信息科学与技术丛书 程序设计系列

Flex与ActionScript编程

王睿 编著

附光盘

- 全面介绍了 Flex 和 ActionScript
- 应用案例贴近实际，富有启发性
- 从编程的角度出发讲解 ActionScript 和 Flex 的应用方式
- 附带大量可运行的样例代码

机械工业出版社
CHINA MACHINE PRESS

信息科学与技术丛书
程序设计系列

Flex 与 ActionScript 编程

王 睿 编著

机械工业出版社

本书系统地讲解了作为RIA（Rich Interactive Application，丰富的交互应用程序）开发手段之一的Flex技术。前3章介绍了Flex技术的基础知识、Flex开发环境的配置以及MXML语言的语法规则；第4到第8章详细阐述了作为Flex开发核心部分的ActionScript 3的语法及其面向对象的语言特性；第9到第18章介绍了Flex开发中的常用内容；第19和20章介绍了一些常用的ActionScript类型和用于运行Flex程序的Flash Player的安全性问题。

本书适用于希望系统地学习Flex的读者，也可供正在使用Flex进行软件开发的程序员参考。书中包含了大量的代码实例，读者可通过运行这些实例程序来加深对Flex技术的了解。随书光盘中含有书中的源代码。

图书在版编目（CIP）数据

Flex与ActionScript编程 / 王睿编著. —北京：机械工业出版社，2008.1
（信息科学与技术丛书程序设计系列）
ISBN 978-7-111-23252-0

Ⅰ. F… Ⅱ. 王… Ⅲ. ① 软件工具—程序设计 ② 动画—设计—图形软件，Flash ActionScript Ⅳ. TP311.56 TP391.41

中国版本图书馆CIP数据核字（2008）第008554号

机械工业出版社（北京市百万庄大街22号 邮政编码100037）
策　　划：车　忱
责任编辑：车　忱
责任印制：李　妍

保定市中画美凯印刷有限公司印刷
2008年2月·第1版第1次印刷
184mm×260mm·23.5印张·579千字
0001—5000册
标准书号：ISBN 978-7-111-23252-0
ISBN 978-7-89482-569-8（光盘）
定价：41.00元（含1CD）

凡购本书，如有缺页，倒页，脱页，由本社发行部调换
销售服务热线电话：（010）68326294
购书热线电话：（010）88379639　88379641　88379643
编辑热线电话：（010）88379753　88379739

出 版 说 明

随着信息科学与技术的迅速发展，人类每时每刻都会面对层出不穷的新技术、新概念。毫无疑问，在节奏越来越快的工作和生活中，人们需要通过阅读和学习大量信息丰富、具备实践指导意义的图书，来获取新知识和新技能，从而不断提高自身素质，紧跟信息化时代发展的步伐。

众所周知，在计算机硬件方面，高性价比的解决方案和新型技术的应用一直备受青睐；在软件技术方面，随着计算机软件的规模和复杂性与日俱增，软件技术受到不断挑战，人们一直在为寻求更先进的软件技术而奋斗不止。目前，计算机在社会生活中日益普及，随着因特网延伸到人类世界的层层面面，掌握计算机网络技术和理论已成为大众的文化需求。由于信息科学与技术在电工、电子、通信、工业控制、智能建筑、工业产品设计与制造等专业领域中已经得到充分、广泛的应用，所以这些专业领域中的研究人员和工程技术人员越来越迫切需要汲取自身领域信息化所带来的新理念和新方法。

针对人们对了解和掌握新知识、新技能的热切期待，以及由此促成的人们对语言简洁、内容充实、融合实践经验的图书迫切需要的现状，机械工业出版社适时推出了“信息科学与技术丛书”。这套丛书涉及计算机软件、硬件、网络、工程应用等内容，注重理论与实践相结合，内容实用，层次分明，语言流畅，是信息科学与技术领域专业人员不可或缺的图书。

现今，信息科学与技术的发展可谓一日千里，机械工业出版社欢迎从事信息技术方面工作的科研人员、工程技术人员积极参与我们的工作，为推进我国的信息化建设作出贡献。

机械工业出版社

前　言

Flex 是 Adobe 公司新推出的一种用来开发 RIA 的解决方案。开发者使用 Flex 可以迅速创建具有丰富组件内容、特殊的可视化效果并且高效运行的客户端体验式应用程序。许多国外的大公司已经开始应用这项技术，譬如 eBay、Amazon、Yahoo!等。

随着 Flash 技术的发展，Flash 的应用也逐渐演化成动画设计和交互设计两个方向。Flex 的出现可以说是把交互设计从 Flash 应用中剥离出来，开拓了一片更为专业、更为有效的交互设计的新天地。

RIA 这个概念已经出现了一段时间。起初，RIA 被 Adobe 公司定义为 Rich Internet Application（丰富的 Internet 应用程序），后来微软公司又将 RIA 定义为 Rich Interactive Application（丰富的交互应用程序）。作为一种创建 RIA 的工具或者手段，Flex 技术可以说既能够满足互联网应用又能够满足交互应用。

RIA 时代的到来是计算机世界发展的大势所趋，随着网络、计算机硬件和软件的发展，传统的基于浏览器的互联网应用程序必将被新一代的 RIA 取代，而在这场演变中的一个重要角色之一就是 Flex 技术。

事实上，以 Flex 编写的程序并不仅仅只能运行于基于互联网的浏览器中。2007 年初，Adobe 公司再次重拳出击推出了 AIR 技术。该技术让那些运用 Web 技术语言的开发者可以使用同样的语言编写跨操作系统的桌面式应用程序。这些网络技术包括 Flash、Flex、HTML、Java Script 和 AJAX。开发者编写一次就可将该程序以桌面程序的方式运行于 Windows 或者 Mac 这些不同的操作系统上。为了达到出色的界面布局和交互性能，创建 AIR 应用程序首推 Flex 技术。

而对于非 PC 终端的手机来说，目前已经有很多手机支持 Flash 播放，也就是说以 Flex 编写的程序同样可以运行于手机客户端中。例如，2007 年最有名的手机 iphone 上的许多界面都是 Flex 编写的。

当我第一次接触到 Flex 技术时，我的第一印象是，这个基于浏览器的应用程序真是太酷了。随着对这门技术的深入了解，我发现 Flex 技术正是我一直都在追寻的东西，于是撰写了此书。本书适用于那些希望系统学习 Flex、MXML 和 ActionScript 的读者，同样它也可供具有 Flex 开发经验的读者参考。书中的实例源代码可在随书光盘中找到。

由于编写时间仓促，书中难免有不足和疏漏，恳请读者指正。

谨以此书献给我的妻子秦臻，感谢她对我的支持和鼓励。同样感谢我的父母和姐姐。

王　睿

2007 年 9 月

目　录

第1章 Flex概述

1.1 什么是Flex

Flex从本质上说是Adobe公司开发RIA（Rich Interactive Application，丰富的交互应用程序）的一种技术框架，也可以说是一系列产品，它包括Flex SDK、Flex Builder、Flex Data Service和Flex Chart。Flex SDK是开发Flex应用程序的软件开发包，它是免费的，开发者可以从Adobe的网站上直接下载。Flex Builder是开发Flex应用程序的IDE，它是基于Eclipse开发环境的图形化开发工具，但它不是免费的。Flex Data Service是用来开发Flex应用程序和服务端的数据进行交互的服务组件，它包括JMS、Java Remote Object通信等高级通信方式。在具有单个CPU的计算机上使用Flex Data Service是免费的。Flex Chart是Flex提供的一套用来开发交互生动的图表程序的软件类库包，它的使用是需要得到许可的。

本书中所要讲述的是Flex2 SDK（到本书编写完毕的时候，Adobe公司已经推出了Flex3 Bata）。随着Flex在RIA领域的不断发展，无论是Adobe公司还是其他第三方软件提供商都会提供更多的像Flex Chart这样的软件类库包。要想迅速掌握如何使用这些软件包或者是其他Flex的高级特性，开发者必须系统、熟练地掌握Flex SDK中的框架结构、编程语言和基本的类库使用方法。

本书要传达给读者的是Flex SDK的软件框架结构究竟是什么样的，如何通过Flex的开发语言去编写Flex应用程序，以及Flex中一些常用的类库的使用方法。

1.2 Flex可运行于什么环境

Flex可运行于任何具有Flash Player 9的终端内。这些终端程序可以是浏览器程序，Adobe AIR发布的桌面程序，在将来甚至是手机或者PSP游戏机内的某个程序。

1.3 Flex应用程序的开发语言

Flex是开发RIA的一种软件工具，很多有经验的开发者一定好奇，它的编程语言究竟是什么样的呢？

开发Flex应用程序可以使用MXML或者ActionScript两种语言。其中MXML是XML语言的一种扩展，它是完全根据Flex SDK中的某些类别所作的XML扩展，这种类别主要是图形化界面上的类。本书所讲的ActionScript指的是ActionScript 3，它是一种基于面向对象编程的语言。

注意： ActionScript 3和以前版本的ActionScript有着比较大的差别。

在编译 Flex 应用程序时，MXML 被编译器映射转换到对应的 ActionScript 类别上，然后无论是 ActionScript 编写的还是由 MXML 映射转换过来的 ActionScript 代码都被统一转换成字节，编译为可被 Flash Player 中的 AVM2 识别的 SWF 文件。

注意：AVM 是 ActionScript 的虚拟机，目前的 Flash Player 9 内嵌 AVM 2 和 AVM 两种虚拟机。AVM 2 用来解释 ActionScript 3 部分的语言，而 AVM 用来解释 ActionScript 3 版本以前的 ActionScript。

一般情况下，开发者开发 Flex 应用程序时多采用 MXML 语言来编写界面部分的逻辑，而用 ActionScript 来处理其他更复杂的逻辑，比如数据交互。然而 MXML 实际上就是封装成 XML 的 ActionScript，开发者要想更透彻地掌握 MXML 语言或者要实现更加复杂的功能，必须熟练掌握 ActionScript 语言。

1.4 Flex 与 Flash 的关系

Flex 可以说是从 Flash 衍生出来的一种新的技术，但它是完全基于 Flash 技术的。

首先，客户端运行 Flex 程序必须在 Flash Player 9 或者其更高的版本中。其次，Flex 中的类库是基于 Flash API 的。这就是说那些适用于 Flash 的类库，配置方法等同样都可适用于 Flex。

与 Flash 相比，Flex 提供了更丰富的交互控件，更强大的数据模型和数据交互方式。如果开发者要想开发一个具有 RIA 特性的程序，使用 Flex 是其首选，而要开发一个完全的动画程序，还得选择 Flash。

第 2 章　Flex 基础知识

2.1　配置 Flex SDK 开发环境

2.1.1　安装 JDK

Flex 应用程序在开发阶段是依赖于 Java 开发环境的，所以开发者首先要安装 Java 运行时环境 JRE。读者可从 SUN 的网站下载最新版本的 JRE 或 JDK，下载地址为 http://java.sun.com/。

注意：安装 JDK 比安装 JRE 会占用更多的硬盘空间，因为 JDK 还附带了开发 Java 语言的 SDK，而仅安装 JRE 的时候只是安装 Java 语言的运行时环境。安装 JDK 时务必选择安装 JRE。

2.1.2　安装 Flex SDK

从 Adobe 网站下载 Flex SDK2 后，解压到本地目录。下载地址为 http://www.adobe.com/cfusion/tdrc/index.cfm?product=flex。

2.1.3　配置系统环境变量

假设在 Windows 环境下，Flex SDK 被解压到了 C:\Flex_sdk_2 目录内。为了能以命令行的方式通过 SDK 直接开发 Flex 应用程序，开发者还需要设置一些额外的环境变量。

图 2-1 展示的是 Windows 操作系统的环境变量的设置面板。右键单击桌面上的“我的电脑”，选择“属性”→“高级”→“环境变量”，选择“Path”变量并进行编辑，将 C:\flex_sdk_2\bin 添加到该变量值内。注意，用“;”（分号）分隔前面已设置好的变量内容。

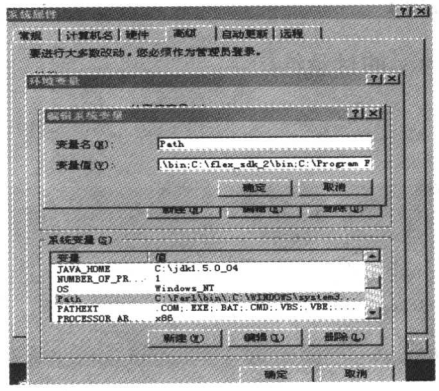

图 2-1　添加 Flex SDK 的 bin 目录到系统的 Path 变量内

苹果机、UNIX、Linux 用户在配置 Flex SDK 的命令行开发环境时，也需要把 Flex SDK 的 bin 目录加入到系统的 Path 变量内。但需使用“:”（冒号）分隔前面已设置好的变量内容。

在设置好了开发环境之后，为了检验是否配置正确，我们可以键入以下命令：在 Windows 下运行 cmd 命令，键入 mxmlc；在苹果机、UNIX、Linux 系统的终端窗口内键入 mxmlc，如果出现 Adobe Flex Compiler 的字样，则表示环境变量配置成功。图 2-2 显示了在 Windows 环境下配置完毕后，运行 mxmlc 后输出的结果。

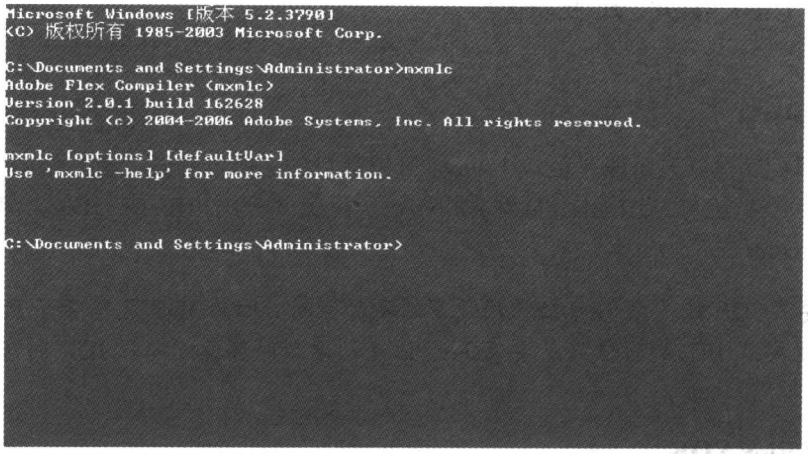

图 2-2　检验 Flex 配置开发环境在 Windows 系统下的输出结果

提示：mxmlc 是 Flex 的编译指令，它可以把 mxml 文件、ActionScript 文件或者 css 文件编译成 Flash Player 可以识别的 SWF 格式的二进制文件。

到此为止，我们已经配置了一个基于 Flex SDK 命令行的开发环境。接下来，我们需要了解更多关于 Flex SDK 的内容。

2.2　Flex Framework

2.2.1　Flex SDK 的目录结构

Flex SDK 应有如图 2-3 所示的目录结构。

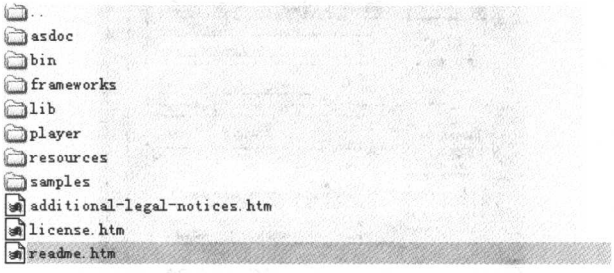

图 2-3　Flex SDK 的目录结构图

（1）asdoc：该目录包含用来创建 html 帮助文档的命令行工具和样式资源。

（2）bin：该目录包含 mxmlc 编译器、compc 编译器、fdb 调试器、asdoc 文档生成指令和 Java 虚拟机设置。

（3）frameworks：该目录包含 Flex 在运行时所需要的所有内容，如运行时所需的公共类库 libs/framework.swc、运行时的所有主题、Flex 的部分源代码、Flex 编译器的选项配置文件和其他的配置文件。flash-config.xml 是一个比较重要的配置文件。frameworks 的子目录有：

1）libs：目录内存放的是 Flex 程序运行时所需要的类库，比如 framework.swc、rpc.swc。

2）locale：目录存放的是本地化的资源，目前的 Flex SDK 只有对 en_US 的支持。

3）source：目录存放的是 Flex 提供的开源的部分代码。

4）themes:目录存放的是 Flex 所提供的主题。主题会在后面的章节中介绍。

（4）lib：该目录包含编译 Flex 程序时所需要的 Java 类库文件。

（5）player：该目录包含一个调试版本的 Flash Player 9。此调试版播放器与正式版播放器的不同在于，它能在调试的时候输出更多有用的文字信息。

（6）resources：该目录包含一些在发布 Flex 应用程序时所需的 HTML 模板和 JavaScript 的封装类代码。

（7）samples：该目录包含一些 Flex 应用程序自带的样例及其源代码。

2.2.2 Flex 程序的生命周期

由于 Flex 程序在编译后生成 SWF 格式的文件，它必须在 Flash Player 中运行。与 Flash 不同的是，通过 Flash 专业工具制作出来的 Flash 程序是通过时间轴来播放动画效果的，也就是说所有的动画都是制作在各个不同的帧上的，然后在播放时通过时间轴把这些帧上的内容“串连”起来，从而组成一个完整的动画。与 Flash 程序不同，Flex 应用程序总共只有两帧，第一帧显示的是载入 SWF 文件时读取下载 SWF 文件的进度信息，第二帧显示的是下载完成后执行的 Flex 程序的用户界面。因此，在制作具有时间轴效果的动画时应该首选 Flash 制作工具，而制作与时间轴无关的动画的 RIA 交互程序时则应首选 Flex。

2.2.3 Flex 在运行时和 Flash Player 的关系

Flex 在一定意义上可以解释为运行在 Flash 基础上的一种框架技术。缺少了 Flash，Flex 也无法运行。这是由于，Flex 所提供的类库，比如图形化控件或者布局控件等都是基于 Flash API 的，没有了 Flash API，Flex 应用程序是不能独立运行的。

Flash Player 中已经默认自带了对 Flash API 的支持，这样 Flex 程序只需要把自己需要的“额外”的 Flex 部分 API 加入到编译生成的 SWF 文件后，再统一由 Flash Player 去运行就可以了。这也是为什么 Flex 编译后的 SWF 文件比 Flash 制作的 SWF 大一些的原因（这里对比的是同样采用 ActionScript 编写后生成的 SWF 文件，不能包括那些导入的图片、视频资源等）。随着网络带宽的发展，将来 Flash Player 也许会默认自带对 Flex API 的支持。对比 Flex 能提供给我们的体验式交互效果，这一点点增大的文件结果不会对使用 Flex 有任何影响。

2.2.4 编译 Flex 程序常用的指令

编译 Flex 程序时一般只用到 mxmlc 和 compc 指令。mxmlc 编译指令用来把 mxml 文件或者 ActionSctipt 文件编译成 Flash Player 能识别的 SWF 格式的二进制文件。compc 编译指令用来把 mxml 或者 ActionScript 文件编译成 Flex 程序运行时需要的 SWC 格式的类库文件。

提示：SWC 文件实际上就是普通的 zip 格式的文件，它内含一个文件装载清单和一个带有运行库的 SWF 文件。在 Flex 中这种技术被称为 RSL（Runtime Shared Libraries）。应用这种技术可以减少 Flex 程序所需要的硬盘空间。假设 A 程序需要类库 Lib，而 B 程序也需要类库 Lib，如果不应用 RSL 技术，那么下载到客户端的 A 和 B 程序都会分别包括类库 Lib 的内容，如果能把公用的类库提取出来，让 A 和 B 共用，这样就能节省一部分硬盘空间，而且便于管理公共库 Lib 的内容，这就是 RSL 所能达到的目的。本书中对 RSL 部分没有详细介绍，读者可去 Adobe 网站查阅其使用方法。

开发者在使用 mxmlc 或者 compc 指令时可以键入

```
mxmlc –help 或者 compc –help
```

查看该指令的具体选项。下面我们介绍几个这些指令的常用方法。

```
mxmlc -file-specs myApp.mxml
```

上面的指令用-file-specs 选项指定编译名为 myApp.mxml 的源文件。编译后将在当前目录下默认生成一个名为 myApp.swf 的文件。

```
mxmlc myApp.mxml
```

上面的指令没有用任何编译选项，它将编译名为 myApp.mxml 的源文件。编译后将在当前目录下默认生成一个名为 myApp.swf 的文件，这是由于 mxmlc 默认的指令选项指令就是-file-specs。

```
mxmlc myApp.mxml -output bin/main.swf
```

上面的指令使用-output 选项指令，它将编译名为 myApp.mxml 的源文件。编译后将在当前目录的 bin 目录下生成名为 main.swf 的文件。

```
mxmlc -source-path . C:\FlexSource myApp.mxml
```

上面的指令使用-source-path 选项指令指明了编译时的源代码路径除了当前目录.外，还有 C:\FlexSource 目录内的文件也将作为源代码路径。编译时会在指定的源代码路径下查找 myApp.mxml 文件和其他需要的相关文件。编译后将在当前目录下默认生成一个名为 myApp.swf 的文件。

除了使用这种添加命令行选项的方式去编译 Flex 应用程序外，还可以使用外部配置好的编译选项配置文件去编译 Flex 应用程序。

```
mxmlc -load-config=configuration.xml myApp.mxml
```

上面的指令通过-load-config 选项指定了当前目录下的 configuration.xml 作为编译的配置文件去，当编译 myApp.mxml 的时候，将根据 configuration.xml 中的 xml 结构的内容去编译 Flex 程序。

提示：在我们使用 mxmlc 编译指令的时候，如果不使用任何编译选项，那么 mxmlc 编

译器将自动根据 Flex SDK\frameworks\flex-config.xml 内的内容去编译 Flex 应用程序。通过编译器直接指定的选项将覆盖 flex-config.xml 内的内容。

以下内容是一个合法的外部配置文件的内容。

```
<flex-config xmlns="http://www.adobe.com/2006/flex-config">
<compiler>
    <incremental>true</incremental>
</compiler>
<metadata>
    <title>Example</title>
</metadata>
</flex-config>
```

它是一个 xml 文件，由<flex-config>作为根元素，根元素使用了命名空间。然后使用了子元素 compiler 和 metadata。那么，哪些元素才是外部配置文件中的合法元素，这些元素的顺序又应该如何安排呢？开发者可以键入指令：

```
mxmlc –help list
```

这个指令会输出 mxmlc 编译指令所允许的所有选项。比如-compiler.debug 这个选项。它在配置文件中则必须是 compiler 的子元素。如果某个指令出现[value] [...]这样的情况，那么它代表这个 value 值的标签必须作为子元素出现一次或多次。如果某个指令出现<uri> <manifest>这样的情况，那么它代表 uri 和 manifest 这两个元素必须作为子标签顺序出现在某个指定的父元素下。

compc 的使用方法类似于 mxmlc 的使用方法，只不过它是针对共享组件 RSL，而且它的使用比 mxmlc 简单。

提示：在某些大工程项目中，开发组可以采用 Ant 进行项目管理。使用 Ant 或者 Adobe Lab 的 Flex Ant 可以有效地组织 Flex 应用程序编写的项目工程。

第3章 MXML

MXML 是 XML 语言的一种扩展。由于 XML 比 HTML 语言更能结构化地表达数据内容，因此 MXML 语言能更好地描述对象属性和对象间的层次关系。本书假设读者已经熟练掌握 XML 的语法。

绝大多数 MXML 标签都能在 Flex API 中找到其对应的类别。如 MXML 中的<mx:Application>标签，对应于 API 中的 mx.core.Application。

MXML 格式的文件最终被编译器编译成 SWF 文件的时候，编译器会自动把 MXML 元素所对应的标签转换成对应的 ActionScript 类别，而元素标签所具有的属性被设置在对应类别的实例上。这有点类似于 JSP 文件在运行时会被编译成 Servlet。

在 Flex 开发中，MXML 语言的主要作用是用来设置可视化的界面。例如，设置一个能垂直摆放控件的容器，并在容器内添加多个可视化控件。开发者除了可以使用 Flex 内置的 MXML 标签外，还可以根据自己的需求去扩展定制新的标签。

3.1 MXML 文件的命名规则

MXML 文件的命名只能以字母或下划线“_”开始，在这之后只能由字母、数字和下划线组成。MXML 文件名的后缀必须以小写的.mxml 来结束。

在同一路径下 MXML 的文件名不能采用以下的命名方式：

MXML 的文件名不能与 ActionScript 的类文件名重名。如同一路径下不能存在名为 Test.mxml 和 Test.as（ActionScript 编写的类文件是以.as 作为后缀名）的文件。编译时，编译器会提示错误。

MXML 的文件名不能与 MXML 内某个组件的 id 值重名。如 Test.mxml 有标签<mx:Button id="Test">，由于 Button 标签的 id 值为 Test，而 MXML 的文件名也为 Test，因此编译时会提示错误。

不要把 MXML 文件命名为 application.mxml，不要让 MXML 的文件名与 MXML 内某个元素的标签名相同。如 MXML 内拥有<mx:Button>元素，就不要用 Button.mxml 去命名 MXML 文件，这样的命名会导致编译后的程序在运行时发生意想不到的结果。

3.2 第一个 Flex 应用程序：你好 Flex

虽然 Flex Builder 这样的 IDE 可以方便开发者快速地开发 Flex 应用程序，不过还是建议大家在刚开始学习 Flex 的时候使用文本编辑器。这样能透彻地了解 Flex 的更多细节。

实例 3-1 你好，Flex（Chapter3/HelloFlex）[⊖]

新建一个名为 Test.mxml 的文本文件，并使用文本编辑器打开它。在空白文本内插入以

⊖ 括号中的内容表示它在光盘中的位置。