

<<GWT揭秘>>

图书基本信息

书名：<<GWT揭秘>>

13位ISBN编号：9787111294016

10位ISBN编号：7111294017

出版时间：2010-1

出版时间：机械工业出版社

作者：徐彬

页数：316

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>

<<GWT揭秘>>

前言

1995年底，Netscape发布了Navigator 2.0版本，它加入了一个重要的特性：可以在页面中嵌入JavaScript语言。

在Navigator 2.0发布前，Netscape将这种新语言命名为LiveScript，但最终发布时却使用了JavaScript这个名字，因为Netscape想借助Java这门当时最炙手可热的语言迅速走红。

Navigator 2.0发布后，Web程序员们立刻喜欢上了这门新语言。

使用JavaScript能够在网页中实现用户输入信息的校验工作，比如检查E-Mail的地址格式是否正确，核对密码和校验密码是否一致等。

在当时那个网络带宽只有28.8kBps的年代，能在浏览器端验证这些信息，无疑是Web开发界的一次重大变革。

在JavaScript出现之前，如果用户在浏览器端输入一个错误信息，往往需要等待10多秒钟才能收到从服务器端发回的反馈，这对用户来说简直是不可接受的。

JavaScript出现以后，很好地解决了这个问题，Web应用的用户体验有了一个质的飞跃，嵌有JavaScript的Web应用备受用户青睐。

1996年8月，微软发布了Internet Explorer 3.0。

那时，微软在浏览器市场上还只是个跟随者。

为了能兼容大量嵌入JavaScript的网页，Internet Explorer 3.0也提供了对JavaScript的支持。

微软将他们的语言命名为JScript，以避免版权和商标上的麻烦。

微软为JScript加入了一些新的特性，但这导致JavaScript和JScript出现了兼容性问题。

为避免这一问题继续扩大，Netscape把JavaScript语言规范提交给ECMA（European Computer Manufacturing Association，欧洲计算机制造协会），申请制定JavaScript语言标准。

1997年年中，ECMA正式制定了ECMA.262标准。

从此，JavaScript又被称为ECMAScript。

随着PC计算能力的增强，JavaScript不再满足于完成验证用户输入这样的简单工作。

2005年，AJAX进入程序员的视野。

AJAX把页面的逻辑全部搬到了浏览器端，仅把数据留在服务器上。

网页不再需要刷新，用户能够在浏览器中享受和桌面应用程序一样的流畅体验。

然而，事物总是具有两面性。

JavaScript作为一种动态语言，在验证用户输入和实现一些网页特效方面有得天独厚的优势。

但是，对于开发需要多人合作的大型项目来说，它却存在着以下问题：第一，动态语言的代码编写具有随意性和自由性。

对于一个需要长期维护、多人参与的大型项目开发来说，JavaScript在代码维护、团队合作、编码规约、技术交流上存在着诸多不利因素。

第二，动态语言的状态属性需要在运行时才能确定，IDE无法完整地实现智能提示。

<<GWT揭秘>>

内容概要

国内第一本基于GWT 2.0的经典著作，4大专业社区联袂推荐，权威性毋庸置疑！

《GWT揭秘》内容全面，不仅详细介绍了GWT的主要模块和控件、GWT与JavaScript对象的交互、在GWT中使用XML、开发自定义GWT控件、GWT-RPC和Ext GWT等必备的基础知识，而且还深入讲解了GWT与Flex整合、图片缓存、本地化、GWT动画、历史管理、延时 / 等待 / 分片执行等高级知识：《GWT揭秘》注重实战，所有知识点都配有精心设计的范例，尤为值得一提的是，还以迭代的方式重现了经典的俄罗斯方块游戏和一个完整的报销审批系统的开发全过程，既可以以它们为范例进行实战演练，又可以将它们直接应用到实际开发中去。

<<GWT揭秘>>

作者简介

徐彬，资深软件工程师、项目经理和系统架构师，从业10余年，实战经验丰富。
精通C++（CSDN论坛C++版专家）以及GWT等AJAX技术，擅长C#、Java、JavaScript开发，对Oracle也有深入研究。
多年来一直从事国土和规划领域的政府单位的信息化建设，曾负责过西安、南京、乌鲁木齐、杭州

<<GWT揭秘>>

书籍目录

前言	第一部分 准备篇	第1章 准备开发环境	1.1 安装JDK	1.2 安装Eclipse	1.3 安装GWT 1.7
的Eclipse插件	1.4 安装GWT2.0的SDK以及Eclipse插件	1.5 辅助工具	1.5.1 Internet Explorer Developer Toolbar	1.5.2 Firebug	1.6 小结
第2章 Hello GWT	2.1 在Eclipse中创建项目	2.2 运行调试项目	2.3 JUnit测试	2.4 发布网站	2.5 ANT发布
2.6 小结	第二部分 基础篇	第3章 GWT模块	3.1 GWT模块详解	3.2 模块入口点	3.3 模块继承
3.4 附加CSS和Java Script	3.5 路径配置	3.6 延迟绑定	3.7 发布JAR	3.8 深入研究	3.9 小结
第4章 JSNI	4.1 JSNI和JavaScript	4.2 类型映射	4.3 Java Script Object	4.4 Js Array	4.5 JSNI中的特殊变量
4.6 在JSNI中访问GWT函数	4.7 异常处理	4.8 外部JavaScript访问GWT函数	4.9 整	合jQuery	4.10 在GWT中使用正则表达式
4.11 小结	第5章 在GWT中使用XML	5.1 XML简介	5.2 XML操作	5.2.1 XML文档	5.2.2 XML节点
5.2.3 XML元素节点	5.2.4 XML 文本节点	5.3 在GWT中使用X Path	5.4 小结	第6章 GEWT控件详解	6.1 界面对象
6.2 Widget	6.3 面板	6.4 控件生命周期	6.5 控件事件管理	6.5.1 注册控件事件处理句柄	6.5.2 阻止控件默认行为
6.5.3 阻止事件冒泡	6.5.4 on Browser Event	6.6 GWT全局类	6.6.1 DOM类	6.6.2 GWT类	6.6.3 Window类
6.7 小结	第7章 使用GWT控件	7.1 简单控件	7.1.1 包装已有控件	7.1.2 图片超链接	7.1.3 图片按钮
7.1.4 复选框	7.1.5 单选框	7.1.6 文件上传	7.1.7 Date Picker	7.2 列表和菜单	7.2.1 列表
7.2.2 带建议的输入框	7.2.3 树	7.2.4 菜单	7.3 面板	7.3.1 装饰面板	7.3.2 流
7.3.3 水平布局面板和垂直布局面板	7.3.4 绝对布局面板	7.3.5 Tab面板	7.4 表格	7.4.1 简单表格	7.4.2 复杂表格
7.5 使用与自定义GWT控件	7.5.1 Combo Input控件结构	7.5.2 Data Item	7.5.3 List Box	7.5.4 Selected事件	7.5.5 Trigger Input
7.5.6 Combo Box	7.6 小结	第8章 GWT.RPC	8.1 GWT中的XMLHTTP Request	8.1.1 HTTP Request Imp1	8.1.2 Request Builder
8.1.3 请求动态数据	8.1.4 XML	8.2 JSON	8.2.1 JSON Value	8.2.2 JSON Value的派生类	8.2.3 完整的JSON例子
8.3 GWT.RPC	8.3.1 RPC远程接口	8.3.2 服务器实现	8.3.3 异步接口文件	8.3.4 在程序中使用GWT.RPC	8.3.5 整合Hibernate
8.4 小结	第9章 Ext GWT	9.1 Ext GWT简介及安装	9.2 Window	9.2.1 窗口布局	9.2.2 Viewport
9.2.3 Dialog	9.2.4 窗口的显示与关闭	9.3 数据管理	9.4 Grid	9.5 树控件	9.5.1 图标显示
9.5.2 节点选择事件	9.6 皮肤	9.7 小结	第三部分 实战篇	第10章 俄罗斯方块游戏	10.1 游戏设计
10.1.1 游戏简介	10.1.2 游戏规则	10.1.3 设计思路	10.2 编码实现	10.2.1 游戏场地	10.2.2 方块
10.2.3 游戏控制器	10.3 界面实现及操作完善	10.3.1 候选面板	10.3.2 计分与等级	10.3.3 增加操作便利性	10.3.4 界面美化
10.4 联机对战	10.4.1 原理	10.4.2 等待游戏开始	10.4.3 交换数据	10.4.4 胜负判定	10.4.5 障碍行
10.5 小结	第11章 费用申请审批流程	11.1 流程设计	11.1.1 流程简介	11.1.2 工作流引擎	11.1.3 表单
11.2 工作流引擎实现	11.2.1 环境配置	11.2.2 业务规则定义	11.2.3 jBPM数据库初始化	11.3 表单实现	11.3.1 HTJML面板
11.3.2 表单对话框	11.3.3 权限控制	11.3.4 类型校验	11.4 审批流程的界面实现	11.4.1 Async Callback Ex	11.4.2 Outlook面板
11.4.3 工作列表	11.4.4 整体布局	11.4.5 用户登录	11.4.6 全局事件管理	11.4.7 遮蔽效果	11.4.8 实现BizAction
11.5 业务流转	11.5.1 申请表	11.5.2 审批表	11.6 小结	第四部分 高级篇	第12章 GWT与Flex整合
12.1 Flex的优势	12.2 查看流程图的Flex	12.3 加入网页	12.4 与Flex交互	12.5 小结	第13章 图片缓存
13.1 CSS Sprite简介	13.2 GWT的实现	13.3 后端生成了什么	13.4 小结	第14章 本地化	14.1 GWT如何识别你的语言
14.2 本地化配置	14.3 日期格式化	14.4 字符串动态本地化	14.5 小结	第15章 其他高级功能	15.1 动画
15.2 历史管理	15.3 延时执行	15.4 等待执行和分片执行	15.5 代码分割	15.5.1 SOYC	15.5.2 初始化序列
15.5.3 编程模式	15.5.4 预加载	15.6 Ui Binder	15.6.1 HTML绑定	15.6.2 控件绑定	15.6.3 CSS绑定

15.6.4 事件管理

15.6.5 资源打包

15.6.6 创建控件

15.7 小结

章节摘录

在HTTP规范中，URL请求的长度不能大于1024个字节。
所以使用GET方式只能传递少量的参数。
而使用POST方式访问服务器，在发送URL地址后，还能再给服务器发送独立的数据实体e。
如果要传送大量数据给服务器，那就只能用POST方式。
最后一种情况是，向某些Web服务器上的请求特定类型文件时，只能使用GET方式去获取。
在代码清单8-2的第20行，通过setCallback（）函数，将RequestCallback对象设置给RequestBuilder对象。
如果发生错误，会回调onError（）数；请求完成，则会回调onResponseReceived（）函数。
触发onResponseReceived（）回调函数，可能是请求成功了，也可能是服务器发生了异常，无法完成请求。
程序第28行通过response.getStatusCode（）获取请求完成的状态值。
经常会遇到的状态值一般是。
□200请求成功 □303重定向 □400请求错误 □401未授权 □403禁止访问 □404文件未找到
□500服务器错误 把程序第19行的data.txt改为data1.txt，然后在程序第28行将response.getStatusCode（）打印出来，这时就会发现response.getStatusCode（）返回的是404，表示在服务器上找不到data1.txt文件。
最后在程序第35行，调用request.send（）发送请求。
request.send（）函数可能会触发RequestException异常，这里需要用try...catch...将其包起来。
现在就可以运行程序了，运行结果与代码清单8-1运行结果一样。

<<GWT揭秘>>

编辑推荐

《GWT揭秘》RIA带来了全新的Web用户体验，同时也简化了传统的Web开发模式。

以JavaScript为基础的AJAX技术与现有HTML技术一脉相承，便于与第三方技术相整合，开发效率高，技术起点低，是RIA领域中当之无愧的王者。

然而，AJAX并非没有缺陷。

对于开发中小型项目来说，JavaScript有易用性、灵活性等优势，但是对于开发如何在GWT中使用× Path？

- 如何根据实际项目的需要开发出有特色的自定义GWT控件？

如何使用GWT-RPC的方式与服务器交互？

它与使用UrlEncode、XML、JSON等方式与服务器交互有何不同？

GWT-RPC如何与Hibernate整合？

- 没有美术功底的开发者如何使用EXT GWT开发出漂亮的用户界面？

为何EXT GWT能完全利用GWT的编译优势？大型项目来说，它也存在着不少问题：代码编写具有随意性，不便于维护和团队合作；IDE无法完整地实现智能提示；变量类型和函数参数等的错误在运行时才会暴露出来；浏览器对JavaScript执行错误的提示简单且含糊，不便于调试和纠错；在不同的浏览器上存在着兼容性问题，开发人员必须针对不同的浏览器分别进行开发、调试和测试，工作量大

..."GWT完美而彻底地解决了JavaScript的这些弊端，为RIA开发注入了全新的活力。

如果你也在思考下面这些问题，也许《GWT揭秘》就是你想要的！

- 为何白宫采用GWT技术构建OpenForOJestions网站？

为何JBoss将GWT作为其中间件的核心组成部分？

为何Seam集成GWT？

为何Google Wave完全采用GWT开发？

- 在JSNI中如何轻松地访问GWT函数？

外部JavaScript如何直接访问GWT函数？

- 如何在GWT中高效地使用正则表达式？

如何在GWT中操作XML和JSON格式的数据？

- GWT如何与JavaScript和Flex交互？

GWT如何与jQuery、Ext JS、jBPM相整合？

- 如何利用GWT的动画功能轻松实现各种自定义的动画特效？

- 如何通过GWT 2.0新增的UIBinder使用XML布局定义组件和实现与组件的交互？

- GWT 2.0为何要新增代码分割功能？

如何开发支持本地化的GWT应用？

- 在GWT中如何利用CSS Sprite技术实现图片缓存，从而提高网站的访问速度？

GWT内部运行机制是什么？

如何提高GWT应用的性能？

国内首部基于GWT 2.0的经典著作，4大专业社区一致鼎力推荐！

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>