自动生成,以及最新的 Intel NSP 三维真实造型技术;第三部分为应用篇,使用该软件设计集成电路功能成品率仿真系统的可视化环境;第四部分为附录,给出了随书软件的功能接口。

本书思想新颖,风趣易懂,读者对象为广大科技工作者、学生、程序员。随书软件(学习版)曾获第二届中国大学生应用科技发明大奖赛二等奖,可用于快速开发图形 CAD、科学与工程可视计算和商业图形统计等软件。

## 微机科学可视化系统设计

林 锐、蔡文立 著

责任编辑 徐德源

西安电子科技大学出版社出版发行

地址:西安市太白南路 2 号 邮编:710071

陕西地质印刷厂印刷

陕西省新华书店发行 各地新华书店经售

开本 787×1092 1/16 印张 17 4/16 字数 400 千字

1996 年 2 月第 1 版 1996 年 2 月第 1 次印刷 印数 1—5 000

ISBN 7-5606-0414-5/TP·0168 定价:29.80 元(含软盘)

## 前 言

先想想吧，还有什么比图形更吸引你的？

我承认我是喜欢图形的，却不知道怎么做迷上了“可视化”。一年前才听说过这个名字，那时我正在做本科毕业设计，被集成电路统计优化的数值计算弄得晕头转向（在这之前，我做了两年有关集成电路可靠性分析的数值计算，事情是一样的糟糕）。我发现用图形来表征、解释一堆堆数据可以让自己不再迷糊下去，那感觉就像刚睡醒时洗凉水一样。之后，我在想是否可以把集成电路工业大生产的宏观机制引入到科学可视化(SV)软件的开发中。如果每一个新的软件系统都可以由软件集成电路构造而成，岂不省事？至少我可以不必每天都被那些隐藏于几万行程序中的 Bug 戏弄着。我把那种设想称为“科学可视化计算机辅助软件工程——SV CASE”。

我居然就津津有味地做下去了。几个月后，“基于微机的科学可视化计算机辅助软件工程”在第二届中国大学生应用科技发明大奖赛中获得了二等奖。接下几个月，与“可视创意”软件小组的伙伴们忙碌着开发“微机科学可视化软件开发工具——SV CASE/Aids for Windows & DOS Ver 1.0，接受“中国首届 PC 机应用软件设计大赛”的考验（不久我们成为 Intel NSP 专门小组的成员）。再过几个月，也就是一年后的现在，我们拥有了一个 60 000 行 C++ 代码的 SV CASE/Aids for Windows & DOS Ver 2.0，同时写了这本书。事情就这么简单。

SV CASE 是基于有效实现科学可视化应用“学科无关性”的软件工程方法学，它采用系统构件化与集成构造策略，可大幅度提高科学与工程可视化应用软件的生产率与质量。SV CASE/Aids for Windows & DOS Ver 2.0 则是 SV CASE 的技术产物，直接目的是用来快速构造微机平台的科学与工程可视化软件，主要应用在：

- (1) 计算机辅助设计(CAD)与图形学(CG)
- (2) 科学与工程计算(数值分析)可视化
- (3) 商业图形统计

等领域的软件开发。

本书的理论核心是 SV CASE 方法学，系统设计与技术实现是以 SV CASE/Aids for Windows & DOS Ver 2.0 为背景的。我一直认为软件是优秀的，并且是有创新意义的。本书也是目前国内第一本关于微机科学可视化系统设计的专著。全书风趣易读，并随书赠送软件的学习版，即使是初次接触科学可视化系统的读者亦会很快进入主题，尝试开发应用软件（在你之前已有很多人实践过，并已成为熟练的用户）。

从事教学、科研、公益事务等非商业应用的用户可以非常优惠地购买到软件标准版，学生另有专用版（详见附录“软件销售与服务”），希望这样做能获得更好的社会效益。我乐意为您解答任何关于本书的疑问，更希望能得到批评与有益的建议（忘了告诉你，我现在还是学生，我的伙伴们也全是非计算机系学生）。

全书分 3 篇：第一篇（第一章与第二章）主要论述科学可视化的概念与 SV CASE 方法

学。第二篇(第三章至第七章)论述微机科学可视化软件开发工具的设计方法与技术。主要专题有人机界面设计美学、图形用户界面程序设计、Windows 与 DOS 环境的科学可视化算法程序设计、科学可视化程序自动生成以及 Intel NSP 三维真实造型技术。第三篇(第八章)以集成电路功能成品率仿真软件的可视化环境设计为例,介绍如何使用 SV CASE/Aids for Windows & DOS Ver 2.0 来开发应用软件。附录详细说明了 SV CASE/Aids for Windows & DOS Ver 2.0 的功能接口,以及关于该软件销售与服务的信息。本书第一章与第七章由蔡文立执笔,其余各章由林锐执笔,全书由蔡文立主审。

除了简要评注本书内容外,我想在此谈谈对科学可视化系统研究的几个观点,以及 SV CASE 的研究经历(这有益于读者对本书及软件的理解)。

### 一、对科学可视化系统研究的几个观点

#### 1. 在微机上开展科学可视化工作的必要性与重要性

国际上对可视化的研究大多是在图形工作站上开展的。国内在这方面的工作仍属起步阶段,多数研究单位与用户并没富裕到可以随时买台 SGI 工作站的程度,硬件的落后无疑会严重阻碍对可视化的研究与应用。而微机在国内已经普及,目前高档微机的软硬件系统的性能已逼近中档工作站,中小型的可视化工作完全可以从工作站移至微机上进行。因此,从国情出发,在微机上开展科学可视化工作是十分必要的。在微机上研究可视化系统设计的一般性问题是富有开创性的工作,它将推进可视化的普及。

#### 2. 继承 DOS 图形财富,向 Windows 和谐发展

Windows 无疑是 Intel 体系结构的微机操作系统的主流,但对 Windows 太多的赞誉使得人们盲目地轻视了 DOS 下的工作。科学与工程研究的基本出发点是务实,一味追求新事物并不会取得成效(不同于时装与化妆品)。可以说,目前 Windows 环境下最流行的是字处理、办公、财务管理等一类通用性、商业性较强的软件,而科学与工程类软件所占比重则太少。由于科学与工程面临的问题多种多样,不可能寻求一种高通用的软件来解决所有问题。糟糕的是,真正能结合任务开发 Windows 应用软件的人要比声称会使用 Windows 的人少得多。工程实践与科学探索随时都要应用熟悉的工具与开发新工具,在目前这种情况下轻易地抛弃 DOS,那是一种愚蠢的做法(如同倒洗澡水时把婴儿一起泼掉)。软件系统设计者要考虑使当前的工作适应于现在与未来的需求,必须走继承与发展的道路。

SV CASE/Aids for Windows & DOS Ver 2.0 在系统设计时已充分考虑到继承 DOS 图形财富、向 Windows 和谐发展的问題,并得以很好地实现。用它开发 DOS 或 Windows 环境的科学可视化应用软件,无论是人机交互还是图形算法的实现,都具有良好的一致性(可移植性)。用户可以根据任务需求,开发相应的 DOS 版本或 Windows 版本软件。这样既重用了已有财富,又顺应了发展潮流。

#### 3. 图形软件的设计要追求美

这本是不用多作解释的道理。但总有人担心“花很多精力、物力让界面那么漂亮、图形那么逼真是否值得?”这种问题不能强求别人与你一致,最好就是自问自答(我当然回答值得,没什么道理,就是喜欢)。

### 二、SV CASE 的研究经历

一个偶然的机会,我听说了“可视化”(在与浙江大学 CAD & CG 国家重点实验室蔡文立老师的通信中),并迷上了它。由于几年勤工俭学在软件开发方面尝到了够多的苦头,所

以对软件工程有了一点感受。在1994年9月,我开始研究SV CASE,并与马佩军合作参加3个月以后的第二届中国大学生应用科技发明大奖赛(又过上了打仗般的日子)。困难真是无穷多,没有资料、没人指导、没有经费、时间太少,最要命的是我还没想明白什么是SV CASE,怎样去实现。我用了两个月才琢磨出SV系统构件化与集成构造策略。那点子真有用,我只花了50个小时,竟开发出5000行C++代码的Windows环境二维、三维可视化算法核心库(真是人逼急了,什么事都做得出来)。在剩下的一个月,与马佩军(他编程时不太像人,像指针,我嫉妒死了)一起开发了SV CASE的软件模型(约25000行C++代码),之后在竞赛中得了二等奖。这期间一直得到蔡文立老师的鼓励(我们是成为朋友后才认识的)。

1995年春季,与戴玉宏、马晓宇、曾浩垠等同学成立了可视创意软件小组,着手开发SV CASE/Aids for Windows & DOS Ver 1.0,并参加中国首届PC机应用软件设计大赛(Intel主办)。我们确实好好地磨炼了一番,那时的口号是“让春天消失”,晚上还要咬尖辣椒来提神。1995年8月开始开发2.0版,同时与蔡文立老师合著本书(得到自然科学基金资助)。

## 致 谢

SV CASE/Aids for Windows & DOS Ver 2.0的开发与本书的编写几乎全是靠学生们的力量完成的,我无比感激同学们的热情相助。戴玉宏、史江一、马晓宇、马佩军参与了软件核心系统的开发。冯筠、马晓慧等同学参与了软件维护工作。全书文字由马晓慧、马晓华、芪凤仪等同学输入,马晓慧设计了全书插图,并负责排版。

在微电子所勤工俭学将近4年,我总是以办公室为家。老师们的关心与帮助,使我可以投入于所热爱的研究之中。我的导师郝跃教授给予了最多的支持和鼓励,并引导我将科学可视化系统设计与集成电路可制造性工程结合,提前进入硕士毕业设计工作(我享受了开放教育)。我曾师从贾新章教授研究集成电路可靠性,这使我可以从另一种角度认识软件的质量与可靠性问题,为现在的CASE研究提供了工程背景。我将一直感激周维真老师对我的软件启蒙教育(可惜我们只相处一年),他的严谨、负责、劳累曾让我不忍心问他太多的问题。对SV CASE的研究得到了浙江大学CAD & CG国家重点实验室石教英教授、蔡文立老师的帮助。Intel技术发展(上海)有限公司提供了NSP 3DR材料。本书的出版得到西安电子科技大学出版社的支持。

与可视创意软件小组的伙伴们结下了深厚的友谊,他们的勤奋与乐观一直鼓励着我。可我不知将以什么样的心情与他们分别,愿长相忆,再相聚,人若依旧。

感谢父母为我的学业倾注了心血,感谢冯筠、谢伟、梁洁给予的关心与勉励,感谢所有帮助过我的老师与同学。

林 锐

西安电子科技大学微电子所

1995.11

## 本书导读

当你有心情翻看这本书时,如果觉得新鲜并大致有点儿喜欢,请你不要错过更好的感觉。

现在假定你是本书的读者(当然也是随书软件的用户),如果你也和作者一样处于入门状态,那就不要强迫自己憋一口气看完。推荐的方法是:首先浏览本书,了解大概就行,记着同时尝试使用随书软件的演示程序、学习程序等可看可感受的部分。因为软件要比书更形象直观,使用它有助于读者快速进入主题。

接着是顺序阅读本书,主要是了解 SV CASE 方法学与软件的关键技术。这时将随书软件与本书配合是绝对必要的,因为书的篇幅有限,技术细节与更多的实例只能从软件中得到。如果对 SV CASE 方法学的理解一时不能明明白白,可以先不求甚解,因为在实践中你会自然体会到的。

最后,你应该尝试着开发几个应用软件,不然真的是浪费了三分之一的书价。这时你干脆把本书当作使用手册看,用到什么就查什么。本书附录提供了软件的功能接口,如果你觉得还不够充分,可以查看随书软件中的一大堆头文件(已经汉化)。

本书各章的要点如下:

**第一章 科学可视化导论**,本章主要是讲述概念。其中科学可视化神经机理(1.2节)与科学可视化内涵与研究规律(1.5节)是值得好好体会的。本章有较多的例子,容易理解,其中虚拟现实(1.4节)与微机可视化软件开发平台(1.7节)的内容相对较新。读者不要把科学可视化看成是一种纯理论,科学可视化只有与具体应用结合才能取得成效,因此阅读本章时请不要局限于作者的思路。

**第二章 科学可视化计算机辅助软件工程**,本章是全书的理论核心。本章 2.1 节至 2.4 节是 SV CASE 原理的论述,读者应该细看。如果读者是程序设计爱好者,阅读软构件的测试(2.6节)与软构件的维护(2.7节)是会有共鸣的(如果你还没有软件测试与维护的体会,这可是技术上的一大失误了)。倘若读者是系统管理员,并想在自己的工作域内实现大规模软件重用(顶好的想法),本章软构件库管理(2.5节)与 SV CASE 的非技术障碍与对策(2.8节)或许对你有帮助。

**第三章 微机科学可视化软件开发工具系统综述**,读者对随书软件的了解就从这里开始。本章人机界面设计美学(3.6节)是很值得品味的。软件开发爱好者之所以能痴迷地投入工作,是因为有很多美可以享受,界面之美就是其一。

**第四章 DOS 环境图形用户界面程序设计**,采用本章所论述的系统设计与关键技术,可以开发出一个在视感与效率上可与 Windows 3.1 相媲美的图形用户界面系统。对于从事科学与工程应用的人员而言,拥有一个独立自主的图形用户界面系统是很有必要的。对 DOS 有好感的读者请看看本章引言(4.1节),4.2节至 4.12 节请配合随书软件阅读。

**第五章 科学可视化算法程序设计**,本章论述如何设计兼容 Windows 与 DOS 的科学可视化算法核心库。这些二维三维可视化算法程序不仅实用而且美观。5.2 节至 5.9 节请

配合随书软件阅读。

第六章 科学可视化程序自动生成，本章主要论述程序自动生成的一般方法与技术。其中可视编程与仿真运行集成环境的设计是核心内容，6.1 节至 6.4 节请配合随书软件阅读。

第七章 Intel NSP 三维真实造型技术，本章论述 Intel 最新主体信号处理平台的三维真实建模与三维真实显示技术。本章取材于 Intel 架构实验室发行的 3DR SDK 光盘。

第八章 集成电路功能成品率仿真软件的可视化环境设计，本章是科学可视化应用软件开发实例。大多读者可能与集成电路可制造性的研究无关，但这课题确实太有意思，多了解一点也许是有益的。实际上，可视化技术用于微观领域的研究是卓有成效的。

本书附录提供了软件功能接口，注意，这部分内容是用于查看而不是用于阅读的。

当你觉得已经相当了解本书及随书软件时，请不要忘记下一步应该使用正版软件，成为注册用户。如果对本书与软件有什么难以理解的，或有什么建议等等，请及时告诉作者。



# 目 录

## 第一章 科学可视化导论

|                            |    |
|----------------------------|----|
| 1.1 科学可视化的诞生、意义与推动力 .....  | 1  |
| 1.2 科学可视化的神经机理 .....       | 2  |
| 1.3 可视化技术的一些应用 .....       | 4  |
| 1.3.1 SGI 的贡献 .....        | 4  |
| 1.3.2 影视业 .....            | 4  |
| 1.3.3 金融可视化 .....          | 5  |
| 1.3.4 石油工业 .....           | 6  |
| 1.3.5 飞机设计 .....           | 6  |
| 1.4 虚拟现实 .....             | 7  |
| 1.4.1 VR 的特征与历史 .....      | 7  |
| 1.4.2 VR 的感知原理 .....       | 8  |
| 1.4.3 VR 的一些应用 .....       | 9  |
| 1.4.4 VR 的未来 .....         | 9  |
| 1.5 科学可视化的内涵与研究规律 .....    | 10 |
| 1.5.1 科学可视化的内涵 .....       | 10 |
| 1.5.2 科学可视化的研究规律 .....     | 10 |
| 1.6 工作站可视化软件开发平台 .....     | 12 |
| 1.7 微机可视化软件开发平台 .....      | 13 |
| 1.7.1 契机与行动 .....          | 13 |
| 1.7.2 Intel NSP 平台概览 ..... | 14 |
| 1.8 科学可视化研究现状 .....        | 14 |
| 1.9 小结 .....               | 15 |

## 第二章 科学可视化计算机辅助

### 软件工程

|                                       |    |
|---------------------------------------|----|
| 2.1 SV CASE 思想与体系结构 .....             | 16 |
| 2.2 软件工程的核心问题 .....                   | 17 |
| 2.2.1 提高软件生产率与质量的<br>途径——软件重用 .....   | 17 |
| 2.2.2 软件质量因素 .....                    | 18 |
| 2.3 SV CASE 的软件系统构件化<br>与集成构造策略 ..... | 19 |
| 2.3.1 走上构件化之路 .....                   | 19 |
| 2.3.2 软件系统集成构造策略 .....                | 20 |
| 2.4 软构件可制造性 .....                     | 21 |
| 2.4.1 软构件的存在性 .....                   | 21 |
| 2.4.2 软构件特征 .....                     | 22 |

|                             |    |
|-----------------------------|----|
| 2.4.3 软构件设计原则 .....         | 24 |
| 2.4.4 技术可行性 .....           | 25 |
| 2.5 软构件库管理 .....            | 26 |
| 2.5.1 重用管理模式 .....          | 26 |
| 2.5.2 接口要求 .....            | 27 |
| 2.5.3 实例与解释 .....           | 27 |
| 2.6 软构件的测试 .....            | 27 |
| 2.6.1 对测试的理解 .....          | 2  |
| 2.6.2 测试人员的选择 .....         | 28 |
| 2.6.3 全面测试 .....            | 28 |
| 2.7 软构件的维护 .....            | 29 |
| 2.7.1 维护常识 .....            | 29 |
| 2.7.2 软构件排错 .....           | 30 |
| 2.8 SV CASE 的非技术障碍与对策 ..... | 30 |
| 2.9 小结 .....                | 32 |

## 第三章 微机科学可视化软件开发

### 工具系统综述

|                                        |    |
|----------------------------------------|----|
| 3.1 软件使用说明 .....                       | 33 |
| 3.1.1 运行环境 .....                       | 33 |
| 3.1.2 安装 .....                         | 33 |
| 3.1.3 使用入门 .....                       | 33 |
| 3.2 主要内容与功能特性 .....                    | 35 |
| 3.3 软件目录结构及要点 .....                    | 35 |
| 3.4 Borland C++ 编译器配置说明 .....          | 37 |
| 3.5 命名约定 .....                         | 37 |
| 3.6 人机界面设计美学 .....                     | 38 |
| 3.6.1 界面设计中美的需求与<br>导向作用 .....         | 39 |
| 3.6.2 界面美的内涵 .....                     | 39 |
| 3.6.3 界面美与可视化——一种界面<br>设计潮流的美学剖析 ..... | 41 |
| 3.6.4 界面设计广义美 .....                    | 42 |

## 第四章 DOS 环境图形用户界面

### 程序设计

|                   |    |
|-------------------|----|
| 4.1 引言 .....      | 43 |
| 4.2 界面基础模块 .....  | 44 |
| 4.2.1 功能与结构 ..... | 44 |
| 4.2.2 设计概要 .....  | 45 |

|                     |    |
|---------------------|----|
| 4.2.3 功能演示 .....    | 45 |
| 4.3 鼠标模块 .....      | 47 |
| 4.3.1 功能与结构 .....   | 47 |
| 4.3.2 设计概要 .....    | 48 |
| 4.3.3 功能演示 .....    | 49 |
| 4.4 扩展内存管理模块 .....  | 49 |
| 4.4.1 功能与结构 .....   | 50 |
| 4.4.2 设计概要 .....    | 51 |
| 4.4.3 功能演示 .....    | 51 |
| 4.5 窗口管理模块 .....    | 51 |
| 4.5.1 功能与结构 .....   | 51 |
| 4.5.2 设计概要 .....    | 52 |
| 4.5.3 功能演示 .....    | 53 |
| 4.6 汉字显示模块 .....    | 54 |
| 4.6.1 功能与结构 .....   | 54 |
| 4.6.2 设计概要 .....    | 55 |
| 4.6.3 功能演示 .....    | 56 |
| 4.7 图像存取模块 .....    | 57 |
| 4.7.1 功能与结构 .....   | 57 |
| 4.7.2 设计概要 .....    | 59 |
| 4.7.3 功能演示 .....    | 59 |
| 4.8 对话模块 .....      | 60 |
| 4.8.1 功能与结构 .....   | 60 |
| 4.8.2 设计概要 .....    | 61 |
| 4.8.3 功能演示 .....    | 62 |
| 4.9 控制模块 .....      | 63 |
| 4.9.1 功能与结构 .....   | 63 |
| 4.9.2 设计概要 .....    | 64 |
| 4.9.3 功能演示 .....    | 66 |
| 4.10 主控界面模块 .....   | 68 |
| 4.10.1 功能与结构 .....  | 68 |
| 4.10.2 设计概要 .....   | 71 |
| 4.10.3 功能演示 .....   | 71 |
| 4.11 对话集成构件模块 ..... | 73 |
| 4.11.1 功能与结构 .....  | 73 |
| 4.11.2 设计概要 .....   | 74 |
| 4.11.3 功能演示 .....   | 76 |
| 4.12 交互式绘图模块 .....  | 79 |
| 4.12.1 功能与结构 .....  | 79 |
| 4.12.2 设计概要 .....   | 80 |
| 4.12.3 功能演示 .....   | 81 |
| 4.13 小结 .....       | 82 |

## 第五章 科学可视化算法程序设计

|                                        |     |
|----------------------------------------|-----|
| 5.1 引言 .....                           | 83  |
| 5.2 DOS 环境科学可视化算法<br>核心库设计概要 .....     | 83  |
| 5.3 分形图形程序设计 .....                     | 84  |
| 5.3.1 基本概念 .....                       | 85  |
| 5.3.2 分形实例 .....                       | 86  |
| 5.4 二维数学与统计图形程序设计 .....                | 88  |
| 5.4.1 曲线与散点图组 .....                    | 88  |
| 5.4.2 条形图组 .....                       | 91  |
| 5.4.3 饼形图 .....                        | 93  |
| 5.4.4 等高线 .....                        | 94  |
| 5.5 三维图形程序设计 .....                     | 96  |
| 5.6 Windows 环境科学可视化算法<br>核心库设计概要 ..... | 96  |
| 5.6.1 WINSV LIB 设计思想 .....             | 97  |
| 5.6.2 WINSV LIB 对设备资源的<br>封装管理 .....   | 98  |
| 5.7 WINSV LIB 与 OWL 结合的<br>开发技术 .....  | 102 |
| 5.7.1 多窗口程序设计 .....                    | 102 |
| 5.7.2 交互式分形图形程序设计 .....                | 105 |
| 5.8 Windows 环境二维数学与统计<br>图形程序设计 .....  | 108 |
| 5.8.1 多窗口分时程序设计 .....                  | 108 |
| 5.8.2 曲线与散点图组 .....                    | 109 |
| 5.8.3 条形图组 .....                       | 111 |
| 5.8.4 饼形图和等高线 .....                    | 112 |
| 5.9 Windows 环境三维图形程序设计 .....           | 113 |
| 5.9.1 三维线框图 .....                      | 113 |
| 5.9.2 数学曲面 .....                       | 114 |
| 5.10 小结 .....                          | 116 |

## 第六章 科学可视化程序自动生成

|                         |     |
|-------------------------|-----|
| 6.1 可视编程与仿真运行集成环境 ..... | 117 |
| 6.1.1 主控界面设计 .....      | 117 |
| 6.1.2 功能演示及对话设计 .....   | 120 |
| 6.2 函数指令编程 .....        | 126 |
| 6.2.1 指令集说明 .....       | 126 |
| 6.2.2 创建函数与描述文件 .....   | 128 |
| 6.2.3 指令参数输入对话 .....    | 129 |
| 6.3 函数指令虚拟运行 .....      | 131 |
| 6.3.1 描述文件结构与实例 .....   | 131 |

|                                   |     |                                   |     |
|-----------------------------------|-----|-----------------------------------|-----|
| 6.3.2 虚拟运行类的数据结构 .....            | 133 | 8.3 主控界面设计 .....                  | 165 |
| 6.4 程序游标设计 .....                  | 135 | 8.3.1 主程序设计 .....                 | 165 |
| 6.4.1 模式 .....                    | 135 | 8.3.2 版图视见操作的图标设计 .....           | 166 |
| 6.4.2 数据结构及应用 .....               | 135 | 8.3.3 版图分层显示选择框 .....             | 167 |
| 6.5 小结 .....                      | 138 | 8.4 缺陷模拟中的对话设计 .....              | 168 |
| <b>第七章 Intel NSP 三维真实造型技术</b>     |     | 8.4.1 工艺参数输入对话 .....              | 168 |
| 7.1 引言 .....                      | 139 | 8.4.2 圆片间的缺陷模拟 .....              | 169 |
| 7.2 三维真实造型(3DR)概览 .....           | 139 | 8.4.3 圆片上的缺陷模拟 .....              | 170 |
| 7.3 三维真实建模(3DR/GP) .....          | 142 | 8.5 小结 .....                      | 170 |
| 7.3.1 造型几何与视觉几何 .....             | 142 | <b>附录 A DOS 环境图形用户界面库功能接口</b>     |     |
| 7.3.2 坐标系与坐标变换 .....              | 143 | A.1 界面基础模块 .....                  | 172 |
| 7.3.3 光照模型 .....                  | 143 | A.2 鼠标框块 .....                    | 174 |
| 7.4 3DR/GP 的数据格式 .....            | 144 | A.3 扩展内存管理框块 .....                | 177 |
| 7.4.1 顶点的数据表示 .....               | 144 | A.4 窗口管理模块 .....                  | 179 |
| 7.4.2 点、矢量和矩阵的数据表示 .....          | 145 | A.4.1 窗口管理之快速图形堆栈模块 .....         | 179 |
| 7.4.3 图元的数据表示 .....               | 145 | A.4.2 窗口管理之窗口模块 .....             | 180 |
| 7.5 3DR/GP 函数组 .....              | 145 | A.5 汉字显示模块 .....                  | 182 |
| 7.5.1 GC 管理函数 .....               | 146 | A.6 图像存取模块 .....                  | 186 |
| 7.5.2 有关光线函数 .....                | 146 | A.6.1 图像存取之 XMS-VGA 模块 .....      | 186 |
| 7.5.3 有关材质函数 .....                | 147 | A.6.2 图像存取之 BMP 模块 .....          | 189 |
| 7.5.4 有关相机函数 .....                | 148 | A.7 对话模块 .....                    | 190 |
| 7.5.5 矩阵变换函数 .....                | 148 | A.8 控制模块 .....                    | 192 |
| 7.5.6 数学工具函数 .....                | 149 | A.9 主控界面模块 .....                  | 204 |
| 7.5.7 图元着色函数 .....                | 149 | A.9.1 VIMP GUI 模块 .....           | 204 |
| 7.5.8 选择和操作函数 .....               | 149 | A.9.2 主控界面之图标控制模块 .....           | 208 |
| 7.6 三维真实显示(3DR/RE) .....          | 149 | A.10 对话集成构件模块 .....               | 210 |
| 7.6.1 基本概念与术语 .....               | 150 | A.11 交互式绘图模块 .....                | 214 |
| 7.6.2 着色引擎 .....                  | 153 | A.12 键码宏定义 .....                  | 218 |
| 7.7 3DR/RE 数据格式 .....             | 156 | <b>附录 B DOS 环境科学可视化算法功能接口</b>     |     |
| 7.7.1 几何图元 .....                  | 156 | B.1 算法基础模块 .....                  | 220 |
| 7.7.2 空间坐标 .....                  | 156 | B.2 二维算法模块 .....                  | 228 |
| 7.7.3 颜色坐标 .....                  | 158 | B.3 三维算法模块 .....                  | 236 |
| 7.7.4 纹理坐标 .....                  | 158 | <b>附录 C Windows 环境科学可视化算法功能接口</b> |     |
| 7.7.5 位图 .....                    | 159 | C.1 算法基础模块 .....                  | 241 |
| 7.8 3DR/RE 变量组与函数组 .....          | 159 | C.2 二维算法模块 .....                  | 249 |
| 7.9 小结 .....                      | 159 | C.3 三维算法模块 .....                  | 258 |
| <b>第八章 集成电路功能成品率仿真软件的可视化环境设计</b>  |     | <b>参考文献</b> .....                 | 263 |
| 8.1 集成电路成品率的研究及其意义 .....          | 161 |                                   |     |
| 8.2 集成电路功能成品率仿真软件的系统结构与关键技术 ..... | 162 |                                   |     |

## 第一章 科学可视化导论

### 1.1 科学可视化的诞生、意义与推动力

计算技术的飞速发展,特别是高速计算机的广泛使用,使科学家与工程师们正面临着滚滚而来的数据“洪流”。每天从地面、海面、空间,由种类繁多的信息源产生的大量数据,远远超出了人脑分析解释这些数据的能力。由于缺乏大体积数据的有效分析手段,大约有95%的计算被浪费,这严重阻碍了科学研究的进展。难怪科学家们惊呼:“我们能够做的仅仅是收存数据!”

面临挑战,1987年在华盛顿召开的一次科学计算专题会议上,美国计算机成像专业委员会(CIP)应美国国家科学基金会的要求提出了解决方法——可视化(Visualization)。将图形学、成像技术与计算科学结合诞生了一个全新的领域——科学计算可视化(Visualization in Scientific Computing, Visc),又称科学可视化(Scientific Visualization, SV)。这次会议形成了正式的 Visc 报告,从而为 Visc 作为一门交叉学科的形成与发展奠定了基础。

可视化是从多个与计算机相关的学科中孕育出来的,不同学科对其描述不尽相同,其基本思想是“用图形与图像来表征数据”。可视化是作为解释大体积数据最有效的手段而率先被科学与工程计算领域采用,并发展为极为热门的研究领域——科学可视化。

科学计算的规模与深度在很大的程度上影响着一个国家科学技术发展的快慢和水平的高低。科学计算产生大量的数据,从这些数据中得出整体物理概念是非常复杂的问题。不仅如此,科学工作者期望在计算过程中能及时了解到出现什么情况和问题,从而能实时地对计算过程加以调节和控制。因此科学可视化有两层明确的含义:不仅要求计算结果可视化,而且要使得计算过程可视化。形象地说:“科学可视化使计算变得趣味盎然,它把浩如烟海的数据描绘成‘数据风景’,让人们漫游其中”。

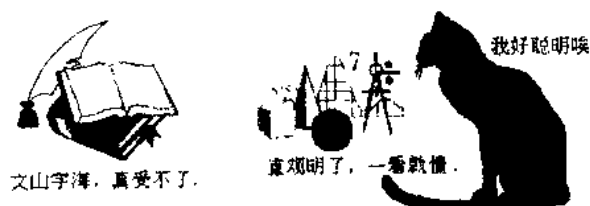


图 1.1 理解数据的不同方式

科学可视化的诞生赋予了可视化鲜明的内涵与科学的研究方法学,并更具一般性(因为在本质上,任何问题的可视化最终都归结为计算可视化)。这里引述美国 CIP 在 Visc 报告中对可视化的定义:“可视化是一种计算方法,它将符号(数据)转换成几何,使研究者能观察他们的模拟与计算。它将不可见的变成可见的,丰富了科学发现的过程,给予人们深刻与意想不到的洞察力,在很多领域使科学家的研究方式发生了根本变化”。该报告宣称:

“可视化对科学生产与重大科学突破产生巨大的影响，这可与超级计算机的影响相比拟。”

尽管科学可视化的研究主要集中在科学与工程计算领域，而其应用则远远超过计算科学自身。这是因为可视化技术适用于任何有大量数据存在的场合，而本质上并不局限于某种特定的应用。科学可视化一个极具魅力的优点在于“应用学科无关性”，尽管自身是交叉学科孕育而成的。这使得各行各业竞相引入科学可视化的研究成果，并掀起阵阵热潮。



图 1.2 计算机模拟撞车事件

可视化技术的应用大至高速飞行模拟，小至冰淇淋的熔化演示，已无处不在。例如，电影《侏罗纪公园》中恐龙活生生的展现，引起了观众的狂热，这是可视化技术在影视业的杰作。又如在 1994 年 4 月份问世的新型波音 777 喷气式客机，是西雅图波音公司采用可视化技术设计的第一架“无纸飞机”。福特汽车公司也采用可视化技术来设计新型轿车，并模拟撞车事件。

现代科学技术的飞速发展，引发了一场规模空前而持久的信息大爆炸。信息捕获、信息处理方式与速度已成为衡量社会进步的重要尺度，信息处理的一个重大变革是采用了可视化技术。事实上，可视化技术已在科学与工程、商业、娱乐等众多领域取得了卓著成效。《可视化：第二次计算机革命》的作者 Richard Mark Friedhoff 声称：“可视化技术是当今计算机工业的经济驱动力”。在图形软硬件领域独领风骚的 SGI 公司则许诺：“SGI 的可视化技术将给公众带来一幅饶有趣味的娱乐路线图，以便在信息高速公路上奔驰”。

目前业界对可视化、科学可视化、科学计算可视化这些术语未加严格区分。一般地认为“可视化”是一种通俗的称呼，后两种偏向学术，而“科学计算可视化”则更具专业性。本书以“可视化”作为泛称，而用“科学可视化”代表科学计算与工程设计领域的可视化。

## 1.2 科学可视化的神经机理

近年来，人们已逐渐认识到可视化是解释大体积数据的有效手段，并产生了广泛的研究兴趣。然而可视化的思想并不是新的，人们很早就知道图形、图像有助于理解对象的属性以及对象间的关系。在表达和交流技术思想方面，把图运用于科研和生产已经有很长的历史。地图就是自古至今用可视化技术解释大体积数据的典范(图 1.3)。再如经济统计、生产管理等部门，每天绘制大量的统计图形，用来帮助市场分析和预测。可视化还有助于跨越专业鸿沟，比如，倘若有人告诉我们某地区内各污染观察点的统计数字，这对非环保人员来说往往不得要领。但若根据这些数字，用计算机绘制污染等级分布图，则该地区的

污染情况就一目了然。



图 1.3 可视化技术应用典范——地图

可视化之所以能成为解释大体积数据的有效手段，有着内在的神经机理。医学、心理学研究表明，人类日常生活中接受的信息 80% 来自视觉，也就是说视觉信息是人类最主要的信息来源。而图形图像则是人类最容易接受的视觉信息，可视化技术充分利用了人类的视觉潜能。俗话说“一图抵千言”，而往往是千言万语也表达不了一张图包含的信息。利用图形、图像表示信息，可以迅速给人一个概貌，反映错综复杂的关系，最直观且信息最丰富。

人的五种感官都可看作是通向大脑的数据通道，每种感官都有一定的数据传输能力或称带宽。视觉较之其它感官的带宽要高得多，其带宽可估计如下：

| 视觉系统              | 图形类比  | 视觉性能    |
|-------------------|-------|---------|
| 视神经纤维             | 像素数/帧 | 2 兆像素/帧 |
| 可视数据维             | 比特/像素 | 3 比特/像素 |
| 独立视见区             | 帧/秒   | 7 帧/秒   |
| 视觉系统带宽约为 42 兆比特/秒 |       |         |

经验分析表明大多数人阅读速度在 600~1 200 词/分之间(或 400~800 比特/秒)。很明显文字并不是充分利用人脑视觉系统的最佳手段。神经研究发现人脑左右半球有相对明确的功能分工。左脑具有逻辑分析能力，负责语言交流与数学推理，而右半球则是使人具备视觉能力的一侧，识别与构造几何图像的功能，就驻留在右脑(例如，识别人的面孔就是一种非常复杂的功能)。左脑利用文字符号“串行”处理数据的输入与输出，右脑则以“并行”方式处理诸如空间几何关系这类可视信息。

可视化作为一种工具，通过将符号信息(数据)转换成视觉信息(图像)，从而有效地匹配右脑的能力。正如美国 CIP 在 Visc 报告所指的那样：“据估计 50% 脑神经细胞与视觉相联，可视化的目的是使这些神经机器投入运转”。

## 1.3 可视化技术的一些应用

### 1.3.1 SGI 的贡献

谈到可视化技术的应用,就不能不提到 SGI 公司。领导三维图形的潮流是 SGI 的优势所在。在某种程度上说,SGI 的图形软硬件技术象征着当今最先进的可视化技术,SGI 图形产品的每一份新市场都意味着可视化技术的一种新应用。

自苹果公司推出 Machintosh 计算机而在市场上享有威名以来,还没有一家计算机公司能像 SGI 那样垄断图形市场,并且前景辉煌。从电影艺术到分子科学,从交互式电视到电子游戏,SGI 创造的数字影像正把计算机世界引向一个新的思维空间,确切地说是三维图形空间。

在 SGI 以前,大多数工作站只能建立粗糙的三维线框图。工作站的微处理器不能画出复杂的图形。毫无疑问,为了用硬件处理三维实体图形,必须并行地确定成千上万个从不同角度得到的图形形状。任何一个可视实体均由几十万个图形元素组成,欲使图形连续流畅地移动,它们必须每秒变化 30 次。SGI 的创始人 Jamse H. Clark 两个关键的创意,使图形技术突飞猛进地发展。1979 年到 1982 年在斯坦福大学,他和 6 个学生一起研制成功几何图形发生器和图形库。几何图形发生器是一组可加速模型建立的定制芯片,而图形库则是由上百个规则构成的软件包,它能帮助开发人员把模型转变成三维图形,几何图形发生器减轻了中央处理器对大量图形处理的负担,从而大大加快了建模的速度。

SGI 的图形工作站可将大量数据转化为人们很容易理解的图形与图像,诸如机翼、人口增长趋势和银河系的星际图案。这些图像可以反复模放,能观模错综复杂的图形细节和纹理。药理研究员用 SGI 工作站观察细胞表面的物理现象,从而更好地了解疾病和免疫反应。大多数航空公司用 SGI 计算机设计飞机的气流模型,而不必像从前那样化大量时间和金钱作风洞实验。

显然,三维图形技术倍受科学家和技术人员的青睐。人们为美妙的图形开辟了成千上万种的应用。

### 1.3.2 影模业

可视化技术在影视业的应用成绩斐然,电影《侏罗纪公园》中的恐龙动人逼真的展现使观众达到了狂热的程度。过去要费心制作的实物模型与特技镜头,现在只需在 SGI 计算机上运行相应的软件就能实现。甚至电影演员都可以用虚拟现实技术来实现。在电影《阿甘正传》中出现了约模·肯尼迪和阿甘见面的场合。电影《The Crow》的主角 Brandon Lee 在拍摄结束前被杀,后来就用 SGI 机器把 Lee 的形象数字化后输入到没拍完的胶卷上,这部电影最终得以完成。

可视化技术将给电视带来新面貌。Time Warner 公司有线电视部计划在电视能达到的地方都采用 SGI 计算机。在 Orlando 的试验中,有 4 000 多个地方都采用改装的 SGI 工作站做交互式电视。人们可以随心所欲地选择电视、电影节目,不出门就可以购物。

### 1.3.3 金融可视化

金融可视化与科学可视化一样，它可以将大量字母数字数据转换成图形或可见物体，从而使数据更易理解和分析。股票市场就是采用这种技术的主要领域。



图 1.4 金融可视化提供了直观的选择

70 年代起，股票市场已从纸上的交易过渡到电子化的交易。从此以后，金融市场每天的交易额和复杂程度大大地增加了。很多公共团体投资商现已可以同时进行各种各样的有价证券交易，或收进各种债券。

Metaphor Mixer(比拟综合器)是使用虚拟现实技术的第一个金融可视化产品，是由纽约的 Maxus Systems International 公司研制的。路透社的特别小组通过卫星将 Metaphor Mixers 当时收集的金融信息实时地发给在虚拟环境中的经纪人。该虚拟环境类似于一个足球场，场上的格子线将各个公司按类别分成若干实业组，然后又按照市场进一步地分组。经纪人要戴一个头盔式显示器，这样他就可以进入一个仿真的金融中心，用一个输入设备(如手持式操纵器或力矩球)，就可以在该仿真环境中活动，并获取各个公司的信息。

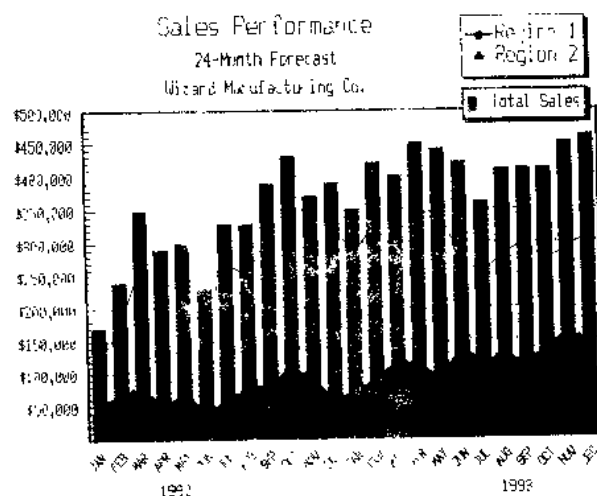


图 1.5 商业统计中的数据可视化

每个公司都用一种带颜色的“棋子”表示，每一种颜色就是一种金融活动的符号。例如红色棋子表示从昨天开始价格下跌，蓝色棋子表示价格上升，灰色棋子表示价格稳定。而一种股票的棋子数就表示该股票的相对成交额，转动和使棋子闪烁则表示为供经纪人参考的每个公司的选择标准。各个市场(如交易所)之间的价格波动就意味着是赔钱还是赚钱，



因而金融可视化的各种应用对于获取瞬息可变的即时数据是一种必不可少的工具。

### 1.3.4 石油工业

可视化技术已经成功地应用到石油工业中,例如在油田开发、油藏数值模拟、石油地质、测井、遥感遥测等方面。在SGI工作站上可以模拟油田开发过程的油、气、水在地下的变化情况,并以二维、三维图形直观地显示出来,为油田开发专家们提供优选和决策的科学依据,使之更加合理地调整新老油田的开发方案。

在油田地质勘探方面,编写能反映地质构造、盆地模拟、地震地质解释、黑油模型、挥发油模型、地质学统计等可视化应用程序,用以逼真地反映地层结构,建立地层界面,处理地层界面关系,可对复杂地层进行任意切割,局部放大,进一步探明地层的内在因素。

推广可视化技术,运用现代科学方法来勘探、开发和管理油田,是石油工业向现代化迈进的大趋势。

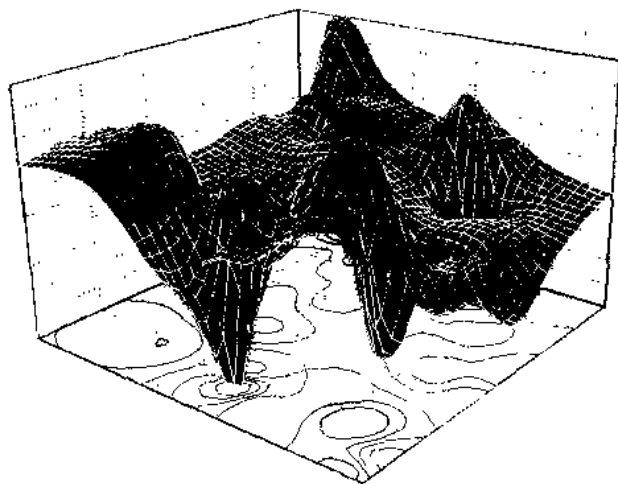


图 1.6 油层的三维图形显示

### 1.3.5 飞机设计

近年来,计算机公司一直在努力实现“无纸办公室”,以在线系统替代那些手册、活页装订及其它基于纸张信息的硬拷贝办公环境。西雅图波音公司用了几年时间努力地创造着“无纸飞机”。随着新型波音777喷气式客机在1994年4月份的问世,波音公司实现了他们的梦想,这在很大程度上要归功于SGI的可视化技术。以前SGI系统被更多地拿来进行深奥问题的研究和流体力学这样的开发应用。目前,它们正进入到更负有关键使命的应用主流中。波音公司采用了近千个SGI系统,使得波音CAD系统能够对飞机的部件进行逼真的表现,并建立了飞机的立体模型。这个立体模型使工程师和其他工作人员在飞机研制过程中,能够“穿行”在数字式飞机上。这种穿行作用还带来另一个好处,使得非技术人员可以加入到飞机设计过程中。

可视化技术设计飞机的方法为波音公司节省了大量的时间和金钱,这种新方法使工程