



# Flex 3.0

## 入门指南

◎ 郝刚 编著

- 结合Flex技术特性，深度剖析经典案例
- 涵盖控件和容器、用户界面优化、图表等主流应用
- 领略Flex 3.0技术的强大与精妙



CD-ROM



人民邮电出版社  
POSTS & TELECOM PRESS

# Flex 3.0 入门指南

◎ 郝刚 编著

人民邮电出版社  
北京

## 图书在版编目 (CIP) 数据

Flex 3.0 入门指南 / 郝刚编著. —北京: 人民邮电出版社, 2009.5  
ISBN 978-7-115-20498-1

I. F… II. 郝… III. 软件工具—程序设计—指南  
IV. TP311.56-62

中国版本图书馆CIP数据核字 (2009) 第026789号

## 内 容 提 要

本书围绕 Flex 3.0 技术的常用技术特性展开讲解, 内容涵盖技术基础、控件和容器、优化用户界面的静态和动态特性、图表应用等。本书注重理论和实践紧密结合。书中除第 1 章外其他各章都划分为理论基础和应用实践两大部分。理论基础部分介绍相关技术特性; 应用实践部分则充分结合实际开发, 详细讲解开发过程中常见问题的解决方法。

● 本书内容丰富、结构严谨、实用性强, 适合 Flex 技术初中级水平读者阅读。

## Flex 3.0 入门指南

- ◆ 编 著 郝 刚  
责任编辑 汤 倩
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子函件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京隆昌伟业印刷有限公司印刷
- ◆ 开本: 787×1092 1/16  
印张: 18.5  
字数: 510 千字  
印数: 1—3 500 册
- 2009 年 5 月第 1 版  
2009 年 5 月北京第 1 次印刷

ISBN 978-7-115-20498-1/TP

定价: 39.00 元 (附光盘)

读者服务热线: (010)67132692 印装质量热线: (010)67129223  
反盗版热线: (010)67171154

近年来，富互联网应用程序（Rich Internet Application，RIA）的开发和应用逐渐成为业界热点。由于 RIA 具有丰富的用户界面、可提供与桌面应用程序类似的 Web 交互体验等众多优点，因此应用前景十分广泛。为此，业界各大巨头纷纷出手，推出自己的 RIA 解决方案。目前，主流解决方案包括 Adobe 公司推出的 Flex 3 技术，以及 Microsoft 公司发布的 Silverlight 2 技术。二者对比而言，Flex 3 在技术的成熟性、市场占有率等方面均超过 Silverlight 2。为了帮助国内众多开发人员迅速有效地掌握 Flex 3 技术，作者特撰写此书。希望本书能够起到抛砖引玉的作用，引领读者加快学习之路上的步伐。

Flex 3 技术涉及内容众多，为确保读者学习效果，本书并未采取面面俱到的介绍方式，而是仅对 Flex 3 的常用技术特性进行详细讲解。书中除第 1 章外其他各章都包括两大部分：理论基础和应用实践。理论基础部分主要讲解常用的技术特性，力争通过准确精炼的说明，使读者在短时间内初步掌握基础知识。在随后的应用实践部分中，选取与理论有关的多个典型示例详细讲解。本书共分为以下 6 个章节。

第 1 章 Flex 3 技术基础——本章向读者介绍 RIA 和 Flex 的基本知识，重点描绘一幅 Flex 3 技术的总体概念图。

第 2 章 控件和容器——Flex 3 内置丰富的控件和容器，它们是构建应用程序的基础。本章通过典型示例讲解控件和容器的应用方法。

第 3 章 优化用户界面（静态特性）——Flex 3 提供多个用于美化用户界面，增加丰富性和交互性的技术特性。本章重点讲解其中的静态特性，内容包括样式、皮肤、主题、滤镜、字体、工具提示、指针管理等。

第 4 章 优化用户界面（动态特性）——本章主要讲解优化用户界面的动态特性，内容包括行为、视图状态、过渡、拖放管理、项渲染器和项编辑器等。

第 5 章 数据绑定，验证和格式化——本章介绍基本数据处理的内容，包括数据绑定、验证和数据格式化的方法。

第 6 章 图表应用——本章对内置的图表控件应用方法进行详细讲解。

笔者将尽全力确保内容的准确性、实用性和可读性。由于时间仓促，加之水平有限，书中不足之处在所难免，敬请读者批评指正。

# 目 录

## 第 1 章 Flex 3 技术基础 ..... 1

### 理论基础 ..... 1

- 1.1 什么是富互联网应用程序 ..... 1
- 1.2 目前有哪些 RIA 解决方案 ..... 3
- 1.3 什么是 Flex 技术 ..... 5
- 1.4 Flex 3 具有哪些技术特性 ..... 6
- 1.5 如何理解 Flex 应用程序  
框架 ..... 7
- 1.6 如何理解 Flex 技术框架 ..... 10
- 1.7 如何学习 Flex 技术 ..... 12
- 1.8 Flex Builder 3 具有哪些  
主要功能 ..... 12
- 1.9 安装 Flex Builder 3 的  
注意事项 ..... 13
- 1.10 如何提高 Flex Builder  
性能 ..... 16
- 1.11 Flex 应用程序的开发  
步骤 ..... 18
- 1.12 如何创建 Flex 项目 ..... 18
- 1.13 设置应用程序默认属性 ..... 21
- 1.14 简介 Flex 事件模型 ..... 22
- 1.15 使用事件对象 ..... 24

## 第 2 章 控件和容器 ..... 26

### 理论基础 ..... 26

- 2.1 Flex 3 提供哪些控件 ..... 26
- 2.2 Flex 3 提供哪些容器 ..... 28

### 2.3 如何设置组件大小 ..... 30

### 2.4 组件定位和布局的方法 ..... 31

### 2.5 简介数据提供程序和集合 ..... 33

### 应用实践 ..... 35

### 2.6 如何设置 Alert 对话框 ..... 35

### 2.7 如何修改按钮控件外观 ..... 40

### 2.8 如何实现具有定时功能的 按钮 ..... 42

### 2.9 如何实现字体下拉框 ..... 45

### 2.10 如何实现联动下拉框 ..... 47

### 2.11 如何创建 Flex 播放器 ..... 50

### 2.12 如何创建文本编辑器 ..... 52

### 2.13 如何创建菜单栏 ..... 57

### 2.14 如何创建弹出菜单 ..... 59

### 2.15 如何创建不可拖动的 Alert 对话框 ..... 63

### 2.16 如何创建不可拖动的 TitleWindow 容器 ..... 64

### 2.17 如何利用 HorizontalList 控件创建相册 ..... 67

### 2.18 如何实现自定义中文 日期选择 ..... 69

### 2.19 如何为 List 控件项添加 图标 ..... 72

### 2.20 如何为 DataGrid 实现 过滤显示功能 ..... 76

### 2.21 如何为 DataGrid 实现 分页显示功能 ..... 80

|                                              |     |                                       |     |
|----------------------------------------------|-----|---------------------------------------|-----|
| 2.22 如何添加和删除 Tree 控件的节点 .....                | 86  | 鼠标 .....                              | 150 |
| 2.23 综合应用容器 .....                            | 90  | 第 4 章 优化用户界面 (动态特性) .....             | 154 |
| 第 3 章 优化用户界面 (静态特性) .....                    | 102 | 理论基础 .....                            | 154 |
| 理论基础 .....                                   | 102 | 4.1 什么是行为 .....                       | 154 |
| 3.1 在 MXML 中设置样式的方法 .....                    | 102 | 4.2 创建行为 .....                        | 156 |
| 3.2 使用 ActionScript 管理样式 .....               | 105 | 4.3 应用复合效果和缓动函数的方法 .....              | 158 |
| 3.3 Flex 3 提供的 3 种皮肤类型 .....                 | 106 | 4.4 创建视图状态 .....                      | 162 |
| 3.4 Flex 3 提供的两种字体类型 .....                   | 109 | 4.5 创建过渡 .....                        | 165 |
| 3.5 创建自定义主题的方法 .....                         | 111 | 4.6 实现拖放管理 .....                      | 167 |
| 3.6 应用主题的方法 .....                            | 114 | 4.7 创建自定义项渲染器和项编辑器 .....              | 174 |
| 3.7 使用滤镜 .....                               | 116 | 应用实践 .....                            | 180 |
| 3.8 创建工具提示的方法 .....                          | 119 | 4.8 如何为 HSlider 控件实现缓动效果 .....        | 180 |
| 3.9 管理工具提示 .....                             | 121 | 4.9 如何为 ProgressBar 控件设置完成效果 .....    | 183 |
| 3.10 实现指针管理 .....                            | 123 | 4.10 如何实现以拖放操作删除 DataGrid 数据 .....    | 186 |
| 应用实践 .....                                   | 125 | 4.11 如何创建包含 CheckBox 的 List 控件 .....  | 189 |
| 3.11 如何使用 CSS 设置 HSlider 控件样式 .....          | 125 | 4.12 如何显示 MP3 文件的 ID3 信息 .....        | 191 |
| 3.12 如何使用多种方法为 NumericStepper 控件应用内嵌字体 ..... | 128 | 4.13 如何为 Tree 控件实现拖放操作 .....          | 195 |
| 3.13 如何为工具提示设置特效 .....                       | 132 | 4.14 如何创建走马灯效果 .....                  | 202 |
| 3.14 如何实现换肤功能 .....                          | 134 | 4.15 如何在不同 DataGrid 控件之间移动和复制数据 ..... | 205 |
| 3.15 如何将彩色图片转换为黑白色 .....                     | 139 | 4.16 如何为 List 控件列表项改变时添加效果 .....      | 208 |
| 3.16 如何利用 trackSkin 属性去除滑动条皮肤 .....          | 140 | 4.17 如何为操作添加音效 .....                  | 210 |
| 3.17 如何检查颜色名字是否合法 .....                      | 143 | 4.18 如何在运行时修改项渲染器 .....               | 213 |
| 3.18 如何设置自定义拉伸光标 .....                       | 146 | 第 5 章 数据绑定, 验证和格式化 .....              | 217 |
| 3.19 如何为 VBox 容器设置 Scale-9 图形背景 .....        | 147 | 理论基础 .....                            | 217 |
| 3.20 如何显示简单动态                                |     | 5.1 数据绑定基本概念 .....                    | 217 |
|                                              |     | 5.2 实现数据绑定 .....                      | 218 |

|                       |     |                      |     |
|-----------------------|-----|----------------------|-----|
| 5.3 实现数据验证.....       | 221 | 6.2 设置图表控件.....      | 254 |
| 5.4 实现数据格式化.....      | 224 | 6.3 图表控件应用示例.....    | 255 |
| 应用实践.....             | 226 | 6.4 设置间距属性.....      | 268 |
| 5.5 如何实现界面语言          |     | 6.5 格式化刻度线.....      | 269 |
| 动态切换.....             | 226 | 6.6 使用笔触.....        | 270 |
| 5.6 如何验证信用卡号.....     | 229 | 6.7 设置填充.....        | 271 |
| 5.7 如何验证并格式化货币        |     | 6.8 处理事件.....        | 273 |
| 输入.....               | 233 | 6.9 应用效果.....        | 275 |
| 5.8 如何实现表单数据验证.....   | 235 | 应用实践.....            | 277 |
| 5.9 如何格式化颜色值.....     | 241 | 6.10 如何呈现多种类型的       |     |
| 5.10 如何验证 18 位身份证     |     | BarChart 控件.....     | 277 |
| 号码.....               | 244 | 6.11 如何为 PieChart 控件 |     |
| 5.11 如何验证 RadioButton |     | 添加旋转效果.....          | 279 |
| 控件.....               | 249 | 6.12 如何显示多坐标轴        |     |
| 第 6 章 图表应用.....       | 253 | 图表.....              | 282 |
| 理论基础.....             | 253 | 6.13 如何自定义数据标签       |     |
| 6.1 Flex 3 提供了哪些内置    |     | 外观.....              | 285 |
| 图表控件.....             | 253 | 6.14 如何显示图表中的        |     |
|                       |     | 网格线.....             | 287 |

# 第 1 章 Flex 3 技术基础

近年来,随着网络的高速发展,富互联网应用程序(Rich Internet Application, RIA)逐渐成为技术热点和发展方向。为此,涌现出多种实现 RIA 的解决方案,其中以 Adobe 公司的 Flex 技术和 Microsoft 公司的 Silverlight 为主。这些主流技术都非常优秀,都拥有众多用户和强大支持。至于谁将称雄 RIA 世界,实在是个难题,这也不在本书讨论范围之内。

从本章开始,将向读者详细介绍最新的 Flex 3 技术。本章作为开篇,主要针对读者所关心的内容和 Flex 3 技术的常见问题进行了详细讲解。希望通过这些内容,能够使读者对于 Flex 3 技术建立一定了解,从而为阅读后续章节奠定基础。

## 理 论 基 础

### 1.1 什么是富互联网应用程序

【关键字】富互联网应用程序 RIA 应用程序模型

随着互联网的迅速发展,越来越多的用户不满足于仅仅是看看网页,听听音乐等简单应用,这些用户要求通过浏览器完成功能更强,界面更赏心悦目的高级交互。然而,由于技术发展的制约等多种原因,这种需求还很难得以满足。为了提高用户体验,实现更为复杂的功能和人机交互,出现了一种新的互联网应用程序,这就是富互联网应用程序(Rich Internet Application, RIA)。那么究竟什么是 RIA 呢?

RIA 的概念由 Macromedia 公司(已被 Adobe 公司收购)于 2002 年 3 月首先提出,它是一种运行于浏览器中的网络应用程序,其突出优势在于能够将桌面应用程序具有的丰富用户交互体验,与传统 Web 应用的易于传播部署的灵活性结合起来,目的在于极大地提高用户的体验,彻底革新人与互联网之间的交互关系。

从互联网发展的角度来看,RIA 是水到渠成的结果。图 1-1 显示了 RIA 在互联网发展过程中所处的位置。

如图 1-1 所示,应用程序经过从本地向大范围直至全球,从文字界面向丰富直至综合媒体图形用户界面的方向发展着。显而易见的是,发展的总体方向是构建遍及全球的具备综合媒体图形用户界面的 RIA。相信随着软硬件技术的迅猛发展,特别是全球各大企业组织认识到 RIA 可产生显著的商业利润、提高生产率及降低成本的优势后,RIA 的发展势头会越来越猛烈,而这些结果

的核心原因是，RIA 能够最大程度的提高广泛性和丰富性。由此可见，2002 年 Macroedia 公司引入 RIA 是大势所趋，即使它不提出，也会有公司做这件事情。

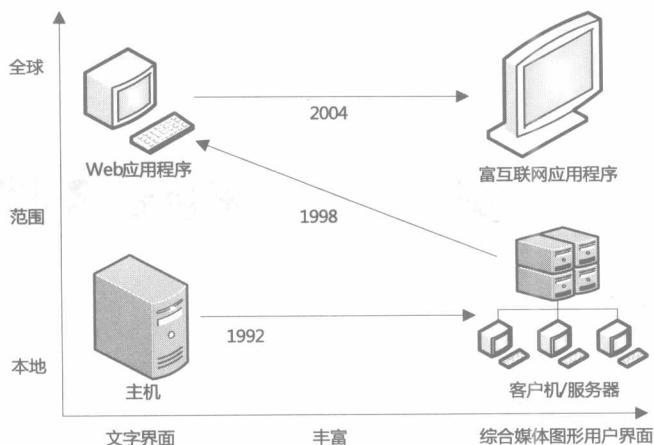


图 1-1 互联网发展过程图

从技术的角度来看，RIA 是极具革命性的概念，它是桌面应用程序与 Web 应用程序的完美结合。简单而言，RIA 所具有的桌面应用程序的特点包括：在消息确认和格式编排方面提供互动用户界面，在无刷新页面之下提供快捷的界面响应时间，提供通用的用户界面特性如拖放式及在线和离线操作能力。RIA 所具有的 Web 应用程序的特点包括：用户友好性和丰富的交互性；跨平台兼容性；一次加载，多次使用的特性；客户端数据缓存功能；高效的网络数据信息传输；多个应用同时工作；快速的开发和部署；高度的个性化可配置性；具有通信包括实时互动的声音和图像的特点，充分利用被广泛采纳的互联网标准。

以上内容是帮助读者从外部了解 RIA，下面从内部机理，即应用程序模型的角度介绍 RIA。图 1-2 显示了 RIA 的应用程序模型。

如图 1-2 所示，RIA 应用程序模型由 5 个层组成，由底至上包括：资源层、集成层、业务层、表示层和客户层。资源层中包含数据库、外部系统和传统资源。集成层实现连接到数据和传统系统。业务层实现连接业务对象。表示层包括 ASP、JSP、Servlets、Flex 和其他 UI 元素。以上四层比较常见，且都很容易理解，在此不做赘述。此处，重点说明一下客户层。RIA 中的客户层主体是富客户端，它能够提供可承载已编译客户端应用程序（以文件形式，用 HTTP 传递）的运行环境。客户端应用程序使用异步客户/服务器架构连接现有的后端应用服务器，这是一种安全、可升级、具有良好适应性的新的面向服务模型，这种模型由其所使用的 Web 服务所驱动。

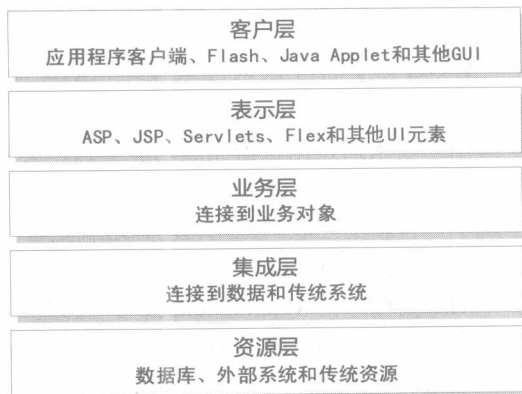


图 1-2 RIA 应用程序模型

通过以上内容的讲解，相信读者也能够总结出一些 RIA 的优点：（1）RIA 具有更加丰富的用户界面，其在不重载页面的情况下，也可提供更多迷人的用户体验；（2）RIA 可向用户提供由用户事件触发的实时反馈和验证；（3）RIA 可实现与传统桌面应用程序一致的观感；（4）RIA 能够包括完整的多媒体体验，如音频和视频；（5）RIA 具有很多功能，例如实时聊天和远程协作。这

些优势是传统 Web 应用程序难以企及的。

虽然 RIA 具有很多优点,受到很多人的青睐,但是事物的发展总是需要一个过程。目前,距离大规模开发应用 RIA 尚需时日。然而,近年来 RIA 在企业中得到了一些应用,例如,构建企业资源计划 ERP,客户关系管理系统 CRM 等。对于企业而言,部署 RIA 的好处在于,RIA 可以继续使用现有的应用程序模型(包括 J2EE 和 .NET),因而无需大规模替换现有 Web 应用程序。通过富客户端技术,可以轻松构建更为直观、易于使用、反应更迅速,并且可以脱机使用的应用程序。同时,RIA 还可以帮助企业提供多元化的重要业务效益,包括提高销量、提高品牌忠诚度、延长网站逗留时间、较频繁的重复访问、减少带宽成本、减少支持求助以及增强客户关系等。

RIA 的确具有非常耀眼的光芒,它是未来互联网应用发展的主流方向。这种发展方向既是用户的强烈需求,也是各个技术巨头争相努力的目标。对于开发人员和爱好者而言,把握这种方向就是把握未来,就能够不被时代所抛弃,从而与互联网共前进共发展。

## 1.2 目前有哪些 RIA 解决方案

【关键字】AJAX Flex Silverlight JavaFX Laszlo XUL

可能部分读者会大胆猜测:“在不久的将来,RIA 可能会完全替代目前的 HTML 应用”。对此猜测,只能说“仁者见仁,智者见智”,站在不同的角度会产生不同的观点。然而,值得肯定的是,无论上面的猜测是否成立,对于那些采用 C/S 架构的胖客户端技术运行复杂应用系统的组织,以及采用基于 B/S 架构的瘦客户端技术部署 Web 应用系统的组织而言,RIA 确实提供了一种廉价的选择。

目前,构建 RIA 的技术还处于“诸侯混战”的态势。开发人员面临多种选择,这其中包括 AJAX、Silverlight、Flex、JavaFX、Laszlo、XUL 等,它们各自有着不同的技术特点和针对性。了解这些技术,对于读者加深 RIA 的理解具有一定益处。

### AJAX

AJAX 是“Asynchronous JavaScript and XML”(异步 JavaScript 和 XML)的缩写。很多人容易将 AJAX 误解为一种新技术,它实际是几种技术以新的方式组合而成。每种技术都有其独到之处,结合在一起就形成了这种功能强大的技术。这几种技术包括:(1)基于 Web 标准的 XHTML+CSS 的表示,(2)使用 DOM(Document Object Model,文档对象模型)进行动态显示及交互,(3)使用 XML 和 XSLT 进行数据交换及相关操作,(4)使用 XMLHttpRequest 进行异步数据查询、检索,(5)使用 JavaScript 将所有的东西绑定在一起。在技术应用方面,AJAX 主要用于实现轻量级的 RIA 开发,例如,为标准网站添加丰富内容和响应特性。AJAX 的最大优点是能够在不更新整个页面的前提下维护和刷新数据,同时 AJAX 应用不需要任何浏览器插件,只需要用户允许在浏览器中执行 JavaScript 即可。AJAX 的不足主要是开发不易上手,同时还涉及浏览器兼容性问题。

### Silverlight

Silverlight(开发代号是 WPF/E,Windows Presentation Foundation Everywhere)是由 Microsoft 公司发布的基于 .NET 的全新 Web 呈现技术(目前最新是 Silverlight 2 版)。借助这种技术,开发人员能够创建出内容丰富、视觉效果绚丽的交互式用户体验,例如矢量图形、动画、视音频等。Silverlight 采用插件方式部署,这有点类似 Flash 插件。它具有良好的跨平台特性,支持多种操作系统,例如 Windows、Mac,支持多种浏览器,例如 IE、Firefox、Safari,甚至能够在移动设备上

运行。在技术应用方面, Silverlight 主要用于创建重量级 RIA 应用, 其允许在浏览器中创建更加丰富的用户界面。如果将 Silverlight 与 AJAX 结合, 利用 AJAX 的异步数据, 那么 Silverlight 将如虎添翼, 用户会得到非常舒适和丰富的交互体验。Silverlight 具有很多优点, 例如开发效率高、易于上手、运行速度快、跨平台, 允许开发者和设计人员协同工作等。

## Flex

在 Flex 出现之前, 试图解决用户界面问题的开发人员常常因为 HTML 的局限性而遭受挫折。原因很简单, HTML 起初是为浏览静态文档而设计, 并不适合于创建内容丰富、交互性强和数据驱动的应用程序。为了满足 RIA 开发人员需求, Flex 技术横空出世。对 Flex 最简单的理解是: Flex 技术能够根据 .mxml 文件 (包含 XML 和 ActionScript 代码) 生成相应的 .swf 文件, 然后传送到客户端, 由客户端的 Flash Player 或者 Shockwave Player 解释执行, 从而提供给用户丰富的内容和交互体验。实际上, Flex 并非这样的简单, 否则它也无法称为当前最完整、最强大、市场份额最高的 RIA 开发解决方案 (最强大的竞争对手是 Silverlight)。Flex 技术提供了基于标准的、声明性的编程方法和流程, 以及运行时服务。利用 Flex 技术提供的表示服务和应用程序框架, 能够创建既具有桌面应用程序的交互性和丰富内容, 又结合了 Web 的广泛性和易于部署性的应用程序。

## JavaFX

JavaFX (即 JavaFX Script) 属于 SUN 产品家族的一员, 它是 Sun 公司推出的最新 RIA 解决方案。利用 JavaFX, 开发人员能够快速创建在支持 Java 虚拟机的计算机和手机等设备中运行的应用程序。JavaFX 是一种声明式的静态类型编程语言, 它具有第一级函数、声明式语法、列表推导及基于依赖关系的增量式求值等特征, 并能够直接调用平台上的 Java API。由于 JavaFX 脚本是静态类型的, 因此它具有相同的代码结构、重用、封装特性 (例如 package、类、继承、分离编译、部署单元), 这使其能够用于创建和维护大型 Java 程序。从语法来看, JavaFX 是 Java 和 VRML 的混合体, 既具有 Java 的优点 (例如, 面向对象、继承等), 又具有 VRML 的优点 (图形描述)。在 IDE 支持方面, 目前已经推出了针对 JavaFX 的 NetBeans 和 Eclipse 的插件。

## Laszlo

Laszlo 是一个开源的 RIA 开发技术, 其类似于 Flex。由于它是开源的, 所以可免费为开发人员使用。开发人员可使用 JavaScript 和基于可扩展标记语言 XML 的编程语言 LZX 来构建 Laszlo 应用程序, 接着运行在 J2EE 应用服务器上的 Laszlo 平台会将源代码编译成 SWF 格式的文件, 最后传输给客户端展示。从这点上来说, Laszlo 的本质和 Flex 是一样的。由于 Laszlo 编译结果为 SWF 文件, 所以它与使用 Flex 技术构建的 Flash 应用程序相同, 一次编写到处运行的特点, 从而解决了浏览器之间的移植问题。另外, 在未来的计划中, Laszlo 还可以将 LZX 编译成 Java 或 .NET 本地代码, 从而大大提高运行效率。Laszlo 超越 Flex 的优点之一是, 下一个版本的 OpenLaszlo (命名代码为 legals) 将允许编写一次代码, 可编译生成 SWF、动态 HTML (DHTML) 或者 AJAX 应用程序。当然, Laszlo 也面临着支持组件不足, 市场开拓有限, 以及 Flash Player 的约束等问题。

## XUL

XUL (XML User Interface Language, XML 用户界面语言) 是一种基于 XML 的用户界面语言, 它来自于 Mozilla 的开放源码项目。XUL 中包括很多预定义的小部件 (按钮、单选按钮等)。仅有兼容 Mozilla 的浏览器 (例如 Firefox) 才能显示这些小部件。虽然开发人员可以在 RIA 的创建过程中使用 XUL, 但是 XUL 主要用于 Mozilla 应用程序和扩展应用的创建过程当中。XUL 最

大的优点在于，与大多数其他 XML 用户界面描述语言相比，XUL 是一种非常简洁和具有表达力的语言。XUL 的主要缺点是，目前 XUL 还没有获得一个主要商业实体的支持。

RIA 的解决方案比较多，并且各自有着优点和不足，这对于应用开发人员和组织而言的确有些难以取舍。目前，开发 RIA 主要使用的是 Flex 技术。在国内，虽然 Flex 的技术文档，开发工具都是英文（对于国内学习者可能会造成一定障碍），虽然 Flex 的技术推广工作还不是很到位，但是这些不足都无法掩盖 Flex 的魅力光芒，无法降低学习和使用 Flex 技术人员的热情。因此，本书将重点讲解 Flex 3 技术。

## 1.3 什么是 Flex 技术

### 【关键字】发展历程 应用程序生成和应用过程

对于 Flex 技术的基本概念理解可以从多方面加以介绍。作为初学者应由浅入深，从最基本的方面入手比较合适，例如技术发展历程，Flex 应用程序的生成和应用过程等。

### 发展历程

2004 年，Macromedia 公司发布了 Flex 1.0 技术，这是 RIA 发展过程中的重大事件，标志着 RIA 向大规模普及迈出了第一步。在经历一年的发展之后，Flex 1.0 凭借其优秀的特性得到了业界和广大开发人员的好评。此时，Macromedia 顺势发布了 Flex 1.5。该版本在上一版本的基础之上进行了很大改进，添加了很多新功能，例如新版的 DataGrid 控件，图表系列组件，皮肤功能等。由于各种原因，Macromedia 公司后来被 Adobe 公司收购。Adobe 公司在 2006 年发布了最新版本的 Flex 2，同时为了进一步扩大市场，争取更多组织和开发人员的支持，Adobe 公司宣布开发人员可完全免费的创建和发布 Flex 2 应用程序（当然一些开发工具和高级功能组件需要付费）。2008 年 2 月，Adobe 公司发布了激动人心的新版本 Flex 3。Flex 3 技术进一步迎合 RIA 的发展趋势以及市场和用户的需求，其大大超越了它的上一版本。尤其是新增的技术特性更是锦上添花，例如全面支持 AIR、新增的高级 DataGrid 控件、增加提高性能和缩减文件体积的特性、增强与后台服务器交互等。经过多年发展过程，Flex 技术已经成为目前最完整、功能最强大的 RIA 开发解决方案。

### 生成和应用过程

有关 Flex 的基本概念，Adobe 官方的定义如下：

“Adobe Flex 技术可以在企业内部或在 Web 上创建并交付丰富 Internet 应用程序（RIA），堪称最完整、最强大的 RIA 开发解决方案。它使企业能够创建个性化的丰富多媒体应用程序，极大地提高用户的体验，彻底革新人与 Web 的交互关系。”

如果读者是初学 Flex，那么可能对上面的定义会感到陌生和难以理解。实际上，每个学习和使用 Flex 技术的人都会经历这个阶段。对于以上定义的真正理解，需要读者在学习技术的过程中不断思考和实践，下面首先从最简单的介绍生成和应用 Flex 应用程序的过程开始。

图 1-3 显示了 Flex 应用程序的生成和应用过程。

如图 1-3 所示，开发人员使用两种语言 MXML 和 ActionScript 编写应用程序，然后使用编译器将

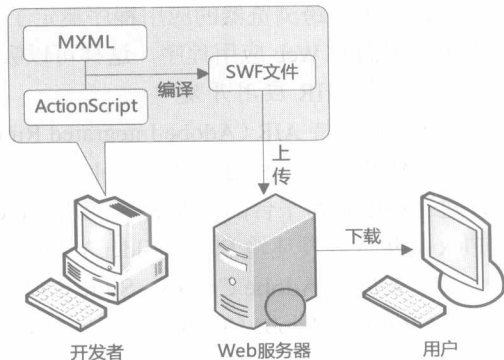


图 1-3 Flex 应用程序生成和应用过程

文件代码编译为 SWF 文件，接着将 SWF 文件上传到 Web 服务器，这样用户就能够通过远程访问 Web 服务器来访问 Flex 应用程序了。

以上过程中涉及的 MXML 语言是一种基于 XML 的标记语言，主要用于实现 Flex 应用程序的用户界面布局。ActionScript 语言是一种兼容 ECMAScript 标准的，面向对象的编程语言，其用于实现应用程序逻辑。开发人员在使用这两种语言编写应用程序时，主要面临两种选择：一种是使用记事本类型的工具编写代码，然后使用 Flex SDK 中的编译器将代码编译为 SWF 文件；另一种是使用 Adobe 公司提供的集成开发环境 Flex Builder（最新是 3.0 版）编写和编译应用程序。前一种方式是完全免费的，后者则需要付费。此外，如果开发人员愿意付出足够的费用，不仅可以得到 Flex Builder 3 的支持，而且还可以使用 Flex 图表和 LiveCycle Data Services 实现更加丰富的，具有强大图表和数据服务功能的 Flex 应用程序。

当读者读到上文提及的 SWF 文件时，可能立刻联想到 Flash，联想到二者之间的关系，甚至猜想是否 Flex 是类似于 Flash 用于构建动画的技术。这的确是一种误解。

虽然 Flex 创建的 SWF 文件能够使用 Flash Player 播放，但是 Flex 是一种用于开发 RIA 应用程序的技术，其与使用 Flash 技术构建的所谓 RIA 完全不同。所有的 Flex 开发都基于自身提供的技术框架，这种技术框架提供了可重用和可扩展的组件、数据服务、数据处理方法等。这些都是 Flash 所不具备的。另外，Flash 是面向设计人员的技术，而 Flex 技术的使用对象是编程开发人员，而且这些人员必须具备面向对象的知识，能够使用 MXML 和 ActionScript 语言编写程序代码。

就目前而言，Flex 技术是构建 RIA 的最佳解决方案。无论国外国内，多数组织和人员都会选择 Flex 作为 RIA 应用开发技术。同时，开发 Flex 技术的 Adobe 公司也在业界和市场上加大了宣传力度，并且加紧该项技术的研发工作。目前，已经发布的最新版本是 Flex 3。据悉，Flex 4.0 的研发已经启动，将于 2010 年左右公布于众。总之，Flex 是一种非常有前景，非常有趣的开发技术。

## 1.4 Flex 3 具有哪些技术特性

### 【关键字】技术特性

目前，已经有多家著名企业使用 Flex 技术实施应用开发，例如 SONY、ebay、SAP、CISCO 等，其所构建的应用范围包括富媒体应用程序、电子商务、数据可视化和在线产品定制等。这些复杂的应用与 Flex 技术所具有的丰富技术特性有着密切联系。究竟最新版的 Flex 3 具有哪些技术特性呢？

下面简单总结了 Flex 3 的主要技术特性。

#### ● 可创建跨平台跨浏览器的 Web 应用程序，并为其提供丰富的用户体验

Flex 应用程序运行在 Adobe Flash Player 9 中，而 98% 以上的计算机中都已经安装该播放器，这为创建跨平台跨浏览器的应用程序奠定了坚实基础。使用 Flex 3 技术能够创建具有高交互性和优秀用户界面的 Web 应用程序，这样可以吸引更多用户，提高效率和满意度。

#### ● 支持 AIR 应用开发

Flex 3 还支持 AIR (Adobe Integrated Runtime) 应用程序开发。AIR 是一个跨平台的运行程序。AIR 能够使开发人员利用现有的 Web 开发技能（包括 Flash、Flex、HTML、JavaScript、Ajax）优势，建立和配置跨平台的桌面 RIA 应用。Flex 3 对 AIR 进行全面支持，其内置新的 AIR API，同时 Flex 的集成开发环境 Flex Builder 3 也对创建 AIR 应用程序进行了优化。

#### ● 支持与 Adobe Creative Suite 3 的协同工作

将 Flex 3 与 Adobe Creative Suite 3 结合应用能够实现流畅统一的工作流程。例如，使用 Flash、Fireworks、Illustrator 和 Photoshop 能够创建为 Flex 所接受的资源，接着利用 Flex Builder 3 就可轻松导入这些资源，并使之成为 RIA 用户界面的构成部分。

### ● 提供高级数据服务

开发人员利用开源 BlazeDS 能够轻松地将 Flex 和 AIR 应用程序与后台分布式数据和 Java 服务器之间连接起来,从而实现实时的数据驱动的交互应用。

### ● 支持开源

Adobe 公司已经宣布对 Flex 3 开源。这样,任何人都能够免费使用,并且对开源框架进行扩展和改进,从而满足自身需要。这是 Flex 3 进一步走向普及,对抗竞争对手的重要措施。

### ● 有效提高开发人员生产效率

为了提高开发人员生产效率, Flex 3 内置了很多应用程序组件,这些组件能够帮助开发人员快速构建规范丰富的用户界面,例如文本类控件、数据和图表组件、各种容器等。另一方面, Flex Builder 3 还为开发、调试、部署应用程序提供了强大支持。

以上提及的 Flex 3 的技术特性,它们只是其丰富特性中的“冰山一角”。对于这些内容,建议读者稍加阅读了解即可,更为重要的是在实际开发过程中,边学习边体会,这样才能真正了解 Flex 技术。

## 1.5 如何理解 Flex 应用程序框架

【关键字】应用程序框架 MXML ActionScript3 Flex 类库

应用程序框架是每种技术提供用于构建应用程序的核心。那么如何理解 Flex 应用程序框架呢?该框架中包括哪些内容呢?

图 1-4 说明了 Flex 技术的应用程序框架。

简单而言, Flex 提供的应用程序框架可总结为“两种语言,一个库”。两种语言是指 MXML 和 ActionScript (最新是 3.0 版),一个库是指 Flex 类库 (Flex Class Library)。MXML 是一种标记语言,其主要用于定义应用程序用户界面。ActionScript

是一种面向对象的语言,主要用于实现业务逻辑和程序控制。Flex 类库中包括很多与组件、图表、布局管理、行为、数据访问、事件等有关的类,使用前面两种语言都可以访问这些类。为了增加可扩展性和灵活性,开发人员还可以基于 Flex 组件的开发模型和业务需要自定义各种组件。

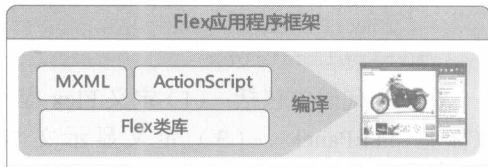


图 1-4 Flex 应用程序框架

### MXML

MXML 是 Macromedia 公司于 2004 年发布的一种基于 XML 标准的标记语言。该语言使用标签声明来描述 Flex 应用程序的用户界面。例如,使用<mx:Button>定义按钮,使用<mx:TextInput>定义文本输入框等。这一点与 HTML 非常类似,二者的不同之处在于, MXML 提供的标签更丰富、功能更强大,而且更为规范和准确。另外, MXML 也可以定义应用程序中的非可视化组件,例如数据绑定、效果、行为等。

使用 MXML 语言编写的应用程序扩展名为.xml,其内容类似于 XML 文件代码,这样的文件还不能显示所实现的用户界面。开发人员必须使用 Flex 编译器将.xml文件编译为.swf文件,然后再将.swf文件嵌入浏览器来方便用户访问。如果使用 Flex Builder 3 创建.xml文件,那么这个过程会变得很简单,开发人员只需关注用户界面的设计即可,具体代码实现、编译、生成、发布等方面都会很轻松。

以下示例使用 Flex Builder 3 创建了一个简单的 Flex 应用程序,该应用程序使用 MXML 语言定义用户界面。具体效果如图 1-5 所示。

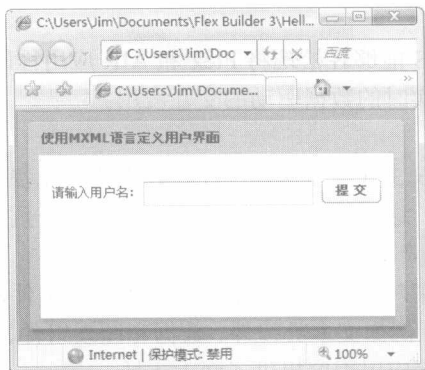


图 1-5 在浏览器中运行的应用程序

如图 1-5 所示,从表面来看,该应用程序用户界面中包括了文本、输入框、按钮和面板等元素。开发人员应该使用 MXML 语言分别定义这些内容。具体代码如下所示:

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="absolute">
  <mx:Panel x="10" y="10" width="354" height="200" layout="absolute" title="
使用 MXML 语言定义用户界面" fontSize="12">
    <mx:Label x="10" y="26" text="请输入用户名:" fontSize="12"/>
    <mx:TextInput id="txt1" x="98" y="26"/>
    <mx:Button id="btn1" x="266" y="24" label="提交" fontSize="12"/>
  </mx:Panel>
</mx:Application>
```

在以上代码中包括: (1) 定义 Flex 应用程序的根标签<mx:Application>; (2) 定义 Panel 容器的标签<mx:Panel>; (3) 定义显示文本的标签<mx:Label>; (4) 定义文本输入框的标签<mx:TextInput>; (5) 定义按钮的标签<mx:Button>。此外,根据需要还为这些组件定义了各自属性,例如设置文本大小的 fontSize 属性,定义坐标位置的 x 和 y 属性等。

## ActionScript 3

ActionScript 是一种面向对象的编程语言,主要用于实现 Flex 应用程序中的业务逻辑和程序控制。该语言和 JavaScript 非常相似,因为它是基于国际标准的脚本编程语言 ECMAScript 来实现,而这个标准和 JavaScript 之间有着密切的联系。尤其需要提及的是,ActionScript 语言提供了很多针对浏览器的支持,以及一些高级技术特性,例如强类型、接口、委托、命名空间和错误处理等。此外,使用 ActionScript 编写的脚本经过编译器编译之后,将直接生成可运行在 Flash Player 中的机器码,这样自然就能够提供更高的性能和更有效的内存管理。

MXML 与 ActionScript 之间有着密切联系。MXML 标签与 ActionScript 类或者类中的属性相互对应。当编译 Flex 应用程序时, Flex 编译器会解析 MXML 标签,并且生成相应的 ActionScript 类,然后这些类被编译为 SWF 文件。另外,开发人员还可以使用 MXML 或者 ActionScript 语言创建自定义组件。

在开发应用程序过程中,开发人员有两种组织 ActionScript 代码的方式。一种是将 ActionScript 3 代码作为脚本块添加到 MXML 文件,另一种是创建单独的 ActionScript 3 代码文件,然后在 MXML 文件中加以引用。下面首先通过示例说明第一种组织方式。

如图 1-6 所示,当用户在文本框中输入文本并单击“提交”按钮之后,在下方立刻显示出包括刚刚输入内容的一行文本。

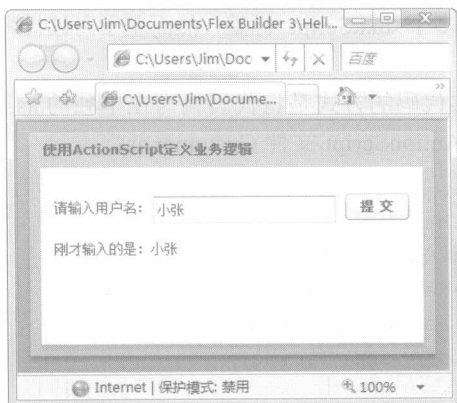


图 1-6 在浏览器中运行的应用程序

该应用程序的用户界面使用 MXML 语言定义，业务逻辑使用 ActionScript 语言实现。具体代码如下所示：

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="absolute">
  <mx:Script>
    <![CDATA[
      public function btn1Click():void
      {
        lbResult.text = "刚才输入的是: " + txt1.text;
      }
    ]]>
  </mx:Script>
  <mx:Panel x="10" y="10" width="354" height="200" layout="absolute" title="
使用 ActionScript 定义业务逻辑" fontSize="12">
    <mx:Label x="10" y="26" text="请输入用户名: " fontSize="12"/>
    <mx:TextInput id="txt1" x="98" y="26"/>
    <mx:Button id="btn1" x="266" y="24" label="提交" fontSize="12" click=
"btn1Click()"/>
    <mx:Label x="10" y="63" fontSize="12" id="lbResult"/>
  </mx:Panel>
</mx:Application>
```

以上内容中包括了 MXML 和 ActionScript 代码，其中 ActionScript 代码作为脚本块嵌入到了 MXML 代码中。正如代码所示，脚本块的嵌入仅允许使用以下标签：

```
<mx:Script>
  <![CDATA[
    ...
  ]]>
</mx:Script>
```

在这个脚本块中可插入根据业务逻辑实现的函数等内容。需要注意的是，开发人员还不能在 `<mx:Script>` 标签中定义类或者接口，因为已经定义的 MXML 文件本身就是一个类。读者可能已经注意到单个的 CDATA 结构。该结构告诉编译器不要将脚本块解释为 MXML，而应该解释为 ActionScript。`<mx:Script>` 标签必须位于 MXML 文件顶级组件标签（本例中是 `<mx:Application>`）的

所在位置。同时，还可以定义多个<mx:Script>标签，然而基于可读性考虑，推荐将这些标签置于同一个位置。

另一种组织 ActionScript 代码的方式是将代码编写在单独的.as 文件中，然后在.mxml 文件中进行引用。例如，创建如下 ActionScript 文件 Customer.as：

```
public class Customer {
    private var _id:int;
    private var _firstName:String;
    private var _lastName:String;
    private var _mobilePhone:String;
    private var _officePhone:String;
    ...
    public function get id ():int
    {
        return id;
    }
    ...
}
```

在单独的.as 文件中，开发人员可充分使用 ActionScript 功能，例如创建类、接口和函数等，而不必受到嵌入脚本块的特定约束。

在 MXML 文件中引用 Customer.as 的方法，如下所示，即设置<mx:Script>标签中的 source 属性即可：

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="absolute">
    <mx:Script source="Customer.as" />
    ...
</mx:Application>
```

## Flex 类库

Flex 应用程序是基于组件模型而创建，因此丰富的组件是 Flex 类库的重要内容之一。例如，Flex 类库包括丰富的组件（容器和控件）、图表、布局管理、行为、数据访问和事件等有关的类，这些类采用包的形式加以组织，以方便调用。下面列举了一些常见的包：

- mx.controls——该包中包括 Flex 用户界面控件；
- mx.collections——该包中包含 Flex 集合组件；
- mx.charts——该包中包含与 Flex 图表相关的类；
- mx.events——该包中包含定义 Flex 事件对象的类；
- flash.filesystem——该包中包含用于访问文件系统的类；
- flash.display——该包中包含 Flash Player 用于构建可视化显示的核心类。

除了以上提到的包之外，Flex 类库中还包括很多功能强大的包和类，限于篇幅就不多做说明了。

## 1.6 如何理解 Flex 技术框架

【关键字】技术框架 SDK Flex Builder 3 高级数据服务 组件 Flash Player 9

技术框架是为实现设计目标而构建的技术核心。Flex 技术框架由 4 个关系紧密的实体组成，