

信息管理与信息系统专业规划教材

面向对象开发方法

石双元 主编



科学出版社
www.sciencep.com

TP312/2530

2007

信息管理与信息系统专业规划教材

面向对象开发方法

石双元 主编

科学出版社

北京

内 容 简 介

本教材围绕面向对象程序开发的两大前沿技术 C#和 ASP.NET 展开内容,在其内容和结构安排上力求做到系统性和连贯性。本书分为三部分。第一部分包括第 1~4 章,是基础知识部分,包括面向对象的基本概念与特征、C#的语法及面向对象特性、Web 开发的背景知识;第二部分包括第 5~10 章,讲解 ASP.NET 开发的主要内容,按照由浅入深的顺序,分别阐述了 ASP.NET 页面程序结构、Web 服务器控件及 ASP.NET 中的数据绑定技术与实现方法;第三部分包括第 11~14 章,是高级应用部分,内容为控件与代码的分离、XML 的应用、DataWindow.NET 控件和创建 ASP.NET 应用。

本书可作为高等院校信息管理与信息系统、电子商务、工商管理及管理科学等专业的教材,也可供相关专业技术人员、工程技术人员和其他人员参考。

图书在版编目(CIP)数据

面向对象开发方法/石双元主编. —北京:科学出版社, 2007

(信息管理与信息系统专业规划教材)

ISBN 978-7-03-019251-6

I.面… II.石… III.面向对象语言—程序设计—高等学校—教材

IV.TP312

中国版本图书馆 CIP 数据核字(2007)第 093506 号

责任编辑:陈晓萍/责任校对:耿耘

责任印制:吕春珉/封面设计:耕者设计工作室

科学出版社 出版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

双青印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2007年7月第一版 开本: B5(720×1000)

2007年7月第一次印刷 印张: 26 1/4

印数: 1—3 000 字数: 514 000

定价: 34.00 元

(如有印装质量问题,我社负责调换〈环伟〉)

销售部电话 010-62136131 编辑部电话 010-62138978-8003

编 委 会

顾 问

马费成 薛华成 侯炳辉

主 任

张金隆

编 委 (以姓氏笔画为序)

王 虎	王学东	刘腾红	孙细明	孙茜雯
何 浩	张唯佳	李 纲	杨新年	肖 华
郑双怡	夏火松	徐绪松	聂规划	曾庆伟
鲁耀斌	雷春明	蔡淑琴	潘开灵	

序

国家教育部于 1998 年 7 月 6 日公布了新的《普通高等学校本科专业目录》，将原来的经济信息管理、信息学、科技信息管理、林业信息管理和**管理信息系统**等专业合并为**管理学科门类中的信息管理与信息系统专业**。可以认为，这次合并既是学科相融的必然，也是国家信息化发展的需要。据有关资料介绍，到目前为止，全国已有超过 200 所高校开设了**信息管理与信息系统专业**。

自 20 世纪 40 年代以来，信息技术经过 60 余年的高速发展，它对人类社会各个领域的影响越来越广泛和深入，其影响最大、受益最多的当属**管理和经济领域**。信息作为最主要的经济资源，已经被人们所接受，并且愈来愈受到重视。信息技术的普及和推广，信息资源的组织、开发和利用，促进了企业的发展和产业结构的调整。当前所实施的**电子商务、电子政务和数字图书馆**等工程直接加速了生产力的发展和促进了社会的进步。我国政府提出的“以信息化带动工业化”的战略举措，必将有力提升我国的综合国力，同时也为**信息管理与信息系统专业**带来极大的发展机遇和发展空间。

信息管理与信息系统是一门交叉学科，它不是信息技术和管理科学的简单组合，而需要融合**管理学、经济学、系统科学、运筹学和计算机科学**于一体，因而，必须要有一套具有本专业特点的知识结构体系和适合本专业需要的教材体系。

信息管理与信息系统专业从 1998 年设立至今的 5 年来，许多专家学者在专业建设和教材建设方面倾注了大量的心血，有力地促进了专业和学科的发展。但是，由于该专业具有**跨度大、内容新和变化快**等特点，如何培养适应现代信息技术高速发展需要的、具有创新能力的、既懂信息技术又懂管理的复合型人才，对广大教育工作者而言是一个巨大的挑战。

在科学出版社的直接推动下，在我国信息管理领域的知名学者**薛华成教授、侯炳辉教授和**马费成教授****的指导下，在**湖北省信息产业厅和经济贸易委员会**及相关企业的支持下，**武汉地区包括华中科技大学、武汉大学、华中师范大学、中南财经政法大学和**武汉理工大学****等 20 余所高校联合编写了这套针对本科生的**信息管理与信息系统专业规划教材**。

这套教材共 22 本，除了数学基础类的《运筹学》外，大致可以归为以下 3 类：

计算机技术类（8 本）：《数据库技术》、《计算机网络技术》、《数据结构——Java 实现》、《面向对象的开发方法》、《数据仓库与数据挖掘技术》、《操作系统》、《多媒体信息管理技术基础》和《实用软件工具》。

信息系统类（6 本）：《信息系统分析与设计》、《信息系统案例分析》、

《项目管理》、《管理信息系统》、《信息系统原理》和《决策支持系统》。

信息管理类（7本）：《信息管理学基础》、《信息资源管理》、《信息经济学》、《信息政策与法规》、《信息组织学》、《信息检索》和《信息安全》。

这套教材具有以下特点：

（1）内容新。正如前面所指出的一样，这套教材并不是简单地分门别类讲解信息技术和管理科学知识，而是站在信息管理与信息系统专业这个全新的角度上，力求全面及时反映信息管理与信息系统领域国内外的最新发展和研究成果。

（2）体系全。为保证本系列教材体系的完整性和内容系统性，编委会曾多次开会讨论并广泛征求国内信息管理与信息系统领域的有关专家的意见，该套教材主要集中于专业基础课和专业课方面，并考虑了这些课程之间的相互衔接和整体上的协调。

（3）注重基础。本系列教材从选题到编写充分考虑到当前我国本科生的知识结构和知识背景及其后续发展的需要，着重于讲解信息管理与信息系统专业的基础知识，注意培养学生的能力。

（4）结合实际，多采用案例教学。本系列教材的作者都是从事一线教学工作的教师，了解本科生的特点和需求，大多数作者又有从事信息系统开发和信息资源管理的经验，了解实际工作对本专业的需求。因此，在编写过程中作者们能注意理论与实践相结合，通过引入适当的案例和实验，加深学生对理论知识的理解和掌握。

我们希望，这套教材的成功出版，能为推动我国信息管理与信息系统专业教育工作的的发展、促进信息化人才的培养起到积极的作用。

这套教材是我们不同类型的学校，不同专业背景、但同属信息管理与信息系统专业教师合作的一种尝试。我们欢迎信息管理和信息系统及相关专业的教师、学生和科研工作者以及有关人士提出意见和建议，以便进一步提高我们的教材质量。

本套规划教材编委会主任
华中科技大学管理学院院长
管理信息研究所所长
张金隆 教授

前 言

未来程序设计朝两个方向发展，一是完全采用面向对象和基于组件的技术，二是程序设计开发语言与平台的无关性和面向 Web 应用。其原因是，目前的应用程序正在从基于独立的操作系统转向基于 Internet 平台，程序员最关心的是如何实现业务逻辑，而不是各种平台之间的通信和管理，微软公司的 .NET 战略和所包含的 C# 正好迎合了这一时代的要求。

本教材围绕面向对象程序开发的两大前沿技术 C# 和 ASP.NET 展开内容。C# 作为完全面向对象的程序开发语言，既支持 Window Form，也支持基于 Web 的应用开发，而 ASP.NET 是一种程序逻辑和 Web 相结合的技术。.NET 支持多种程序设计语言，我们选择 C# 作为背景语言，不仅因为 C# 语言是为了在 .NET 中使用和支持基于组件的编程而特别设计的，更因为其完全面向对象和简单易学。

面向对象技术被誉为软件技术的一场革命，与传统的结构化软件开发技术相比，面向对象技术可以大大提高软件的开发效率和质量，改善软件的可靠性和重用性。面向对象的概念已经渗透到计算机的许多领域，如面向对象分析与设计、面向对象开发方法、面向对象数据库和一些面向对象的开发工具等。

面向对象涉及的范围较广泛，本书侧重于面向对象的程序开发方法。在其内容和结构安排上力求做到系统性和连贯性。本书首先讲述了面向对象技术的基本概念和特性；接着简要介绍了 C# 语言的基础知识以及面向对象特性在 C# 语言中的表现形式和实现方法，为了避免重复，我们将 C# 的类的设计和高级编程技术等内容融合到后面的代码分离技术和自定义控件的设计中；然后，对 ASP.NET 开发涉及的背景知识，如 HTML、XML、ASP 进行了简单介绍，使得没有这些背景知识的读者也能很快熟悉和掌握这些知识（如果课时紧张，背景知识部分可作为选学内容或自学内容）。

本书不仅完整介绍了 ASP.NET 及其服务器控件和基于 ADO.NET 的数据库开发技术，还融入了最新的控件和 Web 开发技术，如 ASP.NET 的扩展控件和 DataWindow.NET 控件。程序代码与页面的分离是 Web 程序员多年的梦想，也是未来程序设计的方向，本书专门介绍了代码分离的四种方法，并分析了各自的优缺点。无处不在的 XML 已经成为基于 Web 的数据语言，本书用一章的篇幅详细介绍了 ASP.NET 页面中 XML 的应用技术。本书的最后部分介绍了如何配置和发布一个基于 ASP.NET 的 Web 应用。为便于初学者掌握，本书每章都有大量的实例程序。

全书分为 14 章。其中第 1 章主要介绍面向对象开发的基本概念与面向对象

的特征;第 2、3 章讨论了 C#语言的基础知识和面向对象方法在 C#语言中的实现;第 4、5 章讲述了 ASP.NET 开发所需的背景知识和基本方法,包括 HTML、XML 和 ASP 等;第 6~14 章以 C#为支持语言、用大量实例系统地介绍了 ASP.NET 的开发方法与技术,其中,第 6~8 章介绍 ASP.NET 的基础控件、验证控件与扩展控件;第 9、10 章介绍基于 ADO.NET 的数据连接、数据绑定技术及各种列表控件与数据绑定的实现方法;第 11 章讨论了 ASP.NET 中的代码与页面分离技术;第 12 章介绍了如何在 ASP.NET 中应用 XML 技术;第 13 章介绍了最新的 DataWindow.NET 控件及其应用方法;第 14 章介绍了应用程序的配置与分发。

通过本书的学习,读者不仅可以系统地掌握面向对象开发的原理与方法,系统地学习目前最新的 C#和 ASP.NET 开发技术与方法,而且可以了解和掌握 Web 开发的最新技术与方法,如页面与代码分离、XML 应用和 DataWindow.NET 等。本书可作为高等院校信息管理与信息系统、电子商务、工商管理及管理学等专业的教材和参考书,还可供相关领域的科学工作者以及系统开发人员阅读参考。

本书是集体创作的成果。主要参加编写的人员有:石双元(第 1 章、第 4~14 章)和甘早斌、程海芳(第 2、3 章部分)。石双元负责全书的策划、统稿以及全书格式和文字校对工作。另外,刘伟、徐丽娜、吴新明、陈琦、史俊、杨琴、张浩等研究生在搜集、整理资料、文字校对方面做了大量的工作,在此表示感谢。

在本书的编写过程中,我们参考和引用了不少国内外文献资料,在此对这些资料的作者表示深深的感谢。本书的编写还得到了华中科技大学管理学院张金隆教授、蔡淑琴教授的大量帮助和指点,以及科学出版社有关同志的大力支持和帮助,在此对他们表示诚挚的谢意。

由于写作时间较短,以及编者水平有限,书中难免存在缺点和错误,敬请各位专家和读者批评指正。

石双元

目 录

第 1 章 面向对象开发技术及其特征	1
1.1 软件危机与软件工程.....	1
1.1.1 软件及其特性.....	1
1.1.2 软件危机及其产生的原因.....	2
1.2 面向对象技术的形成与发展.....	3
1.2.1 面向对象方法与结构化方法的比较.....	3
1.2.2 面向对象问题求解概述.....	5
1.3 面向对象的基本概念.....	7
1.3.1 对象.....	7
1.3.2 对象的属性.....	7
1.3.3 消息和方法.....	8
1.3.4 消息传递.....	8
1.3.5 类与实例.....	8
1.3.6 类的层次结构.....	9
1.4 面向对象开发方法的特征.....	11
1.5 面向对象开发与面向过程开发的主要区别.....	16
第 2 章 C#语言基础	18
2.1 C#语言及其特点.....	18
2.2 中间语言代码与.NET 的跨平台.....	20
2.3 一个简单的 C#程序.....	20
2.4 命名空间.....	25
2.4.1 什么是命名空间.....	25
2.4.2 命名空间的引用.....	26
2.5 C#程序的构成.....	27
2.6 C#的类和对象定义.....	29
2.7 C#的格式与数据类型.....	31
2.7.1 标识符.....	31
2.7.2 C#的书写规则.....	33
2.7.3 数据类型.....	34
2.7.4 数据类型转换.....	36
2.7.5 常量.....	37

2.7.6	变量	38
2.7.7	枚举与数组	40
2.8	运算符与表达式	43
2.8.1	C#运算符	43
2.8.2	表达式	47
2.9	方法的定义与使用	47
2.10	定义方法	50
2.11	流程控制语句	51
2.11.1	结构化程序设计的三种基本流程	51
2.11.2	分支结构	51
2.11.3	循环结构	55
2.11.4	跳转语句	57
2.11.5	递归调用	59
第3章	C#的面向对象特性	61
3.1	类的定义与 C#修饰符	61
3.1.1	类的定义	61
3.1.2	C#修饰符	63
3.2	创建对象与构造方法	64
3.2.1	创建对象	64
3.2.2	构造方法	65
3.2.3	构造方法应用举例	68
3.2.4	析构方法	69
3.2.5	类的修饰符	70
3.2.6	类的嵌套	72
3.3	类的成员及其修饰	74
3.3.1	变量及其修饰	74
3.3.2	方法及其修饰符	81
3.4	访问控制符	87
3.4.1	公共访问控制符 <code>public</code>	87
3.4.2	缺省访问控制符	88
3.4.3	保护访问控制符 <code>protected</code>	88
3.4.4	私有访问控制符 <code>private</code>	89
3.5	C#的继承及其实现	90
3.5.1	继承的定义	90
3.5.2	继承的实现	92
3.6	C#语言中的多态与实现	94

3.6.1	什么是多态	94
3.6.2	构造方法的继承与重载	96
3.6.3	C#多态的实现	98
3.7	接口及其实现	101
3.7.1	单重继承与多重继承	101
3.7.2	C#接口概述	101
3.7.3	接口的定义	102
3.7.4	实现接口	103
3.8	异常处理	105
3.8.1	异常产生的原因	105
3.8.2	捕获异常与抛弃异常	105
第 4 章	ASP.NET 背景知识	107
4.1	C/S 与 B/S 结构	107
4.1.1	什么是 C/S 与 B/S 结构	107
4.1.2	两种结构的比较	108
4.2	HTML 语言概要	109
4.2.1	HTML 文档的基本结构	110
4.2.2	HTML 文档结构标签	112
4.2.3	版面风格控制	114
4.2.4	在 HTML 中显示特殊字符	119
4.2.5	文件的链接	120
4.2.6	HTML 表单	123
4.2.7	HTML 样式表	128
4.3	XML 结构及语法	134
4.3.1	XML 及其特征	134
4.3.2	XML 文档结构	136
4.3.3	XML 文档格式定义	139
4.3.4	XML 文档的显示与转换	143
4.4	ASP 动态页面技术	149
4.4.1	静态网页与动态网页	149
4.4.2	ASP 及其特征	151
4.4.3	ASP 文件结构	152
4.4.4	ASP 页面的运行环境	152
4.4.5	VBScript 语法简介	154
4.4.6	ASP 与 ADO 的连接	160
4.4.7	ASP 内置对象	166

第 5 章	ASP.NET 概览	171
5.1	ASP.NET 及其特征	171
5.1.1	从 ASP 到 ASP.NET	171
5.1.2	ASP.NET 及其主要特性	172
5.1.3	.NET 框架概述	173
5.2	ASP.NET 环境配置	175
5.2.1	ASP.NET 运行环境的配置	175
5.2.2	ASP.NET 开发环境的配置	179
5.3	ASP.NET 中的类与命名空间	182
5.3.1	ASP.NET 中的类	182
5.3.2	ASP.NET 的命名空间	183
5.4	ASP.NET 页面表单与控件	184
5.5	简单的 ASP.NET 页面	186
5.6	ASP.NET 页面的程序结构	187
5.7	控制页面导航	191
5.7.1	参数的传递与接收	191
5.7.2	页面之间的数据共享	192
第 6 章	Web 服务器基本控件	195
6.1	Web 基本控件介绍	195
6.1.1	Label 控件	196
6.1.2	Button 控件	198
6.1.3	TextBox 控件	200
6.1.4	DropDownList 控件	203
6.1.5	ListBox 控件	207
6.1.6	CheckBox 控件	210
6.1.7	CheckBoxList 控件	212
6.1.8	RadioButton 控件	215
6.1.9	RadioButtonList 控件	216
6.1.10	HyperLink 控件	219
6.1.11	列表类控件的属性与方法	221
6.2	ASP.NET 基本控件综合应用	224
第 7 章	ASP.NET 验证控件	227
7.1	表单信息的验证	227
7.2	RequireFieldValidator 控件	228
7.3	RegularExpressionValidator 控件	230
7.4	CompareValidator 控件	234

7.5	RangeValidator 控件	236
7.6	ValidationSummary 控件	239
7.7	CustomValidator 控件	241
第 8 章	ASP.NET 高级控件与扩展控件	244
8.1	ASP.NET 高级控件	244
8.1.1	Panel 控件	244
8.1.2	通过编程控制页面上的控件	251
8.1.3	Calendar 控件	254
8.1.4	HTMLInputFile 控件	254
8.2	ASP.NET 扩展控件	255
8.2.1	IEWebControls 的安装	255
8.2.2	树形视图 TreeView 控件	257
8.2.3	工具条 ToolBar 控件	264
8.2.4	标签卡 TabStrip 控件	266
第 9 章	ADO.NET 与数据绑定	271
9.1	什么是 ADO.NET	271
9.2	ADO.NET 的访问方法	272
9.3	ASP.NET 中的数据库操作	275
9.3.1	检索数据库中的记录	275
9.3.2	检索单条记录	277
9.3.3	在 ASP.NET 中实现动态查询	277
9.3.4	向数据库中添加记录	280
9.4	在查询中使用参数	281
9.5	使用 SQL 存储过程	283
9.6	ASP.NET 中的数据绑定	287
9.6.1	什么是数据绑定	287
9.6.2	绑定数据源到选项类控件	288
第 10 章	在列表控件中显示数据	295
10.1	用 Repeater 控件显示数据	295
10.1.1	绑定 Repeater 控件到数据源	295
10.1.2	在 Repeater 控件中使用模板	297
10.1.3	在 Repeater 控件内绑定其他控件	300
10.1.4	用 Repeater 控件实现主从表	302
10.2	用 DataList 控件显示数据	304
10.3	用 DataGrid 显示数据	309
10.3.1	绑定 DataGrid 控件到数据源	309

10.3.2	在 DataGrid 控件中创建列	311
10.3.3	在 DataGrid 中实现排序	320
10.3.4	DataSet 的使用技术	322
10.3.5	在 DataGrid 中实现分页	323
10.4	Repeater、DataList 与 DataGrid 比较	326
第 11 章	程序代码与页面分离技术	328
11.1	创建和使用 .NET 组件	328
11.1.1	业务组件及其优势	328
11.1.2	ASP.NET 组件的建立	329
11.1.3	ASP.NET 组件的使用	331
11.1.4	使用和访问组件的属性	333
11.2	代码隐藏技术	337
11.3	用户控件	341
11.3.1	用户控件的定义与使用	341
11.3.2	使用用户控件的属性与方法	344
11.4	自定义控件	346
11.4.1	自定义控件及其特性	346
11.4.2	为控件添加属性和方法	350
第 12 章	在 ASP.NET 中使用 XML	353
12.1	为什么要使用 XML	353
12.2	.NET 中的 XML 类	354
12.3	使用 DataSet 对象操作 XML	355
12.3.1	从 XML 读取数据到 DataSet	355
12.3.2	从 DataSet 输出数据到 XML 文档	357
12.3.3	DataSet 与 XmlDataDocument 一起使用	359
12.4	使用 XML 控件	361
12.4.1	直接用 XML 控件显示 XML 数据	362
12.4.2	在 XML 控件中加载 DOM	363
12.5	使用 XslTransform	366
12.6	XML 与数据库绑定综合应用	368
第 13 章	DataWindow.NET 控件	372
13.1	DataWindow 对 Web 的支持	372
13.2	安装 DataWindow.NET	375
13.3	DataWindow.NET 类及其特性	377
13.3.1	DataWindow.NET 的运行过程	377
13.3.2	什么是 WebDataWindow	378

13.4	在 ASP.NET 中使用 DataWindow.NET 控件	381
13.4.1	如何使用 DataWindow.NET 开发 ASP.NET 程序	381
13.4.2	一个简单的 DataWindow 控件应用	384
13.4.3	利用 DataWindow 控件操作数据	386
第 14 章	创建 ASP.NET 应用	391
14.1	什么是 ASP.NET 应用	391
14.1.1	如何配置 ASP.NET 应用程序	391
14.1.2	ASP.NET 应用程序框架	392
14.2	配置 Global.asax 文件	394
14.3	配置 Web.Config	396
14.4	在 Web.Config 文件中配置数据库连接参数	398
14.5	课程设计	399
14.5.1	在线购物系统	399
14.5.2	在线考试系统	401
主要参考文献	403

第 1 章 面向对象开发技术及其特征

1.1 软件危机与软件工程

1.1.1 软件及其特性

目前,被普遍接受的软件的定义是:软件(Software)是计算机系统中与硬件(Hardware)相互依存的另一部分,它包括程序(Program)、相关数据(Data)及其说明文档(Document)。其中程序是按照事先设计的功能和性能要求执行的指令序列;数据是程序能正常操作的信息的数据结构,它是程序操作和处理的对象;文档是与程序开发、维护和使用有关的各种图文资料。

同传统的工业产品相比,软件有其独特的特性。

1) 软件是一种逻辑实体,具有抽象性。这个特点使它与其他工程对象有着明显的差异。人们可以把它记录在纸上或存储介质上,但却无法看到软件本身的形态,必须通过观察、分析、思考、判断,才能了解它的功能。

2) 软件没有明显的制造过程。一旦开发成功,就可以几乎以零成本大量拷贝同一内容的副本。所以对软件的质量控制,必须着重在软件分析、设计与开发上。

3) 软件在使用过程中,没有磨损和老化的问题。软件在生存周期后期不会因为磨损而老化,但会为了适应硬件、应用环境以及需求的变化而进行修改。

4) 软件对硬件和应用环境有着不同程度的依赖性,这导致了软件移植的问题。

5) 软件的开发成本相当昂贵。软件开发需要投入大量的、高强度的脑力劳动,成本非常高,风险也大,现在软件的开销已大大超过了硬件的开销。

6) 软件工作牵涉到很多社会因素。许多软件的开发和运行涉及组织机构、体制和管理方式等问题,还会涉及人们的观念和心理。这些与人有关的因素,常常成为软件开发的困难所在,直接影响到项目的成败。

软件是一种抽象的逻辑思维产品,软件的研制过程实际上是软件人员的“思考”过程。而这种“思考”过程没有固定的模式与标准,即容易被修改。正是由于其“容易”被修改,因而没有一个软件在投入运行之后就不再进行修改扩充的。软件的修改过程既是其不断完善的过程,也是新产品的诞生过程,新软件产品与老产品之间没有明显的界限。也正是由于其可修改性,软件产品也没有明确的完成标志,而且软件产品在完成之前很难预料系统是否成功,很多研制过大型软件系统的工程师在回首往事时,经常把自己完成的一些工程称为“遗憾工程”。

1.1.2 软件危机及其产生的原因

对于人类而言，软件是一个全新的东西，其发展历史不过四五十年。人们对软件的认识经历了一个由浅到深的过程。

在计算机系统发展的初期（20世纪60年代中期以前），大多数软件是由使用该软件的个人或机构研制的，编写者和使用者往往是同一个人，软件规模小，除程序清单外，无其他文档资料，其开发也没有什么系统的方法可以遵循，软件设计是在某个人的头脑中完成的一个隐藏的过程。

从20世纪60年代中期到70年代中期，软件开始作为一种产品被广泛使用，出现了“软件作坊”。这一时期软件开发的方法基本上仍然沿用早期的个性化软件开发方式，但软件的数量急剧膨胀，软件需求日趋复杂，维护的难度与成本越来越大，导致软件维护工作很难进行，甚至不可维护，即不能及时修改运行时发现的错误，不能适应新的硬件环境。软件的“不可维护”导致了“软件危机”。

概括地说，软件危机是指在计算机软件的开发和维护过程中所遇到的一系列严重问题，这些问题主要表现在两方面：如何开发软件以满足对软件不断增长、日趋复杂的需求；如何维护数量不断膨胀的软件产品。

具体地说，软件危机主要有以下表现形式。

- 1) 对软件开发成本和进度的估计常常不准确。
- 2) 用户对“已完成”的软件系统不满意的现象经常发生。
- 3) 软件产品的稳定性不高，质量不可靠。
- 4) 软件的可维护性差。
- 5) 软件通常没有适当的文档资料。
- 6) 软件的成本不断提高。
- 7) 软件开发效率提高的速度远远跟不上计算机应用与普及的速度。

软件危机产生的原因，一方面与软件本身的特点有关；另一方面与软件开发和维护的方法不正确有关。软件开发和维护的不正确方法主要表现为：忽视软件开发前期的需求分析；开发过程没有统一的、规范的方法论的指导，文档资料不齐全，忽视人与人的交流；轻视软件的维护等等。可见，这些大多数都是软件开发过程管理上的原因。

软件危机使得人们开始对软件及其特性进行更深一步的研究。1968年，北大西洋公约组织的计算机科学家在前联邦德国召开国际学术会议，讨论软件危机问题，探讨软件开发与维护的新方法，正式提出软件工程（Software Engineering）这个概念。

在以后30多年的发展中，人们针对软件危机的表现和原因，经过不断地实践和总结，越来越认识到：按照工程化的原则和方法组织软件开发工作是摆脱软