

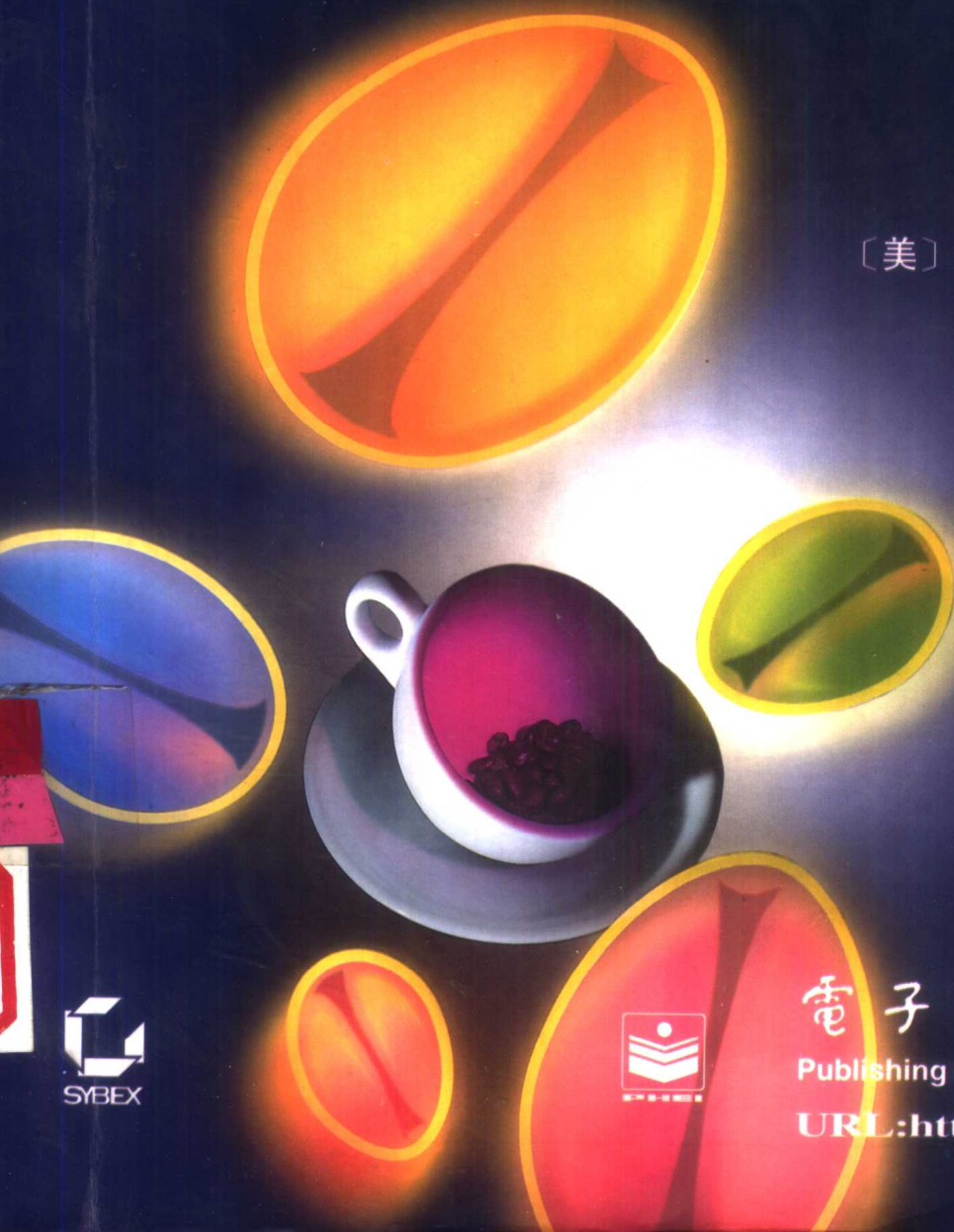
Mastering JavaBeans

JavaBeans

从入门到精通

〔美〕 Laurence Vanhelsuwe 著

邱仲潘 等译



SYBEX



电子工业出版社

Publishing House of Electronics Industry

URL: <http://www.phei.com.cn>

Mastering JavaBeans

JavaBeans从入门到精通

〔美〕 Laurence Vanhelsuwé 著

邱仲潘 等译

電子工業出版社

Publishing House of Electronics Industry

内 容 提 要

Java语言从1995年底诞生至今, 日益受到世人的重视。bean (即Java bean) 就是Java的类, 或者说, 就是Java语言的组件, 充分体现了Java语言平台独立和面向对象编程的优势。

JavaBeans是下一代网络程序员的资源。它是Java语言的组件: bean的接口和通讯部分的标准, 实现了bean的复用。本书致力于教会读者如何运用标准创建和开发Java语言, 由浅入深, 从最易理解的bean属性, 到最具挑战性创造性的AWT1.1事件模型, 定制的属性编辑器和对象序列化等多个方面逐步指导, 提供大量编程经验。本书面向高级程序开发, 介绍了在开发过程中常见的问题和解决技巧。对于令大多数编程人员深感头痛的线程问题, 本书也有详尽描述。

深入了解JavaBeans必将使你的软件开发高度省时、可靠而且适用性极强, 本书正是学习和掌握JavaBeans的权威作品。



Copyright©1997 SYBEX Inc., 1151 Marina Village Parkway, Alameda, CA 94501. World rights reserved. No part of this publication may be stored in a retrieval system, transmitted, or reproduced in any way, including but not limited to photocopy, photograph, magnetic or other record, without the prior agreement and written permission of the publisher.

本书英文版由美国SYBEX公司出版, SYBEX公司已将中文版独家版权授予中国电子工业出版社和北京美迪亚电子信息有限公司。未经许可, 不得以任何形式和手段复制或抄袭本书内容。

书 名: **JavaBeans从入门到精通**

著 者: [美] Laurence Vanhelsuwé

译 者: 邱仲潘 等

责任编辑: 郭小湘

印 刷 者: 北京顺义新华印刷厂

装 订 者: 三河市金马印装有限公司

出版发行: 电子工业出版社出版、发行

北京市海淀区万寿路173信箱 邮编: 100036 发行部电话: 68279077

北京市海淀区万寿路甲15号南小楼二层 邮编: 100036 发行部电话: 68215345

URL [http //www.phei.com.cn](http://www.phei.com.cn)

经 销: 各地新华书店经销

开 本: 787×1092 1/16 印张: 29.375 字数: 770千字

版 次: 1998年1月第1版 1998年1月第1次印刷

书 号: ISBN7-5053-4459-5/TP·2072

定 价: 49.00元

著作权合同登记号 图字: 01-97-1644

凡购买电子工业出版社的图书, 如有缺页、倒页、脱页者, 本社发行部负责调换
版权所有·翻版必究

本书谨献给一个Mus Musculus
它的憨态和大肚子给了我无穷的乐趣

致 谢

本书得以出版，首先是因为Sybex的开发编辑Suzanne Rotondo的一封e-mail，问我想不想做一本JavaBeans方面的书。感谢Suzanne播下这粒种子，产生了读者手头的这份收获。感谢编辑Maureen Adams将痛苦的手稿编辑变成了完全的享受，同时改进了书稿质量。感谢技术编辑Matt Fiedler严格检查编程中的每一处错误，感谢Robin Kibby的认真校对，感谢Maureen Forys生成了读者面前的页面，感谢Patrick Dintino将我的草图变成了精彩的艺术品。

序 言

Java已经对编程世界产生了巨大的影响，它显示出一个好的语言设计可以大大减少编程人员的工作量。指针错误、内存分配问题、繁琐而依赖于平台的库等自有计算机以来一直存在的问题终于得到了解决。

当然，Java的解决方案并不新鲜，事实上，Java最大的魅力之一正是在于它没有任何新东西。Java只是巧妙地集中了其它编程系统中久经考验并被认可的最佳要素和功能。面向对象技术是近五年来才被软件工程师接受的，但学术界已经对此推崇多年。

一个诞生于被证明是最好的软件工程技术的语言向何处发展？当然它不能盲目进入未经证实的或新出现的领域，但有一种技术在软件工程中刚刚出现，而在其它工程领域则早已被实践证明，甚至已经成为其它工程分支的绝对基础。这就是组件（component）技术。

在电子行业，设计人员通常不会直接与硅、碳、云母等原材料打交道，也不会与晶体管 and 逻辑门打交道，而是把所有功能包装为预制组件，如放大器、相锁环、调频鉴频器。手册中列出和介绍了设计人员可大量选取的组件，从头建立这些组件只是本科生的练习或研究改进时所做的工作。

在商业性（而不是学术性）软件开发中，这个思想相对较新，但正确地体现了面向对象模式：只提供清晰定义的接口而不关心内部实现细节，准确定义组件适用的条件，明确指出它如何与其它组件交互。

JavaBeans提供了接口和通讯部分的标准，提倡适当的包装，因为Bean打开的部分通常没有用处。

Laurence Vanhelsuwe在本书中不仅介绍了JavaBeans的背景，使新手能正确理解bean对现代软件工程的价值和性质，而且以清晰连贯的方式介绍了使用的技巧。

尽管bean很容易上手，但深思熟虑的bean可以广泛复用。它通常可按环境要求做些后续改动，完成其原作者难以料想的功能。本书介绍了bean开发的重要问题，介绍这些问题如何产生和如何处理。线程是许多软件工程师较不熟悉的领域，但也是许多意外错误易出现的领域，本书可以帮你处理所有不测，写成适用于各类应用程序的健全而有效的bean。

软件工程或许是最具创造性的工程，其灵活性和表达性远远超过其它受物理条件限制的学科。这种创造性使它成为一门艺术，一门全新的艺术。正是这种灵活性、这种表达想象力的无穷潜力吸引某种人投身其中。一个好的软件工程人员通常比较独立、乐观，具有完美主义的自信，具有艺术家的气质。这种人深深知道对着一件精巧的构思说声“这是我的作品”时那种得意忘形的感觉。为此，他们不厌其烦地推倒重来，精益求精。这在艺术领域是适合的，人们推崇艺术的敏锐性，但遗憾的是，在商业社会不断增加要求和缩短交货期的压力下，这种做法是不现实的。

过去有些方法学通过减少工程师发挥创造性的机会来减少错误，从而提高质量，而组件技术，特别是JavaBeans中的组件技术则让工程师从日常琐事中解脱出来，使其着眼于产品的真正创造性的方面：真正花时间解决新的、实在的和有趣的问题，建筑师设计的不是新砖

块，而是新的建筑。

同样，本书着重介绍JavaBeans重要的创造性方面，精彩地介绍每个方面如何工作，以及可以和不能如何使用。介绍每个特性时，与当前论题无关的方面避而不提，但又不忽略更复杂的交互作用。有了坚实的基础知识后，再明确地介绍交互作用、协同作用和复杂的实际情况。

无论是首次接触组件软件技术还是在旧的组件环境中工作过，本书都将带你进入JavaBeans世界，它的精巧的表达将会加强未来软件的构造与维护。

Sun系统公司软件工程师，高级教员

Simon Roberts

前 言

应用程序不断增加的复杂性和不断缩短的交货期使现代软件的开发变得更困难了。除非另有补偿这种压力的办法，否则编程生产率真是四面受敌，陷入困境。试试用C++开发当今的桌面平台（Windows, Mac, UNIX）程序，你就会理解为什么有人觉得这种情况无可救药。

突然间，柳暗花明，出现了Java，在短短的时间内，Java用格外简单和强大的语言以及核心API技术使这种复杂性大大缓解。Java使人们从烦人的软件开发马拉松中解放出来，高度抽象地处理，避免底层技术细节从背后直刺一刀。利用Java就可以将精力集中在应用程序上，不必考虑别的杂事，从而提高产品的质量。

Java在其不断前进的道路上，用一个保镖将传统编程语言（C、C++、Delphi、Visual Basic）和专用API（Windows、Mac）慢慢推到未来的计算机博物馆中，这个保镖就是JavaBeans。JavaBeans是Java的软件组件标准，软件组件将会是软件开发中一个新的大事件。JavaBeans软件组件继承了Java的平台独立性和简单性这两个标志特性，使其它软件标准（Microsoft的ActiveX、Apple的OpenDOC，等等）大为逊色。Java bean可以运行在任何支持Java的平台上，并且易于编写（这是开发人员梦寐以求的）。

本书介绍如何用JavaBeans标准建立可复用Java软件组件，如何在全面兼容的Java应用程序中使用或作为产品销售。

本书概述

第1章“JavaBeans与基于组件的软件开发”，介绍软件开发的历史层面，重点介绍Java和JavaBeans当前发生的演变，本章还介绍了软件组件的各个重要问题：对软件行业的意义，取得成功的条件，等等。第2章“JavaBeans概述”简述JavaBeans标准，并介绍bean开发工具库（BDK）。第3章“Java 1.1委托事件模型”介绍取代1.0 AWT事件系统的新AWT事件模型，这个新模型在JavaBeans中起着中心作用。第4章“bean属性”介绍JavaBeans最简单的方面：bean属性，并介绍第一个测试bean。

第5章“bean事件”中介绍bean如何通过引发和接收事件而与其它bean相互通讯，由于组件间的联系和通讯是任何基于组件设计的关键，所以bean事件是JavaBeans标准最重要的方面。第6章“Java档案（JAR）文件和API”介绍如何用新的JDK jar工具包装和压缩bean，还介绍jar的大多数功能所依赖的新的1.1 java.util.zip包。第7章“定制bean”介绍如何用JavaBeans的BeanInfo类加强bean。第8章“增加定制的属性编辑器”专门介绍bean属性编辑，用于工具环境中不支持的属性类型。第9章“实际bean的实现”列举了作为成功的工业级Java bean还应额外考虑的方面，列举了实时滚动图形的例子。

第10章“Java核心反射和bean自查”，透视了Java bean的黑盒子接口如何用新的1.1核心反映API进行自动分析。第11章“对象序列化”介绍新的1.1对象持续API及其被JavaBeans工具或同样新颖的远程方法调用（RMI）API使用时的内幕。

最后，附录A“实用程序包”分类列出了本书编写的可复用方法、类和设计模式。

假设

本书假设你可以用Java编程，具有用核心API（特别是抽象窗口工具库AWT）类建立图形用户接口的经验。不需要有对Java1.1 API的经验，但本书尽可能利用Java1.1的方法和类，因为JavaBeans本身是个Java 1.1技术，也因为1.1比1.02有了许多进步。

本书不依赖于Sun的标准JDK（Java开发工具库）1.1与BDK（bean开发工具库）1.0以外的任何Java开发工具，所需工具可以直接从Internet网络上免费下载，

免费下载BDK的站点是：http://www.sun.com/beans/bdk_download.html

免费下载JDK的站点是：http://www.sun.com/beans/jdk_download.html

根据Java的平台中立精神，我们不设定Java运行的平台，但本书的例子是用“Wintel”平台写成的，所以屏幕图形、目录清单等都反映了这个平台（我也同意买一台Sun或SGI个人工作站，但我的老板不认为这是一台“真正妙的机器”）。

关于例子源代码的说明

书上的源代码与公司开发的工业用代码有许多不同，书上的源代码着重于向读者一次说明一个或几个技术、语言特性或API特性，而实际生成代码通常（错误地）忽略了第三方读者，不幸的是，维护人员、同事和客户通常正是这群读者。

为了确保源代码的高度可读性，我尽可能一致地使用下列核心技术：

- 尽量用长的有意义的标识符而少用缩略符，复制Java核心API本身建立的精彩例子
- 充分的说明语句
- 通过方法说明块直观地分开方法定义
- 充分分解方法，避免用大而全的方法
- 类似项目用水平对齐，以便区分相同部分与不同部分
- 每个文件一个类（或接口）

此外，我们将程序清单分成几类，并加进图标，以便使各类间的差别一目了然：



齿轮图标用于包含Java类的源代码，拼板图标表示包含Java接口的源代码，齿轮图标和拼板图标的源代码可以编译，另外三个源代码类型是不可以编译的，其中合同图标表示Java类定义，铅笔图标表示伪代码，HTML图标用于生成测试小程序的页面。

所有Java源文件的开头都包含源代码头，例如：

```
//=====
//RealtimeGraph (C) Jan 1997 Laurence Vanhelsuwe-All Rights Reserved
//-----
//This bean Lets you display streaming, time-stamped real value.
//Applications include process control, medical, scientific,
//financial real-time data displays.
//
```



```
//History:
//-----
//16/01/97:started this file from new.java
//
//Author email: Lva@telework.demon.co.uk
//=====
```

这些头中包含所有程序文件都要有的关键信息：标题、作者名和与作者联系的方式，版权声明，文件内容简述和修改历史。如果森林不是紧缺的宝贵资源（即可以用很多树木来造纸），则本书的每个源代码清单都会保留这个头，但现在木材已经很紧缺，所以只好忍痛割“头”了（但本书的最大例子除外），读者不难想象每个文件的完整形式。

规则

本书用各种规则以最可读的方式表达信息，通篇用下列说明、提示和警告突出重要信息。

说明：这是说明，提供与所述过程或主题相关的信息。

提示：这是提示，提供有助于编程的指示和内部信息。

警告：这是警告，提供应避免的可能出问题的微妙或隐藏元素。

这些规则有助于本书的阅读。

目 录

第1章	JavaBeans与基于组件的软件开发	1
	简介	1
	现代软件的作用和痛苦的现实	1
	软件开发的历史：与复杂性之战	2
	40年代与第一台电子计算机	4
	50年代着重于计算机的生产率	5
	60年代：软件的复杂性导致开发成本激增	6
	70、80和90年代：已经意识到危机，但未能有所作为	6
	简单的办法：控制软件复杂性	7
	前面的路：Java与软件组件	8
	Java革命	8
	软件组件	11
	小结	16
第2章	JavaBeans概述	17
	简介	17
	bean到底是什么	17
	Java bean的黑盒子观点	18
	bean方法	18
	bean属性	19
	bean事件	19
	bean环境	19
	设计环境	20
	运行环境	21
	小程序与应用程序环境	21
	bean开发工具库和bean测试程序BeanBox	21
	BeanBox	22
	BDK演示bean	22
	java.beans包	27
	Beans类	28
	小结	29
第3章	Java 1.1委托事件模型	30
	简介	30

问题：1.0 AWT事件模型	30
解决方案：事件源和事件监听器	33
java.awt.event包	36
监听器接口	37
1.0 java.awt.Event的遗留问题	40
AWTEvent类	41
适配器类	46
事件多目广播和单目广播	49
建立自己的事件源	51
例1：AlarmTimer组件	51
例2：扩展一个AWT组件	62
AWT新问题：AWT单事件队列及其内部事件调度机制	71
小结	77
 第4章 bean属性	78
简介	78
setXXX () 和getXXX () 访问方法	79
定义读属性	79
定义写属性	80
定义读/写属性	80
bean属性类别	81
简单属性	82
布尔属性	82
索引属性	82
关联属性	83
限制属性	95
属性与多线程	105
简单属性的多线程问题	105
属性监听器和多线程	110
小结	114
 第5章 bean事件	115
简介	115
bean事件源	116
例1：目录清单bean	126
例2：图形映射bean	132
bean事件监听器	142
例1：按键事件登记bean	143
例2：建立在DirLister之上的DiskWaste bean	145

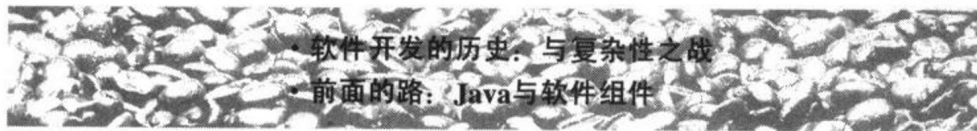
bean事件实用程序	150
例1: ActionEvent激活的Light bean	151
例2: 键/ 鼠标/ 项目事件对ActionEvent的转化器bean	160
例3: 普通AWT事件监视器bean	162
小结	164
第6章 Java档案 (JAR) 文件与API	165
简介	165
JAR命令行工具	165
建立JAR文件	166
新Applet HTML标志选项ARCHIVE	169
JAR元数据和说明文件	170
包装好的bean	171
JAR和Java包	172
java.util.zip包	173
ZipFile和ZipEntry类	173
检查和接口及相关类	179
Deflater、Inflater、DeflaterOutputStream和InflaterInputStream类	187
ZipInputStream和ZipOutputStream类	191
小结	196
第7章 定制bean	198
简介	198
java.beans.BeanInfo接口	198
java.beans.SimpleBeanInfo类	199
调整bean的黑盒子接口	201
bean特性描述符类	202
java.beans.FeatureDescriptor类	202
java.beans.PropertyDescriptor类	203
utilities.beans.VerySimpleBeanInfo类	206
java.beans.IndexedPropertyDescriptor类	212
java.beans.EventSetDescriptor类	212
java.beans.MethodDescriptor类	214
java.beans.BeanDescriptor	214
bean定制器	214
java.beans.Customizer接口	214
例子: PlugNPray bean及其基于页面的定制器	215
小结	236

第8章 增加定制的属性编辑器	237
简介	237
寻找可用的编辑器	237
java.beans.PropertyEditor接口	239
PropertyEditorSupport类	240
文本属性编辑器	241
利用属性值标志表	246
定制GUI属性编辑器	247
例1: 日期属性编辑器	247
例2: 具有正弦参数属性编辑器的利萨如bean	256
小结	291
 第9章 实际bean的实现	292
简介	292
国际化	292
java.util.Locale类	293
Visibility接口	296
行业级组件的特性	296
功能化	296
健全	297
用户友好	297
开发友好	298
纯Java	299
完整例子: 实时图表bean	299
实时图表的要求	299
实时图表分析	300
实时图表设计	300
实时图表实现	302
将RealTimeGraph bean包装成JAR	349
测试RealTimeGraph bean	349
评估和改进	350
小结	351
 第10章 Java核心反射和bean自查	352
简介	352
统一基于对象类型的系统	354
所有类型均可将其数值表示为对象	354
所有类型都有java.lang.Class表示	355

java.lang.Class类	356
java.lang.reflect包	361
Member接口	361
Field类	367
Method类	368
Constructor类	374
Array类	376
统一bean自查与java.beans.Introspector	385
小结	392
第11章 对象序列化	393
简介	393
序列化基础	394
Serializable接口	395
序列化的是什么	397
对于不可序列化对象	406
ObjectOutputStream和ObjectInputStream类	407
专用方法writeObject () 和readObject ()	409
Externalizable接口	413
类的演变	413
serialVersionUID	413
小结	417
附录A utilities包	418
简介	418
例程库	418
杂库	419
数字库	420
字符串库	423
命令行接口库	425
I/O流库	426
DebugSupport类	426
转换类	430
utilities.beans子包	430
bean库	430
VerySimleBeanInfo和VerySimlePropertyDescriptor类	431
VisibleCanvas类	435
utilities.gfx子包	436
图形库	436

Circle类	438
utilities.gui子包	442
GUI库	442
对话框库	443
选项库	447
utilities.files子包	448
文件库	448
TextFileGrinder类和TextLineProcessor接口	450
小结	451
 词汇表	 452

第1章 JavaBeans与基于组件的软件开发



简介

Java给计算机世界带来了巨大的冲击，Java在软件行业真正的革命性和里程碑式的步伐中，为一个也许更重要的技术奠定了基础，那就是JavaBeans。

JavaBeans标准预示着编程实践的一个转变，可以和汇编语言向高级语言的变革以及结构化编程面向对象编程的变革相媲美。这是因为，JavaBeans是Java平台的软件组件标准，Java可以说是本世纪末和下世纪初的C++和Windows，JavaBeans标准最终会发动酝酿已久的软件组件革命。

对于那些被报刊、杂志和Web称为全新、火爆、杀手铜式的软件技术、解决方案和标准，人们总是持以怀疑的态度，人们不止一次地看到被吹得天花乱坠的技术迅速地烟消云散，退出阵地。因此，在进入JavaBeans世界之前，首先要脚踏实地，排除眼前的雾障，冷静地分析软件领域的现状和趋势。

尽管广告的声嘶力竭徒增嘈杂，但Java和JavaBeans的确是顺应时势的结果。本章第一部分介绍历史性趋势，希望读者也能把JavaBeans看成真正至关重要的技术，而不只是今天的促销团，明天再也无影无踪。

现代软件的作用和痛苦的现实

众所周知，我们正处在一个前所未有的大变革时代，技术革命不断改变着我们的生活、社会和星球，令人耳晕目眩。似乎每天都在电子、基因工程、材料或医疗方面有新的突破，不断刺激着消费者。有一个催化剂推动着所有研究和技术发展，这个催化剂本身不象我们身边那些变革的技术，它是全人类的工具，可以称为“类技术”（Homo Technologicus），这个工具当然就是计算机。

提到计算机，这里指的就是软件，因为软件是机器的灵魂。机器是由硬件构成的，但软件比具体硬件更重要。

因此，软件是整个技术革命的核心，我们迅速地到达了这样一个时代，软件已成为技术的生命支持系统。没有软件，大多数现代人类活动（交通、商业、通讯）都将陷入瘫痪状态。因此可以说，软件本身也是一个技术成功的故事。没有软件的成功，当前的技术变革和成就只能退回二战时期的水平。和原子能发电行业有点相似的是，软件也是技术领域中最有疑点的领域之一，软件行业一直羞于承认自四十年代末就困扰着该行业的事实。

软件的问题在于，软件的发展一直受着开发综合症的折磨，即所谓软件危机。这个危机在六十年代中叶已经被认识到，但至今仍然没有得到解决。了解JavaBeans在未来的关键

作用之前，首先要了解至今仍在我们身边的危机，而JavaBeans可以成为这种危机的抗体。曙光已在前方，但走出深渊之前，首先要回顾一下过去的成败得失。只有这样，才能解决软件危机，充分意识到它的来源和特性，将它放软件技术的博物馆中。

软件开发的历史：与复杂性之战

为了了解当前软件开发现状，了解我们前进的方向，需要普通地研究一下软件和计算机行业的历史。表1.1所示的大事记列出了计算机史上的里程碑事件，具体地说，就是软件开发的历史；我们所依靠的编程语言和形成这个领域的其它因素。

表1.1 计算机史大事记

1930-1940	Vannevar Bush和John Vincent Atanasoff等专家开发了各式各样的电子计算设备，通常是些依靠二进制数字的可编程机器。在此之前，Pascal、Leibniz和Babbage等专家已经发明了各种机械计算设备，所以自动计算并不是什么新概念
1941	战时德国的Konrad Zuse推出了第三个电子计算机Z-3，这是第一个电子的程序控制的计算机
1943	Colossus计算机开始在英国政府电码和加密学校运行，Alan Turing破译了德军电代码。第二次世界大战推动了Colossus和美国的ENIAC、EDVAC等计算机的建造
1945	美国在新墨西哥州引爆了第一颗原子弹。核武器的进一步发展要求大量的计算功能来模拟核反应的物理过程，因此，核军备竞赛是战后计算机快速发展的动力之一
1945	John Von Neumann提出将数据和程序内存合并，产生Von Neumann体系（和四十年后哈佛内存体系宣称的Von Neumann瓶颈）。由于程序可以看成数据（进行电子操作），数据也可以看成程序，所以Von Neumann的统一内存模型简化了计算机的实现方法，使强大而灵活的编程技术变成可能
1947	引入数据存储的磁鼓，早期机器用阀和机电设备作为存储单元，既昂贵又容易出错，而磁鼓存储既增大内存又提高了引导可靠性
1948	贝尔实验室工程师开发了晶体管，晶体管预示着计算机微型化的第一个阶段（而IC标志第二个阶段）的到来，庞大的阀管变成了更小更节能的晶体管
1948	曼彻斯特大学Mark I是第一台使用存储程序的计算机
1949	苏联发射第一颗原子弹，美国受到核破坏的威胁，需要提高军队通讯网络以对抗核攻击，这个思想产生后来的Internet
1951	Wheeler、Wilkes和Gill发明了子程序
1951-1952	Grace Murray Hopper编写了第一个编译器A-0
1954	第一台商业性铁芯内存计算机Univac 1103A出现
1957	经过三年开发，IBM的John Backus推出了FORTRAN，这是第一个成功的高级编译语言，面向数学处理。FORTRAN标志着现代编程语言的开端
1957	苏联发射第一颗人造卫星Sputnik，由此引发了空间竞赛，产生了系统微型化的巨大技术动力。为了对苏联的技术领先作出响应，美国建立了高级研究项目局（ARPA），作为国防部的一个分支机构。ARPA后来投资研究防核爆炸的通讯网络，这是Internet的基础