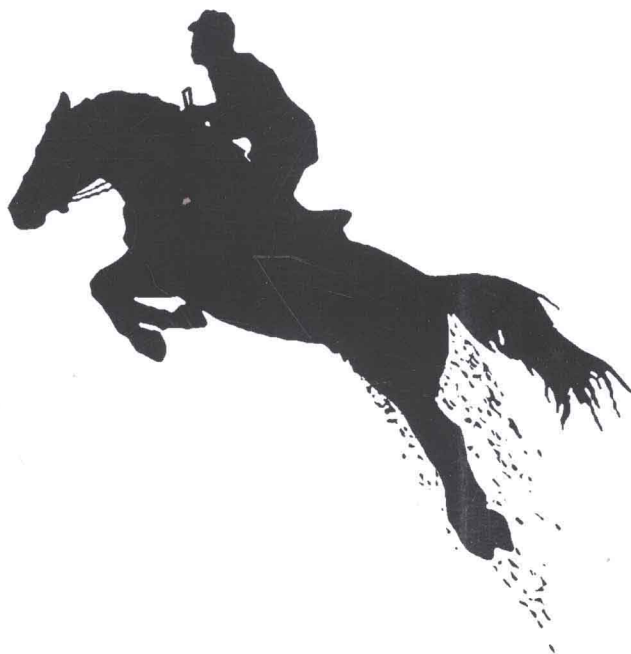


畅销技术图书新版

包含ExtJS 4最新特性，实用案例丰富，掌握ExtJS开发的必备读物



卫军 夏慧军 孟腊春 编著

Web Application Development with ExtJS 4 **第2版**

ExtJS Web 应用程序开发指南



机械工业出版社
China Machine Press



*Web Application Development
with ExtJS 4*

第2版

ExtJS Web 应用程序开发指南

卫军 夏慧军 孟腊春 编著



机械工业出版社
China Machine Press

本书共18章和1个附录,是对第1版的全面升级,增补了大量ExtJS 4.0中的新特性。从基本的ExtJS功能开始讲解RIA Web开发,从而引出用户体验丰富的ExtJS技术。接着通过经典的“Hello World”来快速搭建和配置第一个程序。然后通过4章来详细讲解ExtJS的基本功能,包括:最常用的表单、面板和布局类,常用的工具类与函数,以及ExtJS对事件的响应。本书还结合现今流行的Web框架进行改造,将相关技术融合起来应用。Ajax已经逐渐渗透到Web开发的各个方面了,ExtJS能够与其完美结合。讲解完这些提高内容后,本书还深入介绍了增强型模板,重要的数据模型,以及ExtJS高级组件。考虑到ExtJS与主流服务端框架的整合越来越紧密,本书还特意讲解了Struts和Struts 2框架的整合与配置。最后全面细致地讲解了ExtJS 4.0中新增的主题、图形和图表部分,真正实现完全整合应用ExtJS技术。

本书语言通俗易懂,版式清新,并通过大量的实例去讲解技术。对于广大的Web程序员,本书能帮你找到经典应用的解决方案。对RIA Web应用开发的爱好者,本书可以帮你找到进步的阶梯。本书可以作为Web开发的自学参考书,也可以作为RIA Web开发的培训教材。

封底无防伪标均为盗版

版权所有,侵权必究

本书法律顾问 北京市展达律师事务所

图书在版编目(CIP)数据

ExtJS Web应用程序开发指南 / 卫军, 夏慧军, 孟腊春编著. —2版. —北京: 机械工业出版社, 2011.7

ISBN 978-7-111-35391-1

I. E… II. ①卫… ②夏… ③孟… III. 网页制作工具—程序设计 IV. TP393.092

中国版本图书馆CIP数据核字(2011)第138322号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑: 陈佳媛

北京市荣盛彩色印刷有限公司印刷

2011年7月第2版第1次印刷

186mm×240mm·29.25印张

标准书号: ISBN 978-7-111-35391-1

定价: 69.00元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

客服热线: (010) 88378991; 88361066

购书热线: (010) 68326294; 88379649; 68995259

投稿热线: (010) 88379604

读者信箱: hzjsj@hzbook.com

前言

ExtJS的前身是YUI (Yahoo User Interface)。经过不断发展与改进，ExtJS现在已经成功发布了4.0版本，并且成为构建RIA Web应用的一套最完整、最成熟的JavaScript基础库。利用ExtJS构建的RIA Web应用具有与桌面程序一样的标准用户界面和操作方式，并且能够跨不同的浏览器平台使用。ExtJS已经成为开发具有完美用户体验的Web应用选择。许多程序员在RIA Web开发的征途中选择ExtJS作为克敌制胜的武器，因为他们可以享受ExtJS带来的类似于Win32编程的原生态快乐。

现在用户对体验的要求越来越高。对于Web应用开发者而言，ExtJS无疑是优秀的解决方案，它能够帮助开发者快速实现良好用户界面的开发。而且由于ExtJS是在YUI的基础上拓展出来的，在国内具有广泛的用户，再加上ExtJS又是开源的，所以ExtJS的生命力极强。

ExtJS的功能时刻吸引我们去深入探究，但由于ExtJS自身的特点，使得我们在学习它的时候会遭遇如下的诸多困惑：

- ExtJS为用户提供了功能强大的组件，但这些组件并不能满足项目中功能多变的需求，如何学会扩展已有组件呢？除了扩展组件，ExtJS组件还提供了插件功能，学会插件的编写也是一个不错的选择。那么，如何编写插件呢？
- 要扩展ExtJS组件只阅读API文档是不行的，还必须熟悉相关的组件源代码，这样才能有效地实现组件扩展，所以应该快速掌握ExtJS的源代码。可如何快速掌握呢？
- ExtJS源代码的数量和结构比较复杂，在其中大量使用了JSON对象实现功能的封装，所以熟悉JSON对象是必须的。怎样熟悉JSON对象呢？
- 如果直接使用ExtJS的原始组件，要配置的内容比较复杂，学会组件式的开发方法可以大幅减少代码量，减轻维护的强度。什么是组件式的开发方法呢？
- 在ExtJS的源代码中经常出现一些不常见的语法形式，不太容易理解。怎么理解这些源代码呢？

在本书中你会找到上述所有问题的答案。本书通过各种实例，全面地介绍如何使用ExtJS来进行RIA Web开发，并对ExtJS在开发过程中与其他相关技术的结合进行了说明，最终开发出功能强大、界面标准、用户体验完美的Web应用。本书的案例来自于ExtJS开发的实际应用，大部分是来自实际的Web应用项目，对开发者具有极强的指导意义。

本书特色

阅读门槛低——不要求读者对ExtJS有所了解，只要熟悉JavaScript编程，本书就能带你从头开始领略ExtJS的风采。

平台开发的概念——可以认为ExtJS是一种软件开发平台，而非一个普通JavaScript框架，ExtJS是用桌面程序开发思想引导基于ExtJS的Web应用。

来自于实践——本书中大量的案例都是实际项目情景的缩影，在讲解时穿插大量的实践经验提示，实践的指导意义强大。

典型场景应用——本书在讲述各章内容时穿插了一个对常见Web应用框架使用ExtJS进行改造的实例。Web应用框架是最基本的典型场景应用，每个Web开发者都会遇到。

包含了大量ExtJS 4.0新特性——本书是对第1版的完全升级，对ExtJS 4.0中包含的新特性进行了有重点的详细介绍，是学习ExtJS 4.0新特性的指导手册。

致谢与分工

本书由卫军、夏慧军和孟腊春编著。此外还要感谢王斌、张强林、万雷、张赛桥、刘军华、黄北军、陈洪军、黄中林、陈鲲、赵腾伦等人，他们为本书的出版做出了贡献。本书由成都道然科技有限责任公司策划和审定。最后感谢Testdll、清源、Alex XRDU等多位技术专家提出的宝贵意见和建议。

限于水平和精力有限，书中疏漏之处难免，欢迎读者批评指正。为方便读者更好地理解 and 阅读本书，我们将书中的代码与示例提供给读者，相关代码的下载地址为：www.hzbook.com 或 www.dozan.cn。

目 录

前言

第1章 认识ExtJS	1
1.1 ExtJS的精彩表现	1
1.2 ExtJS的前世今生	2
1.3 是否真的需要学习ExtJS	4
1.4 ExtJS 4.0新特性	8
1.5 ExtJS UI组件基础	10
1.6 了解一下类似技术	13
1.7 本章小结	16
第2章 开始ExtJS之旅	17
2.1 认识ExtJS的开发包	17
2.2 也从Hello world开始	18
2.3 ExtJS中的基本概念	19
2.4 非常有用的开发工具	20
2.4.1 开发插件spket	20
2.4.2 FullSource	23
2.4.3 Microsoft Script Debugger	24
2.5 ExtJS对原有JavaScript对象的扩展	25
2.5.1 Ext.Array	25
2.5.2 Ext.Date	27
2.5.3 Ext.Function	29
2.5.4 Ext.Number	30
2.5.5 Ext.String	31
2.5.6 Ext.Object	32
2.6 本章小结	33
第3章 ExtJS 4.0的基本功能	34
3.1 ExtJS组件配置说明	34
3.1.1 JSON介绍	34
3.1.2 ExtJS组件配置方式介绍	35
3.2 信息提示框组件介绍	36

3.2.1 认识Ext.window.MessageBox	37
3.2.2 Ext.MessageBox.alert()	39
3.2.3 Ext.MessageBox.confirm()	40
3.2.4 Ext.MessageBox.prompt()	41
3.2.5 Ext.MessageBox.wait()	42
3.2.6 Ext.MessageBox.show()	42
3.2.7 Ext.window.MessageBox的其他 功能	45
3.3 进度条组件介绍	48
3.3.1 认识Ext.ProgressBar	48
3.3.2 手工模式的进度条	50
3.3.3 自动模式的进度条	51
3.3.4 自定义样式的进度条样式	52
3.4 实现工具栏和菜单栏	53
3.4.1 认识Ext.toolbar.Toolbar	53
3.4.2 只包含按钮的简单工具栏	54
3.4.3 包含多种元素的复杂工具栏	56
3.4.4 启用和禁用工具栏	57
3.4.5 认识Ext.menu.Menu菜单	58
3.4.6 最简单的菜单栏	58
3.4.7 创建二级或多级菜单	60
3.4.8 将更多组件加入菜单	61
3.4.9 具有选择框的菜单	62
3.5 本章小结	64
第4章 最常用的表单	65
4.1 表单及表单元素	65
4.1.1 了解Ext.form.Basic基本表单	65
4.1.2 认识 Ext.form.Panel表单面板	68
4.1.3 Ext.form.field.Base基础表单字段	70
4.1.4 Ext.form.field.Text文本域	74
4.1.5 Ext.form.field.TextArea文本区	76

4.1.6 Ext.form.field.Number数字输入框	78	5.2.1 Auto自动布局	136
4.1.7 Ext.form.field.Checkbox复选框和 Ext.form.field.Radio单选框	80	5.2.2 Fit自适应布局	137
4.1.8 Ext.form.CheckboxGroup和Ext.form. RadioGroup	81	5.2.3 Accordion折叠布局	138
4.1.9 Ext.form.field.Trigger触发字段	83	5.2.4 Card卡片式布局	139
4.1.10 Ext.form.field.Spinner微调字段	84	5.2.5 Anchor锚点布局	141
4.1.11 Ext.form.field.Picker拾取器字段	85	5.2.6 Absolute绝对位置布局	144
4.1.12 Ext.form.field.ComboBox组合框	86	5.2.7 CheckboxGroup复选框组布局	145
4.1.13 Ext.form.field.Time时间选择框	92	5.2.8 Column列布局	145
4.1.14 Ext.form.field.Date日期选择框	94	5.2.9 Table表格布局	148
4.1.15 Ext.form.field.Hidden隐藏字段	97	5.2.10 Border边框布局	149
4.1.16 Ext.form.field.HtmlEditor编辑器 字段	98	5.2.11 Box盒布局	151
4.1.17 Ext.form.field.Display只读文本 字段	101	5.3 使用ViewPort	152
4.1.18 Ext.form.Label标签字段	102	5.4 Ext.tab.Panel页签	153
4.1.19 Ext.form.FieldSet字段集	103	5.5 本章小结	155
4.1.20 Ext.form.FieldContainer容器字段	105	第6章 常用工具类与函数	156
4.1.21 Ext.form.field.File文件上传字段	107	6.1 非常有用的Ext.core.Element	156
4.2 实现表单验证	109	6.2 Ext常用函数	158
4.2.1 常见的验证类型	109	6.2.1 Ext.onReady()	158
4.2.2 表单验证 (VType)	110	6.2.2 Ext.get()	160
4.2.3 自定义VType验证	111	6.2.3 Ext.select()	160
4.3 表单的提交和加载	114	6.2.4 Ext.query()	161
4.3.1 Ext.form.action.Action基础	114	6.2.5 Ext.getCmp()	163
4.3.2 Ajax模式的表单数据加载	116	6.2.6 Ext.getDom()	163
4.3.3 Ajax模式的表单数据提交	119	6.2.7 Ext.isEmpty()	164
4.3.4 标准模式的表单数据提交	121	6.2.8 Ext.namespace()	165
4.3.5 使用Direct技术	123	6.2.9 Ext.each()	165
4.4 本章小结	126	6.2.10 Ext.apply()	166
第5章 面板及布局类	127	6.2.11 Ext.encode()	167
5.1 面板panel	127	6.2.12 Ext.htmlDecode()	168
5.1.1 认识Ext.panel.Panel	127	6.2.13 Ext.typeOf()	169
5.1.2 Ext.panel.Panel的主要功能	128	6.3 Ext.core.DomHelper	169
5.1.3 使用Ext.panel.Panel	131	6.3.1 Ext.core.DomHepler.append()	170
5.2 标准布局类	136	6.3.2 Ext.core.DomHelper.applyStyles()	170
		6.3.3 createTemplate()	171
		6.3.4 insertAfter()	172
		6.3.5 insertBefore()	172
		6.3.6 insertFirst()	173

6.3.7 insertHtml()	174	6.10.2 将数组中的对象加入到集合中	191
6.3.8 overwrite()	175	6.10.3 移除集合中的对象	191
6.4 Ext.core.DomQuery	176	6.10.4 克隆集合	192
6.4.1 compile()	176	6.10.5 匹配集合中关联对象	192
6.4.2 filter()	176	6.10.6 迭代集合中的对象调用指定的方法	192
6.4.3 is()	177	6.10.7 获取集合中的对象	193
6.4.4 jsSelect()	177	6.10.8 该类中的其他一些有用的方法	194
6.4.5 selectNode()	177	6.10.9 该类中重要的事件 (Events)	196
6.5 Ext.util.CSS	178	6.11 Ext.util.TaskRunner	197
6.5.1 createStyleSheet()	178	6.11.1 构造函数	197
6.5.2 getRule()	179	6.11.2 启动一个线程start()	198
6.5.3 swapStyleSheet()	179	6.11.3 停止一个线程stop()	198
6.5.4 removeStyleSheet()	181	6.11.4 停止所有的线程stopAll()	198
6.6 Ext.util.ClickRepeater	182	6.12 Ext.util.TextMetrics得到块状化文本规格	199
6.6.1 click()	182	6.12.1 块的绑定	199
6.6.2 mousedown()	183	6.12.2 实例化对象	199
6.6.3 mouseup()	183	6.12.3 获取文本的高度	199
6.7 Ext.util.DelayedTask提供setTimeout的简单替代	184	6.12.4 获取文本的宽、高	200
6.7.1 cancel()	184	6.12.5 获取文本的宽度	200
6.7.2 delay()	185	6.12.6 获取指定节点内文本块的宽、高	200
6.8 Ext.util.Format 提供常用的格式化方法	185	6.12.7 指定文本块的宽	201
6.8.1 ellipsis()	185	6.13 Ext.KeyNav为元素提供简单的按键处理方法	201
6.8.2 capitalize()	185	6.13.1 实例化一个键盘绑定对象	202
6.8.3 date()	186	6.13.2 废弃原有键盘绑定	202
6.8.4 htmlEncode()	186	6.13.3 将废弃的键盘绑定重新生效	203
6.8.5 htmlDecode()	186	6.14 Ext.KeyMap 提供更灵活强大的对按键的处理方法	203
6.8.6 stripTags()	187	6.14.1 构造函数	203
6.8.7 substr()	187	6.14.2 给对象添加键盘绑定	204
6.8.8 lowercase()	187	6.14.3 废弃已绑定到KeyMap的配置	205
6.8.9 number()	187	6.14.4 将KeyMap或废弃的配置重新生效	205
6.8.10 round()	188	6.14.5 获取当前KeyMap配置是否有效	205
6.9 Ext.util.JSON 编码和解码JSON对象	189		
6.9.1 decode()	189		
6.9.2 encode()	189		
6.10 Ext.util.MixedCollection	190		
6.10.1 向集合加入对象	190		

6.14.6 事件绑定函数	205	8.5.1 创建实现换肤功能的自定义组件 Ext.ux.ThemeChange	250
6.15 本章小结	206	8.5.2 查看换肤效果	251
第7章 让ExtJS开始响应事件	207	8.6 本章小结	251
7.1 ExtJS事件与浏览器标准事件的异同	207	第9章 ExtJS对Ajax的支持	252
7.1.1 事件绑定方式	207	9.1 Ext.Ajax入门	252
7.1.2 自定义事件	209	9.1.1 Ext.Ajax.request方法详解	253
7.1.3 跨浏览器事件	210	9.1.2 Ext.Ajax.request操作示例	255
7.2 ExtJS事件特性	211	9.2 Ext.ElementLoader基础	259
7.2.1 Ext.util.Observable (事件基类)	211	9.2.1 Ext.ElementLoader.load操作示例	260
7.2.2 举例一: addEvents、fireEvent、 addListener	213	9.2.2 Ext.ElementLoader其他方法示例	262
7.2.3 举例二: capture拦截器示例	214	9.3 利用Ajax优化Web应用框架	264
7.2.4 举例三: addManagedListener受管 事件监听器	215	9.3.1 多级联动菜单	264
7.2.5 举例四: relayEvents传播分发事件	216	9.3.2 天气情况查询	267
7.3 Ext.EventObject (事件对象)	217	9.3.3 自动保存的网络记事本	270
7.4 Ext.EventManager (事件管理器)	218	9.3.4 网页计算器	272
7.5 本章小结	220	9.4 本章小结	274
第8章 常见Web框架的ExtJS改造	221	第10章 增强型模板	276
8.1 常见Web应用框架说明	221	10.1 熟悉Ext.Template	276
8.2 简易书籍管理系统	221	10.2 Ext.XTemplate基础	280
8.2.1 总体设计	222	10.2.1 自动填充数组和作用域切换	280
8.2.2 数据库设计	222	10.2.2 在子模板中访问父对象	281
8.2.3 数据库脚本	222	10.2.3 数组索引和简单运算支持	282
8.2.4 工程框架设计	223	10.2.4 自动渲染简单数组	283
8.2.5 页面设计	231	10.2.5 使用基本的条件逻辑判断	283
8.3 怎样开始	232	10.2.6 在模板中执行任意代码	284
8.3.1 页面改造	233	10.2.7 使用模板成员函数	285
8.3.2 与服务器交互方式的改造	233	10.3 在ExtJS组件中使用模板	286
8.4 开始改造吧	233	10.4 使用模板的优势	288
8.4.1 改造主页面布局	233	10.5 Ext.view.View数据视图类详解	288
8.4.2 改造菜单栏	235	10.6 本章小结	291
8.4.3 改造书籍维护界面	237	第11章 数据模型	292
8.4.4 改造书籍新增界面	239	11.1 ExtJS数据模型基础	292
8.4.5 改造书籍修改界面	246	11.2 Model数据实体模型	293
8.4.6 改造书籍删除功能	248	11.2.1 创建实体	295
8.5 换肤的实现	250	11.2.2 Ext.data.validations数据验证	296
		11.2.3 数据代理 (读取与保存)	299

11.2.4 Ext.data.HasManyAssociation一对多关系	300	12.2.2 数字列Ext.grid.column.Number	340
11.2.5 Ext.data.BelongsToAssociation多对一关系	303	12.2.3 日期列Ext.grid.column.Date	340
11.3 Proxy数据代理	306	12.2.4 动作列Ext.grid.column.Action	341
11.3.1 Ext.data.proxy.Proxy	307	12.2.5 模板列Ext.grid.column.Template	343
11.3.2 Ext.data.proxy.Client	307	12.2.6 行号列Ext.grid.RowNumberer	343
11.3.3 Ext.data.proxy.Memory	307	12.2.7 自定义渲染函数	344
11.3.4 Ext.data.proxy.WebStorage	308	12.3 选择模式Selection	345
11.3.5 Ext.data.proxy.LocalStorage	309	12.3.1 选择模式 Ext.selection.Model	345
11.3.6 Ext.data.proxy.SessionStorage	310	12.3.2 单元格选择模式Ext.selection.CellModel	346
11.3.7 Ext.data.proxy.Server	310	12.3.3 行选择模式Ext.selection.RowModel	348
11.3.8 Ext.data.proxy.Ajax	311	12.3.4 复选框选择模式Ext.selection.CheckboxModel	348
11.3.9 Ext.data.proxy.Rest	315	12.4 表格特性Feature	349
11.3.10 Ext.data.proxy.JsonP	316	12.4.1 表格行体Ext.grid.feature.RowBody	349
11.3.11 Ext.data.proxy.Direct	318	12.4.2 表格汇总Ext.grid.feature.Summary	351
11.4 Reader数据读取器	319	12.4.3 表格分组Ext.grid.feature.Grouping	352
11.4.1 Ext.data.reader.Reader	320	12.4.4 分组汇总Ext.grid.feature.GroupingSummary	353
11.4.2 Ext.data.reader.Json	320	12.5 表格插件plugin	354
11.4.3 Ext.data.reader.Xml	323	12.5.1 单元格编辑插件Ext.grid.plugin.CellEditing	354
11.4.4 Ext.data.reader.Array	324	12.5.2 行编辑插件Ext.grid.plugin.RowEditing	356
11.5 Writer数据写入器	325	12.5.3 拖曳插件Ext.grid.plugin.DragDrop	357
11.5.1 Ext.data.writer.Writer	325	12.6 属性表格property.Grid	359
11.5.2 Ext.data.writer.Json	326	12.7 表格分页	361
11.5.3 Ext.data.writer.Xml	327	12.8 本章小结	362
11.6 Store数据集	327	第13章 Tree组件	363
11.6.1 Ext.data.AbstractStore	327	13.1 树形面板Ext.tree.Panel	363
11.6.2 Ext.data.Store	328	13.1.1 多列树	364
11.6.3 Ext.data.ArrayStore、Ext.data.JsonStore及Ext.data.XmlStore	333	13.1.2 带复选框的树	366
11.6.4 Ext.data.DirectStore和Ext.data.JsonPStore	334		
11.7 本章小结	335		
第12章 Grid组件	336		
12.1 表格面板Ext.grid.Panel	337		
12.2 表格列Column	339		
12.2.1 布尔列Ext.grid.column.Boolean	340		

13.1.3 树面板间的拖曳	366	17.3.1 Ext.draw.Component图形组件 详解	411
13.2 Ext.data.TreeStore	368	17.3.2 Ext.draw.Sprite子画面组件详解	411
13.2.1 分级加载树节点	368	17.3.3 绘制基本图形	412
13.2.2 整体加载树节点	370	17.3.4 绘制不规则图形	413
13.3 本章小结	371	17.3.5 绘制文字	414
第14章 ExtJS与服务端框架的整合	372	17.3.6 线性渐变	415
14.1 XML与JSON的生成	372	17.3.7 图形旋转	417
14.1.1 XStream基础	372	17.3.8 动态变换	418
14.1.2 JSON-lib简介	376	17.3.9 拖曳缩放	419
14.2 ExtJS与Struts的整合方式	378	17.3.10 拖曳移动	420
14.3 ExtJS与Struts 2的整合方式	380	17.4 本章小结	421
14.3.1 Struts 2生成JSON数据	381	第18章 Chart图表	422
14.3.2 Struts 2生成XML数据	383	18.1 Ext.chart图表组件概述	422
14.4 本章小结	384	18.2 坐标轴组件详解	423
第15章 主题	385	18.2.1 Ext.chart.axis.Numeric数值轴	425
15.1 Sass和Compass概述	385	18.2.2 Ext.chart.axis.Category分类轴	426
15.1.1 Sass简介	385	18.2.3 Ext.chart.axis.Time时间轴	426
15.1.2 Compass简介	385	18.2.4 Ext.chart.axis.Gauge仪表轴	427
15.2 ExtJS主题样式开发	387	18.2.5 Ext.chart.axis.Radial雷达轴	427
15.2.1 基本配置文件config.rb	387	18.3 图表序列详解	427
15.2.2 定制主题	388	18.3.1 Ext.chart.series.Pie饼状图	428
15.3 组件UI开发	389	18.3.2 Ext.chart.series.Cartesian直角坐标 系图表	431
15.3.1 组件UI混入函数介绍	390	18.3.3 Ext.chart.series.Bar条形图	432
15.3.2 开发面板组件UI	391	18.3.4 Ext.chart.series.Column柱状图	433
15.4 本章小结	393	18.3.5 Ext.chart.series.Line折线图	435
第16章 Ext.Direct	394	18.3.6 Ext.chart.series.Scatter散点图	437
16.1 如何配置和使用Ext.Direct	394	18.3.7 Ext.chart.series.Area面积图	438
16.1.1 支持类库的下载	394	18.3.8 Ext.chart.series.Radar雷达图	440
16.1.2 配置服务器端功能	395	18.3.9 Ext.chart.series.Gauge仪表图	442
16.1.3 配置客户端	400	18.4 在图表中使用主题	443
16.1.4 远程调用实例	402	18.5 复合图表	446
16.2 本章小结	408	18.6 图表与表格	446
第17章 Draw图形	409	18.7 本章小结	449
17.1 常见浏览器绘图技术介绍及对比	409	附录A	450
17.2 精彩的图形世界	410		
17.3 ExtJS图形包	410		

第1章 认识ExtJS

ExtJS是用JavaScript、CSS和HTML等技术实现的主要用于创建用户界面，且与后台技术无关的前端Ajax框架，还被用来开发RIA（富客户端）的Web应用。

ExtJS的UI组件模型和开发理念继承自Yahoo用户组件库YUI和Java平台上的Swing，它为开发者屏蔽了大量的Web UI操作以及跨浏览器方面的处理。ExtJS要比开发者直接针对DOM和W3C对象模型开发Web UI快速而且轻松许多。无论是从Web UI界面美观丰富上，还是从功能强大完善上来看，ExtJS都可算得上是Web UI组件库方面的佼佼者，也是一款不可多得的JavaScript客户端技术的精品之作。

本章内容提示：

- ExtJS的精彩表现
- ExtJS的前世今生
- 是否真的需要学习ExtJS
- ExtJS 4.0新特性
- ExtJS UI组件基础
- 了解一下类似技术

1.1 ExtJS的精彩表现

“百闻不如一见”，这话对软件开发相关技术来说同样适用，特别是对于以UI闻名于世的ExtJS来说，实践将好过任何语言的描述。而在所有ExtJS的UI组件中，Grid组件是使用最频繁、功能最复杂，同时也是实现最好的。先来见识一下这个“无论是界面之精美，还是功能之强大”都无人能出其右的Grid组件吧！

如图1-1所示，这就是ExtJS的Grid组件的一个展示效果。从图中可以看出，Grid组件的界面基本上和桌面程序的Grid组件一样，从图上也能看出该组件的一些功能，在功能上相对于桌面程序的Grid组件，ExtJS的Grid组件也毫不逊色。

单选行、多选行，高亮显示选中的行，拖曳改变列宽度，按列排序，这些只能算Grid组件的基本功能。自动生成行号，支持checkboxbox全选，动态选择显示哪些列，支持本地以及远程分页，可以对单元格按照自己的想法进行渲染，这些应该算一个功能较强的Grid组件应该拥有的功能，ExtJS的Grid组件也都拥有。

可编辑Grid添加新行，删除一行或多行，提示脏数据，拖曳改变Grid大小，Grid之间拖曳一行或多行，甚至可以在Tree和Grid之间进行数据拖曳，这些即使在桌面程序的Grid组件中都是非常神奇的功能，却在一个基于Web的UI组件中存在，这不得不让人惊奇于ExtJS的Grid组件功能之强大。

Company	Price	Change	% Change	Last Updated
Sort Ascending	\$71.72	0.02	0.03%	09/01/2008
Sort Descending	\$31.61	-0.48	-1.54%	09/01/2008
Lock Column	\$29.01	0.42	1.47%	09/01/2008
Unlock Column	\$83.81	0.28	0.34%	09/01/2008
Columns	\$52.55	0.01	0.02%	09/01/2008
Caterpillar Inc.	\$64.13	0.31	0.49%	09/01/2008
Citigroup, Inc.	\$75.43	0.53	0.71%	09/01/2008
E.I. du Pont de Nemours and Co.	\$67.27	0.92	1.39%	09/01/2008
Exxon Mobil Corp.	\$49.37	0.02	0.04%	09/01/2008
General Electric Company	\$40.48	0.51	1.28%	09/01/2008
General Motors Corporation	\$68.10	-0.43	-0.64%	09/01/2008
Microsoft Corporation	\$34.14	-0.08	-0.23%	09/01/2008
Walmart Stores Inc.	\$30.27	1.09	3.74%	09/01/2008

图1-1 ExtJS的Grid组件效果

当然，Grid组件不是ExtJS的全部，实际上，ExtJS提供了非常丰富的组件库，基本上可以分为基本组件、表单及元素组件、图表组件3大类多达100多种组件，再加上各种各样的扩展非官方组件和可选组件，基本上满足了开发RIA应用的各种需求。虽然每一个组件不可能都如Grid组件一样令人惊讶，但是见微知著，ExtJS的其他组件不管是从UI表现上还是功能上都是非常的出色。

现在越来越多的其他Web UI组件库都在向ExtJS学习或者看齐，这其中尤以基于Google GWT的MyGWT为甚，虽然MyGWT不是基于JavaScript的Web UI组件库，但是在其通过Java生成的客户端UI中，完全实现了ExtJS的组件效果与功能。所以，ExtJS的组件极有可能成为开发RIA应用事实上的UI标准。

要成为事实上的标准必须满足两个条件：第一是使用范围非常广泛。在这一点上，ExtJS的发展非常迅速，已经有越来越多的产品和项目采用它作为UI框架，逐步向Web UI标准迈进，在这个过程中需要有更多的开发者来了解它、学习它，最终让其在更多的项目中发挥作用；第二是要实现得非常完善，功能强大。在这一点上，ExtJS基本上已经做到了，尤其是2011年发布的4.0版本在功能、效率和稳定性上都得到了大幅度提升。当然，完美无止境，希望ExtJS更加完美、更加强大，这将是每一个使用ExtJS开发者共同的期待。

是不是还在惊叹ExtJS的Grid组件那炫目效果与强大的功能呢？那让这种惊叹继续下去吧！

图1-2展示了ExtJS中另一个具有代表性的组件，Tree组件。该组件不仅能展示树型的数据，还能将数据在不同的分支之间拖动，能够动态删除分支与节点，也能动态添加分支与节点，在树之间相互拖曳数据，并且支持流行的TreeGrid效果。

1.2 ExtJS的前世今生

ExtJS来源于YUI，YUI是Yahoo! User Interface Library的简称，它是一个使用JavaScript编写的工具和控件库。最初YUI只是在Yahoo

Task	Duration	Assigned To
Project: Shopping	13h 15m	Tommy Maintz
Housewares	1h 15m	Tommy Maintz
Remodeling	12h 0m	Tommy Maintz
Paint bedroom	2h 45m	Tommy Maintz
Retile kitchen	6h 30m	Tommy Maintz
Decorate living room	2h 45m	Tommy Maintz
Fix lights	45 mins	Tommy Maintz
Reattach screen door	2h 0m	Tommy Maintz
Project: Testing	2h 0m	Core Team

图1-2 ExtJS的Tree组件效果

内部使用，用来构建具有丰富用户体验的Web应用。其实几乎每个做Web开发的软件公司都有这种内部的JavaScript工具与控件库，除了在功能的强大上有一些差别之外并没有什么特别之处。

一切都在Yahoo将YUI开源后发生了变化。开源后的YUI首先是在使用范围上发生了变化，有一些开发组将YUI引入了自己的项目中；然后开发人员组成也发生了变化，以前只有Yahoo的开发人员在维护与改进YUI，由于开源，越来越多的开发者加入到了改进与扩展YUI的工作中来。

在这其中，最早被公认为比较出色的YUI扩展是Jack Slocum（杰克斯·洛克姆）的YUI-Ext。早在2006年年初，Jack就开始了YUI的扩展工作。这些扩展很快组织成一个独立的代码库并以“YUI-Ext”的名义进行了发布。

最开始，Jack只是打算对基于BSD协议的Yahoo! User Interface Library进行自定义的扩展，但是由于Jack的工作使这一扩展非常出色，青出于蓝而胜于蓝，风头一度盖过其父辈YUI。开发者对它的热情非常高，甚至很多人在不太了解它的情况之下就把它投入项目。有这么一个比较形象的比喻：初一谋面，在并不了解一个人的家庭、教育、品行等背景下，只因为她有一副精致漂亮的外观，就对其陷入了疯狂的倾慕之中。因此有些冷静的业内人士建议那些开发者，在投入项目前，要认真仔细地了解YUI-Ext的内在原理以及与其他Ajax库不同的地方。虽然如此，但是这足以说明YUI-Ext的成功。

在2006年秋天，Jack发行了版本为0.33的YUI-Ext。在年底之前，这个库已大受欢迎，并且名字也简化为Ext，这反映了它作为一个独立框架的成熟。

在2007年年初，Jack成立了运作Ext的商业公司，Ext的名字也改成了更能标识Ext实质的ExtJS，在公司的运作下，ExtJS迅速发展。

2007年4月1日，ExtJS发布1.0正式版，使用LGPL3.0和一个商业执照，这也标识了ExtJS的完全成熟与独立，从此ExtJS走入了更多开发者的生活。

2007年12月3日，ExtJS发布2.0正式版，这是ExtJS使用最广的一个版本，它修正了1.0存在的内存泄漏问题，迎来了ExtJS的大发展。

2008年2月24日，ExtJS发布2.02正式版，这是最后一个使用LGPL3.0授权协议的版本。

2008年4月20日，ExtJS发布2.1正式版，这个版本的授权协议由LGPL3.0改为GPL 3.0。

2009年7月6日，ExtJS发布3.0正式版，这个版本对2系列的功能进行了加强，增加了chart图表、direct远程方法调用、新的布局类等。

2009年7月17日，ExtJS发布2.3正式版，这是目前最新的2系列的版本。

2010年11月30日，ExtJS发布3.3.1正式版，这是目前最新的3系列的版本。

2011年4月26日，ExtJS发布4.0.0正式版，这是ExtJS发布以来最重要的一次更新，这个版本对大量组件进行了重构，与jQueryTouch和Raphaël强强联合在移动领域和绘图领域扩展了市场，这个版本也是本书介绍的基础版本。

直至现在，ExtJS已发展涵盖美国、日本、中国、法国、德国等全球范围的用户，而现在的版本也升级到ExtJS 4.0。相信随着版本的不断升级，ExtJS将会越来越强大，使用的范围也

会越来越广。

1.3 是否真的需要学习ExtJS

如果是刚加入到项目组的新人，并且这个项目组正在采用ExtJS来开发RIA应用，那么，这个问题不需要回答，没有选择，必须学习ExtJS。这种刚性需求是没法回避的，并且也是不能够加以讨论的。那除此之外，或许您只是对客户端开发比较感兴趣，那么需要学习ExtJS吗？或许偶然看到ExtJS组件的炫目效果，是否只因为欣赏这种效果而需要开始学习ExtJS呢？甚至只是偶尔听到ExtJS这个名词，从而不经意间翻开了本书，需要学习ExtJS吗？

除了前面提到的刚性需求外，是否需要学习ExtJS其实很好回答。在无刚性需求下，学习一样新的技术无外乎以下3种原因：一是兴趣，您对这个技术方向比较感兴趣；二是潜在需求，您所在的项目或者将来的项目正在考虑引进这种技术；三是形成完整的技术知识体系，为自己的技术发展打下基础。

如果我们非常熟悉DOM、JavaScript和CSS，并且在以前的项目开发中做过Web UI的开发，为此付出过辛勤与汗水，并且收获了喜悦与自豪的话，那么一定要学习ExtJS。因为ExtJS是强大的客户端组件库，它从架构、功能和表现上都是客户端技术的精品，我们肯定不会错过这次绝好的学习机会。

如果项目正在考虑引进ExtJS，那现在开始学习ExtJS吧！除了了解其强大的组件功能，绚丽的界面效果之外，您更应该了解采用ExtJS后对传统Web开发人员固有观念的冲击，项目组对高级JavaScript开发人员将有更迫切需求。在此阶段，不仅要想到采用ExtJS对Web应用在界面美观性与标准性，用户使用体验上的巨大提升，同时也要想到采用ExtJS为开发带来的巨大挑战。

ExtJS的开发理念来源于传统的桌面软件开发，对有过VB、Delphi或者Java Swing等可视化桌面软件开发经历的开发者来说，学习ExtJS开发是一件容易而且愉快的事情。

然而，五年之前开始流行Web应用以来，越来越多的系统都开始采用Web方式来开发了，桌面软件的需求量越来越少，导致对桌面软件开发人员的需求也越来越少。需求决定市场，特别是近几年走上工作岗位的开发者，基本上来说没有从事过桌面软件的开发，从接触开发起就是学习Web应用的开发，他们对桌面软件开发基本上没什么概念。这样一旦接触到ExtJS，在思想上需要做一些转变，所以一开始会感到比较迷茫。

在此有必要对桌面软件开发方式与传统Web应用开发方式作一点说明与对比，在这里传统Web应用开发方式特指没有引入Web UI组件库的开发方式，在图1-3中对这两种开发方式做了一下比较。

如图1-3所示，对于桌面软件开发来说，空白的窗体是其开发的对象，在空白窗体中组合放置各种Windows UI组件，从而最终形成我们常见的Windows界面。而对Web应用来说，空白的Web页也是我们开发的对象，在这里，没有组件，只有HTML、JavaScript、CSS、图片文件，利用这些元素，最终形成Web应用界面。两种开发方式中，桌面软件的组件化与标准化是区别于Web应用的最大标志。而Web应用却是以零散与自由为特点，所有元素开发者都必须事无巨细地照顾到，并最终像美工一样将各种元素汇集在一起，形成一个用户可以接受的用户界面。

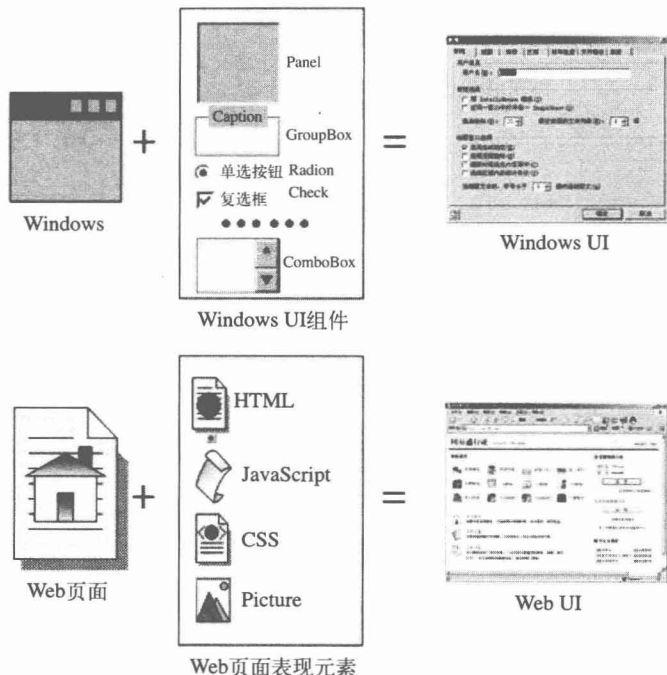


图1-3 桌面软件与传统Web应用开发方式对比

桌面软件的开发方式更容易理解，也容易学习，并且UI元素视觉与操作都有统一标准；但是缺少灵活性，很难突破组件的界定；Web应用开发方式无固定线路可循，对开发者的技术要求高，UI元素视觉与操作都无统一标准；但是其开发灵活，基本上能想到的就能做到。对于Web应用来说，开发一个较好的UI对美工与开发者的水平都是一个重大的考验，并且双方还必须精诚合作；而对于桌面软件来说，一个初学者经过简单的摸索，参照Windows的标准UI，就能很快开发一个符合绝大多数人审美观念与操作习惯的UI。

图1-4展示了采用了ExtJS后Web应用开发的情况。空白的Web页面依然是我们的开发对象，与传统的Web应用开发方式不同的是，因为引入了ExtJS库，有了一套完整的UI组件可以使用。所以可以在空白的Web页面上组合放置各种ExtJS的UI组件，与开发桌面软件放置Windows UI组件一样。

运行时，ExtJS组件库会自动生成形成界面的HTML、JavaScript、CSS和图片等元素，并最终在浏览器上渲染出ExtJS的Web UI。在这个过程中最重要的是开发者不再需要直接面向HTML、JavaScript、CSS和图片等基础元素，也无须去管理这些元素烦琐的放置组合，这一切都由ExtJS的库来代劳；开发者要面对的是UI组件，只要按照组件的开发规则进行开发即可。操作组件相对于操作HTML等元素来说粒度更大；粒度越大，就会获得更多的简易性，但是会失去灵活性。比如说由于某种特殊需要，需要制作一个特别的UI组件，那么就必须重新开发一个ExtJS组件或者基于现有组件进行扩展，而这个工作显然会比直接修改HTML更加复杂。

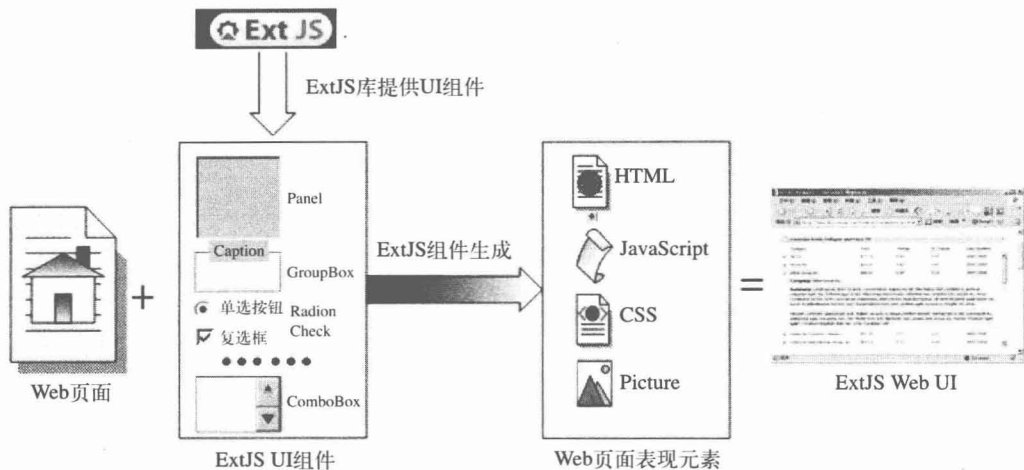


图1-4 采用ExtJS的Web应用开发方式

作为开发人员来说，最好形成一个技术体系，一个完善的技术体系能够有力地支撑技术生涯的发展。可能这样的说明还是比较虚，这里举一个比较具体的例子可能对读者理解此问题有更好的帮助。

在Windows之前微软有过一个MSDOS操作系统，对于最近几年走上开发岗位的开发来说，估计也就在课本上作为计算机历史介绍过。而对于年长的开发来说，其开发生涯就是从MSDOS开始的。虽然并不能认为MSDOS是Windows的前身，但是不可否认的是在Windows的底层确实或多或少存在着MSDOS的影子。在桌面软件开发的实践中，有过MSDOS使用与开发经验的开发人员比从来没有接触过MSDOS的开发人员在理解问题上与解决问题上都棋高一着，这充分体现了一个完善技术体系对开发者的影响。

再回到主题，学习ExtJS对一个开发来说，在其技术体系中到底处在一个什么位置上呢？我们依然要比较一下桌面软件开发与Web应用开发的发展过程，从而明确ExtJS在这个发展过程中所处的位置，图1-5展示了两种开发技术的发展过程。

如图1-5所示，桌面应用的运行环境是Windows，运行方式是以二进制代码方式来运行的。那么二进制代码是怎么获得的呢？在最开始的阶段，通过汇编语言的编程可以获得二进制代码（这里需要说明的是其实桌面应用即Windows软件实际上是没有通过汇编语言直接编程获得的，因为那样做难度太大，在这里提到的只是说明开发方式的一种变迁）。然而汇编语言太接近于机器语言，虽然灵活方便，但是对开发者要求太高，效率低下。接着开发方式发展到Windows API阶段，虽然Windows API屏蔽了绝大多数底层操作，使桌面程序开发有了较大发展，但是毕竟还是存在掌握难度大、开发效率低等缺点。最终开发方式进展到集成、可视化的开发工具时代，VB、Delphi就是其中的优秀代表。到了这个阶段，由于集成可视化开发工具的出现，大大降低了开发的门槛，越来越多的人能够开发桌面应用了，从而强有力地促进了桌面软件的开发，并进入普及阶段。

实际上，Web应用开发也逐渐呈现出与桌面应用开发相似的3个阶段。Web应用的运行环境