

实战Ext JS 6.2，掌握Ext JS前端开发技术

- 详细介绍Ext JS 6.2 Web前端开发技术
- 针对每个知识点使用一个统一的案例进行说明，快速提升实战能力
- 适合所有想全面深入掌握Ext JS 6.2 Web前端开发技术的人员阅读



JavaScript Framework for Web Apps

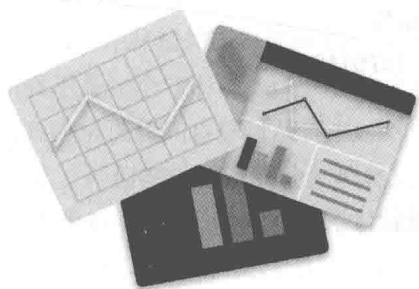
Ext JS 6.2 实战

黄灯桥 著



清华大学出版社





Web前端技术丛书

Ext JS 6.2 实战

黄灯桥 著

清华大学出版社
北京

内 容 简 介

本书是实战类图书，以 Ext JS 6.2 版本为基础，融入作者使用 Ext JS 进行开发的实践经验，通过一个具体的 CMS 应用程序实例，一步一步地带领读者进入 Ext JS 6.2 的开发世界，掌握使用 Ext JS 6.2 开发应用程序的技术。

本书分为 13 章，主要内容包括 Ext JS 6 概述、MVVM 模式、CMS 系统概述、使用 Sencha Cmd 创建应用程序、搭建开发环境、数据传输标准化、迁移管理模板、登录与权限控制、用户管理、媒体管理、文章管理、打包和发布以及 Ext JS 跨平台特性介绍。

本书适合 Ext JS 初学者、前端开发人员，也可以作为高等院校和培训机构相关专业师生的教学参考书。

本书封面贴有清华大学出版社防伪标签，无标签者不得销售

版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

图书在版编目（CIP）数据

Ext JS 6.2 实战 / 黄灯桥著. — 北京：清华大学出版社，2018

（Web 前端技术丛书）

ISBN 978-7-302-48815-6

I. ①E… II. ①黄… III. ①JAVA 语言—程序设计 IV. ①TP312.8

中国版本图书馆 CIP 数据核字（2017）第 273116 号

责任编辑：夏毓彦

封面设计：王 翔

责任校对：闫秀华

责任印制：刘海龙

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>, <http://www.wqbook.com>

地 址：北京清华大学学研大厦 A 座

邮 编：100084

社总机：010-62770175

邮 购：010-62786544

投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者：三河市君旺印务有限公司

经 销：全国新华书店

开 本：190mm×260mm

印 张：19

字 数：486 千字

版 次：2018 年 1 月第 1 版

印 次：2018 年 1 月第 1 次印刷

印 数：1~3000

定 价：59.00 元

产品编号：074289-01

前言

从我发表第一篇与 Ext JS 有关的博客开始，至今已有十个年头了。在这十年来，随着 Web 2.0 的兴起，与 Web 开发相关的框架可谓百花齐放，让人有点应接不暇，可以说是风云变幻，大浪淘沙！按说，我不是一个守旧的人，一有什么新事物总会去尝试一下，但在开发框架的选择上，一直钟情于 Ext JS。虽说其间也想过放弃 Ext JS，因为这个框架越来越趋于商业化，已经与早期的开源之路越走越远了，但在尝试过其他框架之后，我又不得不走回去，继续使用 Ext JS。

是什么让我对 Ext JS 依依不舍呢？情怀？身为程序员，我觉得千万不能做情怀党。所谓一叶障目，很容易陷入被动。除了 Java 真不想碰之外，我使用过的开发语言也算不少了，从最早的 Basic、FoxPro、ASP、Visual Basic、Borland C++、Delphi、ASP，到现在的 C#、PHP 等，只要是项目要求或者是最适合项目使用的语言，我都会去试去用。因而，这绝不是情怀的问题。

在 Ext JS 越来越商业化之后，有过那么一段时间，确实有抛弃 Ext JS 的念头。于是开始去尝试 JQuery UI、Angular、Bootstrap 和 BACKBONE 等框架，甚至还包括一些不太出名的框架，但最终面对一堆散列的脚本代码、一堆需要重写的组件之后，觉得还是使用 Ext JS 吧，省心省力，又或者是我的功力不够，对这些框架的理解还不深的原因吧。

终于等到 Ext JS 6 的发布了，感觉这次 Ext JS 真是脱胎换骨，越来越好使，越来越让我爱不释手了。Ext JS 最让人诟病的地方就是主题不多，样式太过单一；这次好了，终于不再单一了。不过，我对这个意见倒不多，我更关心的是是否能更快、更好地去开发单页面的应用程序。经过几个小项目的使用，可以说，已经摆脱不了 Ext JS 6 了。新的 MVVM 模式，真的让我拍手称快，这模式太好了，省了多少代码啊，开发效率的提高那可是杠杠的！再结合 ASP.NET MVC 和实体框架等快速开发工具，那效率，真的无与伦比！

开发效率一直是我的追求，时间就是成本，开发效率提高了，自然就降低了成本。而这正是我对 Ext JS 不离不弃的真正原因。

说了这么多废话，还是说回本书吧。在 Ext JS 群里，一直有人问我，什么时候写本关于 Ext JS 6 的书啊，太需要这样的书了，现在的中文资料实在太小了，而且也没有系统性开发的介绍。我很理解大家的需求，但这需要等待。清华大学出版社的夏毓彦编辑给了我这样一个机会，让我把《Ext JS 4.2 实战》升级一下。我当然不会错过这样的机会，立刻就答应了下来。可惜，就是太忙的关系，一直拖了很久。不过，这一拖也是值得的，因为相关的工作也基本上是使用 Ext JS 6 来实现的，本书的示例，可以说是这些项目经验的总结吧，也算是自己一个阶段性的总结。希望这些经验总结能帮到大家。

本书在后台语言 ASP.NET MVC 方面做了一点点的升级, 希望对使用 ASP.NET 进行开发的读者也有一点点帮助。本来是打算使用 ASP.NET Core 来实现的, 但总感觉还不是太成熟的产品, 而且 Core 2 还是测试阶段, 因而只好等下一次再和大家探讨了。

有很多 PHP 开发人员问我是否会有 PHP 的代码示例, 我想了想, 在最后一章, 加一节来说明也不是大问题, 也就是加入的 PHP 版本的示例, 但只限于 Yii 框架。在 PHP 项目选型的时候, 有 ThinkPHP、Yii 和 Laravel 三个框架摆在我的面前, 选哪个好呢? 最终我还是选择了 Yii, 因为这个框架对我来说更易于上手, 基本不用看多少文档就可以开始工作, 符合效率第一的要求。

本书的开发环境如下。

- 操作系统: Window 10
- 开发工具: Visual Studio 2017
- 数据库: SQL Server Express 2014
- 浏览器: Firefox 54

本书的源代码也与时俱进, 放在 GitHub 上, 大家可以自行到相应的地址下载。具体的下载地址是:

- ASP.NET 版本: <https://github.com/tianxiaode/SimpleCMS>
- Java 版本 (只有客户端): <https://github.com/tianxiaode/SimpleCMS-JAVA>
- PHP 版本 (只有客户端): <https://github.com/tianxiaode/SimpleCMS-PHP>

希望本书能给每个读者带来帮助, 如果对本书有任何意见和建议, 或者有任何技术上的问题, 请发邮件到 huangdengqiao@outlook.com, 或者加入 QQ 群: 391747779、193238033 和 131404874。如果想了解最新的 Ext JS 动态或我的最新博文, 可访问博客:

<http://blog.csdn.net/tianxiaode>

<http://dqhuang.blog.51cto.com/>

在本书的出版过程中, 得到了清华大学出版社编辑的大力支持, 在他们的努力下, 促成了本书的出版, 在此表示衷心感谢。此外, 还要感谢那些在互联网上默默耕耘的博客作者以及在各大论坛回复问题的人们, 是他们的努力耕耘, 才使我找到了解决问题的办法, 是他们让我有了更进一步提高技能的机会。

著 者

2017 年 11 月

目 录

第 1 章 Ext JS 6 概述	1
1.1 从 Ext JS 4 到 Ext JS 5	1
1.2 从 Ext JS 5 到 Ext JS 6	1
1.3 从 Ext JS 6.0 到 Ext JS 6.2	2
1.4 如何选择版本	2
1.5 基础知识	3
1.5.1 Ext JS 的一些专用术语	3
1.5.2 一些常见的配置项	4
1.5.3 类的命名规则	5
1.6 关于调试	6
1.7 小 结	6
第 2 章 Ext JS 的 MVVM 模式介绍	7
2.1 概 述	7
2.1.1 MVVM 模式概述	7
2.1.2 Ext JS 的 MVVM 模式	8
2.2 组件查询的机制	8
2.2.1 组件管理器: Ext.ComponentManager	9
2.2.2 组件的查询方式	10
2.2.3 直接使用 id 查询组件	12
2.2.4 组件中的查询	12
2.3 视图控制器	13
2.4 视图模型	14
2.5 小 结	15
第 3 章 简单的 CMS 系统概述	16
3.1 基本功能	16

3.2	系统的主要数据结构.....	16
3.2.1	文章类别表: T_Category	16
3.2.2	文章表: T_Content	19
3.2.3	媒体表: T_Media.....	19
3.2.4	标签表: T_Tag.....	20
3.2.5	标签与文章关联表: T_TagInContent.....	20
3.3	ASP.NET Identity	20
3.4	实体框架代码优先 (Entity Framework Code-First)	20
3.5	系统开发环境.....	21
3.6	小 结.....	21
第4章	使用 Sencha Cmd 创建应用程序	22
4.1	概 述.....	22
4.2	安装 Sencha Cmd	23
4.2.1	运行环境配置.....	23
4.2.2	安装 Sencha Cmd	23
4.2.3	验证安装.....	24
4.2.4	语法.....	26
4.3	创建应用程序.....	26
4.3.1	创建应用程序前要考虑的问题	26
4.3.2	开始创建应用程序	27
4.4	应用程序的结构.....	30
4.4.1	目录结构.....	30
4.4.2	index.html	31
4.4.3	bootstrap.css.....	33
4.4.4	bootstrap.js	33
4.4.5	application.js	33
4.4.6	app.js	34
4.4.7	SimpleCMS.view.main.Main	34
4.4.8	SimpleCMS.view.main.List.....	37
4.4.9	SimpleCMS.view.main.MainController	37
4.4.10	SimpleCMS.view.main.MainModel	38
4.4.11	app.json	38

4.5	自定义样式.....	41
4.6	生成应用程序.....	43
4.7	关于乱码.....	43
4.8	小 结.....	44
第 5 章	搭建开发环境	45
5.1	创建项目.....	45
5.2	团队资源管理.....	46
5.3	添加库文件.....	49
5.3.1	Json.NET.....	49
5.3.2	实体框架.....	50
5.3.3	ImageResizer.....	51
5.3.4	System.Linq.Dynamic	52
5.3.5	基于.NET 的 Web 开发接口 (Open Web Interface for .NET, Microsoft.Owin) ..	52
5.3.6	Microsoft.AspNet.Identity	52
5.3.7	ShortGuid.....	52
5.3.8	Mime-Detective	52
5.3.9	Error Logging Modules and Handlers for ASP.NET (ELMAH)	53
5.3.10	NLog	53
5.4	创建实体模型.....	54
5.4.1	配置数据库连接.....	54
5.4.2	数据库上下文: ApplicationDbContext.....	54
5.4.3	与 Identity 相关的实体类.....	56
5.4.4	文章类别实体.....	58
5.4.5	文章实体.....	60
5.4.6	媒体实体.....	61
5.4.7	标签实体.....	62
5.5	数据迁移.....	63
5.5.1	启用迁移.....	63
5.5.2	调整文章分类表的自增种子值	64
5.5.3	添加触发器.....	64
5.5.4	添加“未分类”类别	66
5.5.5	初始化角色.....	67

5.5.6	创建初始账号.....	67
5.5.7	自动迁移.....	67
5.6	配置 Identity	68
5.7	其他配置.....	70
5.7.1	系统配置.....	70
5.7.2	基础控制器 (BaseController)	71
5.7.3	资源文件.....	73
5.8	添加 Ext JS 应用程序.....	75
5.8.1	导入脚本.....	75
5.8.2	创建首页.....	75
5.8.3	配置 app.json	76
5.8.4	关于智能提示.....	76
5.8.5	简体中文的本地化文件的问题	77
5.8.6	修正中文排序问题	77
5.8.7	修正 Ext.util.Format 的 defaultValue 方法	78
5.8.8	扩展 Ext.form.field.VTypes	78
5.8.9	全局配置类.....	80
5.8.10	访问地址类.....	81
5.8.11	状态管理类.....	85
5.8.12	重新封装的 toast.....	90
5.8.13	本地化类.....	91
5.9	小 结.....	93
第 6 章	数据传输的标准化	94
6.1	标准化的数据传输.....	94
6.1.1	标准化的数据传输是什么	94
6.1.2	为什么要实现数据传输的标准化	94
6.1.3	标准化数据传输的好处	94
6.2	如何实现标准化.....	95
6.2.1	Ext JS 的主要数据传输方式	95
6.2.2	扩展代理.....	96
6.2.3	在服务器端统一输出接口	98
6.2.4	统一的错误处理.....	99

6.3 小 结.....	102
第 7 章 迁移管理模板	103
7.1 管理模板简介.....	104
7.1.1 App.js	104
7.1.2 Application.js	105
7.1.3 主视图: Admin.view.main.Main	105
7.1.4 主容器: Admin.view.main.MainContainerWrap	108
7.1.5 主视图控制器: Admin.view.main.MainController.....	109
7.1.6 要迁移的视图.....	115
7.2 实施迁移.....	115
7.2.1 主视图.....	115
7.2.2 空白视图.....	118
7.2.3 404 视图.....	119
7.2.4 500 视图.....	120
7.2.5 登录视图.....	120
7.2.6 重置密码视图.....	121
7.3 小 结.....	123
第 8 章 登录与权限控制.....	124
8.1 权限控制的基本思路.....	124
8.2 登 录.....	125
8.2.1 验证码.....	125
8.2.2 提交.....	128
8.2.3 使用 Identity 实现登录	129
8.2.4 何时登录.....	132
8.2.5 视图切换.....	136
8.2.6 状态恢复.....	137
8.3 退 出.....	138
8.3.1 退出按钮.....	138
8.3.2 实现退出.....	138
8.4 修改密码.....	139
8.4.1 修改密码按钮.....	139

8.4.2	提交表单.....	139
8.4.3	实现密码修改.....	140
8.5	小 结.....	141
第 9 章	用户管理	142
9.1	概 述.....	142
9.2	用户模型.....	142
9.2.1	概述.....	142
9.2.2	基础模型.....	143
9.2.3	用户模型.....	145
9.3	用户视图、视图模型和视图控制器	146
9.3.1	创建文件.....	146
9.3.2	列定义.....	147
9.3.3	选择模型.....	149
9.3.4	网格的样式.....	149
9.3.5	网格的布局.....	150
9.3.6	工具栏与按钮的 UI	153
9.3.7	通过绑定来启用或禁用按钮	155
9.3.8	缓存存储.....	156
9.3.9	排序.....	158
9.3.10	分页.....	161
9.3.11	简单的查询功能	162
9.3.12	高亮显示搜索值与基础视图控制器	165
9.3.13	复选列的状态.....	166
9.3.14	刷新.....	169
9.3.15	删除.....	169
9.4	用户编辑视图.....	172
9.4.1	基础表单面板.....	172
9.4.2	只读字段的样式.....	181
9.4.3	保存和新建按钮.....	181
9.4.4	用户编辑视图.....	184
9.4.5	获取全局共享窗口和新建用户	186
9.4.6	编辑用户.....	190

9.5 小 结.....	191
第 10 章 媒体管理	192
10.1 概 述.....	192
10.2 媒体模型.....	193
10.3 媒体视图.....	193
10.3.1 媒体视图的定义	193
10.3.2 媒体面板.....	194
10.3.3 数据视图.....	195
10.3.4 数据视图的插件	198
10.3.5 分页工具栏.....	199
10.3.6 在分页工具栏中添加按钮	199
10.3.7 上传按钮.....	200
10.3.8 进度条.....	202
10.3.9 文件上传.....	204
10.3.10 存储.....	206
10.3.11 排序.....	208
10.3.12 查询.....	209
10.3.13 关键字高亮显示	212
10.3.14 修改描述字段.....	213
10.3.15 删除.....	214
10.3.16 查看媒体.....	215
10.4 媒体选择窗口.....	216
10.5 小 结.....	219
第 11 章 文章管理	220
11.1 概 述.....	220
11.2 模 型.....	220
11.2.1 文章类别模型.....	220
11.2.2 文章模型.....	221
11.2.3 标签模型.....	221
11.3 文章视图.....	222
11.3.1 整体布局.....	222

11.3.2	列表视图.....	222
11.3.3	边框布局.....	223
11.4	文章类别树.....	224
11.4.1	树存储.....	224
11.4.2	树面板.....	226
11.4.3	文章类别的操作按钮	226
11.4.4	删除文章类别.....	227
11.4.5	刷新文章类别.....	229
11.5	文章类别编辑视图.....	229
11.5.1	基本定义.....	229
11.5.2	返回按钮.....	230
11.5.3	布局.....	230
11.5.4	字段.....	231
11.5.5	按钮.....	232
11.5.6	类别下拉列表字段	232
11.5.7	题图选择字段.....	234
11.5.8	TinyMCE 字段	235
11.5.9	视图切换.....	239
11.5.10	新建文章类别.....	241
11.5.11	编辑文章类别.....	243
11.5.12	完善父类别选择功能	245
11.5.13	返回操作.....	248
11.6	详细信息视图.....	248
11.7	文章列表.....	251
11.7.1	基本定义.....	251
11.7.2	存储.....	252
11.7.3	文章类别默认选择全部文章节点	253
11.7.4	根据文章类别控制文章列表的标题和数据	254
11.7.5	文章查询.....	255
11.7.6	单击标题列的单元格切换到详细信息视图	257
11.7.7	删除文章.....	258
11.7.8	刷新文章.....	259

11.7.9 通过拖动改变文章类别	259
11.8 文章编辑视图.....	262
11.8.1 字段.....	262
11.8.2 完善文章类别的选择功能	264
11.8.3 标签字段.....	265
11.8.4 返回操作.....	267
11.8.5 新建文章.....	267
11.8.6 编辑文章.....	269
11.9 标签管理窗口.....	269
11.9.1 定义窗口.....	269
11.9.2 通过字段值控制按钮	271
11.9.3 存储.....	271
11.9.4 新建标签.....	272
11.9.5 删除标签.....	273
11.9.6 在标签字段中实现新建标签	274
11.9.7 在文章列表的工具栏添加标签管理按钮	274
11.10 小 结.....	275
第 12 章 打包和发布.....	276
12.1 打 包.....	276
12.1.1 统一处理服务器错误	276
12.1.2 创建应用程序的测试包	277
12.1.3 在 IIS 新建一个应用程序.....	277
12.1.4 复制文件.....	278
12.1.5 修改首页文件.....	278
12.1.6 配置数据库连接	279
12.1.7 开始调试.....	279
12.2 发 布.....	279
12.2.1 生成应用程序.....	279
12.2.2 复制应用程序.....	279
12.2.3 cache.appcache 文件.....	279
12.2.4 生成解决方案.....	279
12.2.5 发布前的测试.....	281

12.2.6	第一次部署时所要做的配置	281
12.2.7	持续更新	281
12.3	小 结	281
第 13 章	Ext JS 的跨平台特性	282
13.1	Ext JS 跨平台特性简介	282
13.2	在 Java 平台使用 SimpleCMS	282
13.2.1	概述	282
13.2.2	搭建开发环境	283
13.2.3	添加应用程序	283
13.2.4	创建首页文件	283
13.2.5	修改 app.json 文件	283
13.2.6	修改 SimpleCMS.util.Url	283
13.2.7	添加 Json-lib	283
13.2.8	辅助类 ExtJs	284
13.2.9	创建 Servlet: GetUserInfo	285
13.3	在 PHP 平台使用 SimpleCMS	286
13.3.1	搭建开发环境	286
13.3.2	在 XAMPP 设置访问路径	286
13.3.3	配置 Yii 框架	287
13.3.4	添加应用程序	287
13.3.5	修改首页文件	287
13.3.6	辅助类 ExtJs	287
13.3.7	创建 AccountController 控制器	288
13.4	小 结	289

第1章 Ext JS 6概述

出乎很多人的预料，在 Ext JS 5 刚刚发布不久，Ext JS 6 就这样悄然无声地出现了。一切都是那么的突然，这应该是和 Sencha 公司的战略有关。Ext JS 5 还是一个与 Sencha Touch 分离的版本，如果想使用 Sencha 的产品开发一个跨平台的产品，实现不同的平台使用不同的界面，就需要使用 Ext JS 和 Sencha Touch 分别进行开发，而现在的潮流，如 Bootstrap 这样的框架，都是以响应式布局来实现不同平台的无缝开发，Ext JS 当然也不能忽视这种潮流。

1.1 从 Ext JS 4 到 Ext JS 5

要问在 Ext JS 4 中最大的问题是什么？估计很多开发人员都会回答：控制器。虽然在 Ext JS 4 中引入了控制器以完善三层开发模式，但 4 中的控制器太死板，一经加载，就不能销毁，而且只能加载一次。这样，问题来了，我要在不同的地方创建两个相同的视图用于显示不同的数据怎么办？为了更好地解决这个问题，在 Ext JS 5 中，将 MVC 模型修改为 MVVM 模型，修改后的 MVVM 模型带来的好处是控制器终于可以跟随视图创建或销毁了。

在 MVVM 模型中，另一个值得称道的功能是绑定。例如，在 5 之前的版本中，要根据网格（Ext.grid.Panel）是否已经选择了数据来判断是否启用或禁用按钮，基本的实现思路是监听网格的选择改变（selectionchange）事件，然后根据选择结果来调整按钮状态。使用绑定，这个就很简单了，只要将网格的属性 selection 绑定到数据对象 selection，再将按钮的 disabled 属性与数据对象 selection 绑定就行了。可以说，在 Ext JS 6 的开发过程中，肯定会用到绑定，除非是坚持不用的。

为了支持通用应用程序开发，在 Ext JS 5 中引入了响应式布局，也就是可以根据设备及设备的方向改变而改变应用程序的位置或元素的布局。

在组件方面的改进主要是增强了网格，可以在网格内显示小部件。另一个较大的改变是图表，这次终于与 D3 包一样，使用画布（Canvas）来绘制图表，而不是使用 SVG 这类技术了。

Ext JS 5 还有一个重点就是将 Ext JS 和 Sencha Touch 的核心包融合在一起，以便将两套系统合并在一起。

1.2 从 Ext JS 5 到 Ext JS 6

从 Ext JS 5 到 6 的最大变化就是把 Ext JS 和 Sencha Touch 合二为一了，也就是可以开发出既适用于桌面又适用于移动设备的通用应用程序了。

合并之后，Ext JS 被命名为经典（classic）工具包，而 Sencha Touch 则被命名为现代（modern）工具包。在应用程序中，也在原来的目录结构上添加了 classic 和 modern 两个目录来分别存放使用不同工具包创建的视图。

在 Ext JS 6 中，还加入了一个名为海卫一（Triton）的主题。它的最大特点是可以使用 Font Awesome 字体作为图标，对于我们这些不会创建图标的程序员，终于摆脱了满大街找图标的困境。

1.3 从 Ext JS 6.0 到 Ext JS 6.2

Ext JS 6.2 的发布有点奇怪，居然不是先由付费的高端（Premium）客户使用一段时间，出了新版之后再发布 GPL 版本，而是直接把 GPL 版本发送给了全部客户，估计也是为了测试功能而实行的策略，现在的高端客户使用的已经是 6.2.1 版了，而 GPL 版的还是 6.2。

在 6.2 版本，最大受益者应该是高端客户，因为在这版本中为他们添加了日历包、D3 适配器组件、功能更强的透视表格（Pivot Grid）和导出功能。

对于非高端客户来说，使用 6.2 的唯一动机只能是对 6.0 的臭虫修复了，添加的功能实在是寥寥无几。但旧有的臭虫修复后，带来的是更令人恼怒的臭虫，主要有两个：一是默认使用了适用于阿拉伯等地区习惯的文字右对齐的模式，二是使用 Admin 模板会出错，须自己重写一下类 Ext.list.RootTreeItem¹。

总的来说，6.2 只是高端客户的茶。

1.4 如何选择版本

基于上面几节的版本说明，大家对 Ext JS 5 或 6 的各版本都有了一定的了解，要做出选择也不会太难。

对于新项目，应该毫不犹豫地选择最新版本，这样才能更好地随着 Ext JS 的升级与更新而获得更新，主要目的就是减少 Ext JS 臭虫对项目的影响。不过，最大的风险还是后续版本的兼容性问题，这个是大家都控制不了的。

那么，对于新项目，选择 6.0 还是 6.2 呢？相对来说，6.0 是相对稳定的一个版本，而 6.2 感觉就是一个为高端用户做的测试版。对于初学者或初次用来做开发的用户，建议使用稳定的 6.0 版本，因为这样可以尽可能地避免臭虫的影响，不然，类似文字右对齐这样的错误，可能会束手无策；而对于已经有相当经验的开发人员来说，使用最新版是最好的选择，这样不但可以摆脱前期版本的臭虫，还可以尽可能地利用新的功能。

对于已经在使用 4.2 之前版本开发的项目或正在开发的项目，如果旧版本暂时能满足需求，臭虫的影响也不大，那么可以作为阶段性任务，先使用旧版本完成项目，等项目成功交付后，即可着手下一阶段任务，考虑将项目整体迁移到最新版本。迁移的目的是为了在未来的项目升级中减少臭虫或其他方面的影响。

¹ 具体信息请参考：<https://www.sencha.com/forum/showthread.php?325716-Tree-List-getting-error-as-me-getFloated-is-not-a-function-in-AbstractTreeItem-js&s=3f23fd280155dfd780574c641fb4992e&p=1166528#post1166528>