

<<Flex 4实战>>

图书基本信息

书名：<<Flex 4实战>>

13位ISBN编号：9787302270416

10位ISBN编号：7302270414

出版时间：2012-1

出版时间：清华大学出版社

作者：[美]Tariq Ahmed,[美]Dan Orlando,[美]John C. Bland II,[美]Joel Hooks

页数：567

译者：郭俊凤

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<Flex 4实战>>

### 内容概要

本书是一本全面的指南，为Web设计人员和开发人员提供了Flex的详细介绍。本书从基础讲起，介绍了有关表单和数据的知识，然后逐渐过渡到核心概念，如导航、拖放和事件。即使读者刚刚接触Flex，在阅读本书之后也可以使用新增的Spark组件、数据服务、图表设计、特殊效果等，使自己的应用程序广受欢迎。本书读者需要具备基本的开发技能，但是不要求以前使用过Flex。

## <<Flex 4实战>>

### 作者简介

作者：(美国)艾哈迈德 (Tariq Ahmed) (美国)Dan Orlando (美国)Joel Hooks 等 译者：郭俊凤 Tariq Ahmed 是RIA工程师和F1cX社区的积极拥护者。

Dan Orlando是RIA架构师，擅长F1ex和AIR。

Joel Hooks是Flash平台开发人员利ActionScript专家。

## &lt;&lt;Flex 4实战&gt;&gt;

## 书籍目录

## 第 部分 应用程序基础知识

## 第1章 认识Flex

## 1.1 Web应用程序为何大行其道

## 1.2 RIA解决方案

## 1.2.1 他们什么都想要

## 1.2.2 RIA的解决之道

## 1.2.3 RIA的工作原理

## 1.3 RIA的竞争技术

## 1.3.1 Adobe公司的Flex

## 1.3.2 Microsoft公司的Silverlight

## 1.3.3 Sun公司的JavaFX

## 1.3.4 AJAX——终点站

## 1.4 熟悉Flex

## 1.4.1 利用Adobe Flash的优势

## 1.4.2 Flex与JavaScript可以一起使用

## 1.4.3 Flex技术体系

## 1.5 Flex的工作原理

## 1.5.1 Flex语言

## 1.5.2 事件驱动的环境

## 1.5.3 Flex的局限

## 1.6 Flex 的新功能

## 1.7 小结

## 第2章 从零开始

## 2.1 经济的Flex开发方式

## 2.1.1 建立编译环境

## 2.1.2 建立编辑环境

## 2.1.3 第三步(但愿读者还感兴趣)

## 2.2 使用Flash Builder做真正的开发

## 2.2.1 产品和价格表

## 2.2.2 获得Flash Builder

## 2.3 探索Flash Builder

## 2.4 视图与视角

## 2.4.1 预定义的视角

## 2.4.2 切换视角

## 2.4.3 自定义视角

## 2.5 第一个项目——HelloWorld

## 2.5.1 创建项目

## 2.5.2 输入代码

## 2.5.3 编译和运行

## 2.5.4 查看实际效果

## 2.6 使用设计模式

## 2.7 内置的参考和API文档

## 2.7.1 面向对象语言及其API

## 2.7.2 访问API参考

## 2.7.3 研读API参考

## &lt;&lt;Flex 4实战&gt;&gt;

- 2.8 MXML与ActionScript速写
  - 2.8.1 MXML的结构
  - 2.8.2 MXML与ActionScript的关系
  - 2.8.3 事件由ActionScript处理
- 2.9 小结
- 第3章 使用ActionScript
  - 3.1 注释
    - 3.1.1 行内注释
    - 3.1.2 多行注释
  - 3.2 变量
    - 3.2.1 变量名
    - 3.2.2 严格数据类型
    - 3.2.3 静态类型检查与动态类型检查
    - 3.2.4 顶级类
    - 3.2.5 特殊数据类型
  - 3.3 循环
    - 3.3.1 for(初始值;条件;递增)
    - 3.3.2 for(属性名in数组/对象)
    - 3.3.3 for each(项in数组/对象)
    - 3.3.4 while(条件)
    - 3.3.5 do while(条件)
  - 3.4 条件语句(if和switch)
    - 3.4.1 if...else
    - 3.4.2 switch语句
  - 3.5 数组
    - 3.5.1 索引数组
    - 3.5.2 关联数组
  - 3.6 ActionScript特色语法
    - 3.6.1 花括号
    - 3.6.2 逻辑操作符及快捷方式
  - 3.7 函数、类和包简介
    - 3.7.1 创建函数
    - 3.7.2 将ActionScript代码保存在单独的文件中
  - 3.8 简单的数据绑定
    - 3.8.1 不使用绑定
    - 3.8.2 添加绑定
    - 3.8.3 绑定标签
    - 3.8.4 使ActionScript变量可被绑定
  - 3.9 小结
- 第4章 布局和容器
  - 4.1 Spark与Halo(MX)
  - 4.2 绝对布局
  - 4.3 基于约束的布局
    - 4.3.1 基本约束
    - 4.3.2 增强的约束
  - 4.4 自动布局
    - 4.4.1 使用布局类

## &lt;&lt;Flex 4实战&gt;&gt;

- 4.4.2 需要空间
- 4.5 可变大小和固定大小
  - 4.5.1 可变大小
  - 4.5.2 固定大小
- 4.6 容器
  - 4.6.1 Application容器
  - 4.6.2 Canvas容器
  - 4.6.3 基于Group的容器和SkinnableContainer
  - 4.6.4 Panel容器
  - 4.6.5 ApplicationControlBar容器
  - 4.6.6 DataGroup和SkinnableDataContainer
  - 4.6.7 DividedBox、HDividedBox和VDividedBox容器
  - 4.6.8 Form容器
  - 4.6.9 Grid容器
- 4.7 小结
- 第5章 显示表单与捕获用户输入
  - 5.1 id特性
  - 5.2 Flex控件分类
    - 5.2.1 Text控件
    - 5.2.2 Date控件
    - 5.2.3 Numeric控件
    - 5.2.4 Flex的按钮组件
    - 5.2.5 选单控件
  - 5.3 访问控件的值
    - 5.3.1 向函数传递值
    - 5.3.2 向函数传递事件
    - 5.3.3 直接访问属性
    - 5.3.4 应该选择哪种方式
  - 5.4 小结
- 第6章 验证用户输入
  - 6.1 验证简介
  - 6.2 内置验证器
    - 6.2.1 Validator
    - 6.2.2 StringValidator
    - 6.2.3 NumberValidator
    - 6.2.4 DateValidator
    - 6.2.5 EmailValidator
    - 6.2.6 CreditCardValidator
    - 6.2.7 CurrencyValidator
    - 6.2.8 PhoneNumberValidator
    - 6.2.9 RegExpValidator
    - 6.2.10 SocialSecurityValidator
    - 6.2.11 ZipCodeValidator
  - 6.3 实时验证
  - 6.4 提交值验证
  - 6.5 通过性验证

## &lt;&lt;Flex 4实战&gt;&gt;

- 6.6 脚本式验证
- 6.7 验证技巧
  - 6.7.1 验证器是否总是检查所有条件
  - 6.7.2 控制由什么来触发验证操作
- 6.8 小结
- 第7章 格式化数据
  - 7.1 内置格式化程序
    - 7.1.1 Formatter
    - 7.1.2 NumberFormatter
    - 7.1.3 CurrencyFormatter
    - 7.1.4 DateFormatter
    - 7.1.5 PhoneFormatter
    - 7.1.6 ZipCodeFormatter
    - 7.1.7 SwitchSymbolFormatter
  - 7.2 实时格式化
  - 7.3 脚本式格式化
    - 7.3.1 结合使用函数与格式化程序组件
    - 7.3.2 结合使用函数与格式化程序类
  - 7.4 处理格式化错误
  - 7.5 小结
- 第8章 MX DataGrid、列表和树
  - 8.1 基于列表的一系列MX组件
    - 8.1.1 ListBase和AdvancedList  
Base的属性
    - 8.1.2 MX ListBase事件
  - 8.2 集合与dataProvider
    - 8.2.1 填充dataProvider
    - 8.2.2 集合的类型
    - 8.2.3 集合的使用者
  - 8.3 集合的初始化
  - 8.4 集合的填充
    - 8.4.1 List组件
    - 8.4.2 HorizontalList组件
    - 8.4.3 TileList组件
    - 8.4.4 DataGrid组件
    - 8.4.5 Tree组件
  - 8.5 与基于List的MX组件交互
    - 8.5.1 列表事件
    - 8.5.2 向函数传递事件
    - 8.5.3 向函数中传递数据
    - 8.5.4 直接访问被选定的行
    - 8.5.5 绑定到被选定的行
  - 8.6 小结
- 第9章 使用Spark列表控件
  - 9.1 基于SparkList的一系列组件
    - 9.1.1 使用名称空间标识合适的组件
    - 9.1.2 使用基于List的Spark控件的项渲染器

## &lt;&lt;Flex 4实战&gt;&gt;

## 9.2 基于List的Spark控件

## 9.2.1 ButtonBar控件

## 9.2.2 SparkList控件

## 9.2.3 DropDownList控件

## 9.3 与基于List的Spark组件的交互

## 9.3.1 选择项目时默认分发事件

## 9.3.2IndexChangedEvent对象

## 9.4 理解Flex 基于List的组件架构

## 9.4.1 类层次结构

## 9.4.2 新的基于List的Spark组件类

## 9.5 在Spark架构中构建基于List的自定义组件

## 9.5.1 理解Group和

## SkinnableContainer类

## 9.5.2 构建自定义SparkList组件

## 9.6 小结

## 第10章 自定义列表

## 10.1 自定义数据显示

## 10.1.1 labelField属性

## 10.1.2 标签函数

## 10.1.3 标签函数的类型

## 10.1.4 使用多列标签函数

## 10.1.5 标签函数的用途

## 10.1.6 从全局考虑

## 10.2 项渲染器

## 10.2.1 Spark MXML项渲染器

## 10.2.2 在Spark中对MX组件使用MXML项渲染器

## 10.2.3 创建行内项渲染器

## 10.2.4 使用混入项渲染器

## 10.3 项编辑器

## 10.3.1 启用项编辑功能

## 10.3.2 创建项编辑器

## 10.3.3 项编辑事件

## 10.3.4 合二为一：renderersEditor

## 10.4 高级项渲染器

## 10.4.1

## AdvancedDataGridRendererProvider

## 10.4.2 引用列

## 10.4.3 跨列使用

## 10.4.4 实现通行显示

## 10.5 滤镜函数

## 10.6 小结

## 第 部分 应用程序流与结构

## 第11章 事件

## 11.1 事件系统

## 11.1.1 事件系统——好莱坞原则

## 11.1.2 事件发送系统

## 11.2 发送和接收事件



## &lt;&lt;Flex 4实战&gt;&gt;

- 11.2.1 使用ActionScript添加事件监听器
- 11.2.2 绑定事件
- 11.2.3 移除事件监听器
- 11.3 自定义事件
  - 11.3.1 分派自定义事件类型
  - 11.3.2 创建自定义事件
  - 11.3.3 向自定义分派器中添加事件元数据
  - 11.3.4 停止事件传播
- 11.4 小结
- 第12章 应用程序导航
  - 12.1 准备菜单数据
    - 12.1.1 嵌套的数组
    - 12.1.2 嵌套的数组集合
    - 12.1.3 模型
    - 12.1.4 XML组件和类
    - 12.1.5 XMMLList组件
    - 12.1.6 XMMLListCollection组件及类
  - 12.2 使用菜单
    - 12.2.1 创建菜单
    - 12.2.2 定位菜单
    - 12.2.3 自定义菜单项
    - 12.2.4 与菜单交互
  - 12.3 使用菜单栏
    - 12.3.1 创建菜单栏
    - 12.3.2 定位菜单栏
    - 12.3.3 自定义菜单栏中的项
    - 12.3.4 处理用户与菜单栏的交互
  - 12.4 使用ViewStack
    - 12.4.1 创建ViewStack
    - 12.4.2 为ViewStack添加导航功能
    - 12.4.3 处理用户与ViewStack的交互
  - 12.5 TabNavigator
    - 12.5.1 创建TabNavigator
    - 12.5.2 处理用户与TabNavigator的交互
  - 12.6 Accordion
    - 12.6.1 创建Accordion
    - 12.6.2 填充Accordion
    - 12.6.3 处理用户与Accordion的交互
  - 12.7 小结
- 第13章 弹出式窗口简介
  - 13.1 创建首个弹出式窗口
    - 13.1.1 首先创建标题窗口
    - 13.1.2 使用PopUpManager打开窗口
    - 13.1.3 关闭弹出式窗口
  - 13.2 控制窗口位置
    - 13.2.1 使用centerPopUp()方法
    - 13.2.2 计算窗口的位置

## &lt;&lt;Flex 4实战&gt;&gt;

## 13.3 弹出式窗口与数据的结合

## 13.3.1 从弹出式窗口中取得数据

## 13.3.2 发送包含数据的自定义事件

## 13.3.3 取出数据

## 13.3.4 向窗口中发送数据

## 13.4 使用警告窗口

## 13.4.1 创建简单的警告窗口

## 13.4.2 挖掘警告窗口的潜力

## 13.4.3 高级警告窗口

## 13.4.4 修饰警告窗口

## 13.5 小结

## 第14章 实现视图状态

## 14.1 理解视图状态

## 14.2 Flex中的视图状态

## 14.2.1 使用属性

## 14.2.2 使用事件处理程序

## 14.2.3 使用状态组

## 14.2.4 添加和移除组件

## 14.2.5 更改组件的父组件

## 14.2.6 状态事件

## 14.3 综合运用

## 14.4 小结

## 第15章 使用数据服务

## 15.1 访问服务器端数据

## 15.1.1 使用HTTPService对象

## 15.1.2 通过WebService组件使用Web服务

## 15.2 活动消息格式

## 15.2.1 开源AMF

## 15.2.2 AMF与PHP

## 15.2.3 AMF和ColdFusion

## 15.2.4 BlazeDS

## 15.2.5 LiveCycle Data Service

## 15.2.6 其他技术

## 15.3 使用Flash Builder构建以数据为中心的应用程序

## 15.3.1 建立正确的环境

## 15.3.2 与服务器建立连接

## 15.4 使用ColdFusion进行以数据为中心的Flex

## 15.5 使用Java EE和BlazeDS进行以数据为中心的Flex开发

## 15.6 把模型绑定到视图上

## 15.6.1 拖放式数据绑定

## 15.6.2 生成一个大纲-细节表单

## 15.6.3 检查Flash Builder代码

## 15.7 小结

## 第16章 对象和类

## 16.1 用5分钟的时间理解OO理论

## 16.1.1 对象和类的关系

## 16.1.2 对象拥有属性和方法

## &lt;&lt;Flex 4实战&gt;&gt;

- 16.1.3 继承
- 16.1.4 封装和耦合
- 16.2 操作对象
  - 16.2.1 进一步了解对象
  - 16.2.2 对象的方法
  - 16.2.3 方法的参数
  - 16.2.4 方法返回信息
- 16.3 创建类
  - 16.3.1 创建类文件
  - 16.3.2 指定包
  - 16.3.3 类修饰符
  - 16.3.4 超类：扩展类
  - 16.3.5 接口
  - 16.3.6 分析类
- 16.4 操作属性
  - 16.4.1 添加属性
  - 16.4.2 添加getter/setter方法
- 16.5 创建类的方法
- 16.6 小结
- 第17章 自定义组件
  - 17.1 理解Flex 组件
    - 17.1.1 Spark组件架构
    - 17.1.2 自定义组件的种类
  - 17.2 创建简单的自定义组件
    - 17.2.1 构建简单的ComboBox
    - 17.2.2 简单的Spark组件
  - 17.3 使用Spark Skin对象进行换肤
    - 17.3.1 使用元数据绑定组件皮肤
    - 17.3.2 自定义组件视图状态
    - 17.3.3 定义皮肤部件
    - 17.3.4 声明host
  - 17.4 复合组件
    - 17.4.1 比较Halo与Spark
    - 17.4.2 Spark布局类
    - 17.4.3 创建MXML复合组件
  - 17.5 创建高级Flex 组件
    - 17.5.1 使用ActionScript构建州选择器
    - 17.5.2 确定何时重写
  - 17.6 组件的通信
    - 17.6.1 使用getter和setter作为属性代理
    - 17.6.2 将变量绑定到组件属性
    - 17.6.3 使用事件传递数据
  - 17.7 小结
- 第18章 创建可重用组件
  - 18.1 利用可重用性
    - 18.1.1 Reveal组件的背景
    - 18.1.2 Spark可重用性的理论和概念

## &lt;&lt;Flex 4实战&gt;&gt;

- 18.1.3 创建Reveal组件
- 18.1.4 为第一个Reveal实现添加皮肤
- 18.2 Reveal的新外观
- 18.3 运行时共享库
  - 18.3.1 理解SWC文件
  - 18.3.2 RSL的类型
  - 18.3.3 在Flex应用程序中使用RSL
- 18.4 小结
- 第19章 架构设计模式
  - 19.1 Flex驱动的设计模式
    - 19.1.1 模型-视图-控制器模式
    - 19.1.2 开发自己的架构
  - 19.2 微架构简介
    - 19.2.1 微架构的简单定义
    - 19.2.2 使用微架构的原因
    - 19.2.3 第一代微架构
    - 19.2.4 第二代微架构
    - 19.2.5 控制反转和依赖注入
  - 19.3 使用Robotlegs框架
    - 19.3.1 使用Robotlegs注入依赖
    - 19.3.2 使用Robotlegs映射实用工具来配置依赖注入
  - 19.4 使用Robotlegs MVCS创建应用程序
    - 19.4.1 设置Robotlegs项目
    - 19.4.2 使用Context类引导应用程序
    - 19.4.3 对视图使用中介
    - 19.4.4 使用Robotlegs命令进行控制
    - 19.4.5 服务是通往外界的大门
    - 19.4.6 使用模型来管理数据和状态
  - 19.5 小结
- 第 部分 高级应用
- 第20章 自定义体验
  - 20.1 用户体验设计的原则
    - 20.1.1 围绕用户故事进行构建
    - 20.1.2 考虑上下文
    - 20.1.3 VIBE模型
  - 20.2 视觉吸引力
    - 20.2.1 创建并使用主题
    - 20.2.2 使用CSS设置Flex 应用程序的样式
  - 20.3 交互式体验
    - 20.3.1 使用FXG和Flash Catalyst CS5进行声明性设计
    - 20.3.2 使用效果增强用户体验
  - 20.4 业务优化
    - 20.4.1 使用最佳实践改进体验
    - 20.4.2 通过单元测试改进用户体验
    - 20.4.3 概要分析Flex 中的应用程序

## &lt;&lt;Flex 4实战&gt;&gt;

## 20.5 可扩展性

## 20.5.1 可扩展性与用户体验的间接关系

## 20.5.2 可扩展性与用户体验的直接关系

## 20.5.3 编写干净的代码以实现易用性

## 20.6 小结

## 第21章 使用效果

## 21.1 效果的含义

## 21.1.1 可用的效果

## 21.1.2 合成效果

## 21.2 使用效果

## 21.2.1 创建效果的方式

## 21.2.2 由事件触发的效果

## 21.2.3 以编程方式应用效果

## 21.2.4 使用状态过渡来触发效果

## 21.3 创建合成效果

## 21.3.1 串行效果

## 21.3.2 并行效果

## 21.3.3 组合合成效果

## 21.4 探索效果

## 21.4.1 Animate效果

## 21.4.2 使用AnimateFilter设置滤镜的动画

## 21.4.3

## 使用AnimateTransitionShader设置像素着色器动画

## 21.4.4 自定义效果缓动

## 21.4.5 在播放动画时显示文本

## 21.4.6 创建声音效果

## 21.5 小结

## 第22章 拖放

## 22.1 拖放过程

## 22.2 在Flex组件中实现拖放

## 22.2.1 内置支持拖放的组件

## 22.2.2 在列表中启用拖放

## 22.2.3 移动与复制

## 22.2.4 使用拖放实现用户控制的排序

## 22.2.5 多项拖放

## 22.2.6 双向拖放

## 22.3 使用DragManager

## 22.3.1 DragManager的属性和方法

## 22.3.2 接受或拒绝放置

## 22.3.3 应用自定义放置方式

## 22.4 为非列表组件添加拖放行为

## 22.4.1 设置示例

## 22.4.2 启动拖动

## 22.4.3 使用自定义的拖动代理

## 22.4.4 处理放置

## 22.5 自定义拖放体验

## 22.5.1 修改拖动代理图标

## &lt;&lt;Flex 4实战&gt;&gt;

- 22.5.2 用于拖放的List组件换肤
- 22.5.3 在Spark和Halo之间混合拖放
- 22.6 小结
- 第23章 探索Flex图表
- 23.1 图表简介
  - 23.1.1 图表的构成
  - 23.1.2 图表类型简介
- 23.2 准备系列和数据
- 23.3 创建图表
  - 23.3.1 调用图表
  - 23.3.2 添加图例
  - 23.3.3 改变图表类型
  - 23.3.4 过滤图表数据
- 23.4 堆叠图表
- 23.5 探索图表类型
  - 23.5.1 面积图
  - 23.5.2 条形图和柱状图
  - 23.5.3 折线图
  - 23.5.4 气泡图
  - 23.5.5 烛台图和HLOC图
  - 23.5.6 饼图
  - 23.5.7 散点图
- 23.6 自定义图表
  - 23.6.1 系列笔触
  - 23.6.2 系列填充
- 23.7 小结
- 第24章 调试与测试
- 24.1 调试
  - 24.1.1 安装Flash Debug Player
  - 24.1.2 使用trace()函数
  - 24.1.3 使用Flash Builder  
调试器
  - 24.1.4 监视网络活动
- 24.2 Flex概要分析器
- 24.3 使用自动化工具测试  
应用程序
  - 24.3.1 单元测试
  - 24.3.2 功能测试
- 24.4 小结
- 第25章 项目收尾
- 25.1 自定义HTML包装器
  - 25.1.1 包装器文件
  - 25.1.2 HTML模板
  - 25.1.3 SWFObject
- 25.2 部署
  - 25.2.1 创建生产版本
  - 25.2.2 放置客户端文件

## <<Flex 4实战>>

25.2.3 放置服务器端文件

25.2.4 测试应用程序

25.3 小结

<<Flex 4实战>>

章节摘录

版权页：插图：



## <<Flex 4实战>>

### 媒体关注与评论

所有Flex4开发人员的手边必备参考手册。

——John Griffin全面细致，没有遗漏之处。

——Peter Pavlovich，Kronos Inc.对于新手和有经验的Flex开发人员而言都是一本极佳的指南。

——Kevin Schmidt，Adobe Systems，Inc.内容具有很强的指导意义，闪烁着智慧的光芒，并且十分切题。

——Zareen Zaffar，Amcom彻底揭开了构建丰富用户界面的神秘面纱。

——Rick Wagner，Acxiom Corp.学习完本书就可以在工作中得心应手地使用Flex4。

——Rick Evans，SAS

## <<Flex 4实战>>

### 编辑推荐

《Flex 4实战》是由清华大学出版社出版的。

<<Flex 4实战>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>