

Build Ajax applications with Ext JS

A JavaScript framework for rich Internet application development

Skill Level: Intermediate

[John W. Fronckowiak \(john@idcc.net\)](mailto:john@idcc.net)

President

IDC Consulting, Inc.

01 Jul 2008

Ext JS is a powerful JavaScript library that simplifies Asynchronous JavaScript + XML (Ajax) development through the use of reusable objects and widgets. This article introduces Ext JS, providing an overview of the object-oriented JavaScript design concepts behind it, and shows how to use the Ext JS framework for rich Internet application UI elements.

With the wide variety of Web development frameworks available today, it's difficult for developers to determine which are worthy of their time. Ext JS, a JavaScript development framework, stands out as a tool that Web application developers should seriously consider — a powerful JavaScript library that simplifies Ajax development through the use of reusable objects and widgets. Ext JS started as a group of extensions to the Yahoo! User Interface (YUI) Library by Jack Slocum. With the recent release of version 2.0, however, it has become one of the simplest and most powerful JavaScript libraries on the market.

Get to know Ext JS

Ext JS began as a project to extend the functionality that the YUI Library offered. A key aspect of the YUI Library is the cross-browser support, which you'll also find in Ext JS. This support allows you to build Web applications without worrying about the target browser.

Ext JS provides excellent performance. The framework is fully object oriented and extensible. Because it's written in the JavaScript language, Ext JS's features are ready to use after you download and install it.

Licensing

Before you adopt a new framework, it's important to understand the licensing terms under which the framework has been released. Ext JS provides several licensing options:

- **Open source license:** Under the terms of the Open Source LGPL 3.0 license, this is the most appropriate license if you plan to use Ext JS in another open source project or in a personal, educational, or not-for-profit project.
- **Commercial license:** This license would be most appropriate if you plan to use Ext JS in a project in which you don't want any potential open source license restrictions, you must own a license for internal reasons, or you want to support Ext JS development financially. Take a look at the [Resources](#) section for a link to the Ext JS site, which provides information on rates for commercial licenses.
- **Original equipment manufacturer (OEM)/reseller license:** This license would be most appropriate if you plan to repackage or sell Ext JS as a software development library.

Ext JS browser support

The Ext JS framework is supported in all the major Web browsers, including:

- Windows® Internet Explorer® version 6 and later.
- Mozilla Firefox version 1.5 and later (PC and Macintosh).
- Apple Safari version 2 and later.
- Opera version 9 and later (PC and Mac).

Design patterns and Ext JS

Developers should appreciate the well-thought-out design and implementation of Ext JS. Its object-oriented design patterns influence the relationship and interactions between objects. According to Ext JS, the design patterns used in their development were highly influenced by the book, *Head First Design Patterns*, by Freeman and Freeman (see [Resources](#)). Developers looking at the Ext JS source code will find creational patterns, including the singleton design pattern; structural patterns, including the flyweight design pattern; and behavioral patterns, including the observer pattern.

Build rich Internet applications with Ext JS

Ext JS provides numerous UI elements that are essential to developing rich Internet applications (RIAs). Ext JS includes controls such as message boxes, combo boxes, data grids, and toolbars. In addition, layout managers allow you to specify how elements are displayed on a page. Additional features are available for working with forms and windows.

The include order for the JavaScript files can change if you are using other frameworks. However, Ext JS is typically included in your Web application, as shown in [Listing 1](#), assuming that you've installed Ext JS in the lib/ext directory on your Web server:

Listing 1. Include the Ext JS framework

```
<script type="text/javascript" src="lib/ext/ext-base.js"></script>
<script type="text/javascript" src="lib/ext/ext-all.js"></script>
```

The ext-all.js file includes the entire Ext JS framework. You can optionally choose to reference the file as listed, or you may opt to include only the files necessary for the elements used in your application.

Integration of Ext JS

You can use Ext JS with other common Web development server-side frameworks, including PHP, the Java™ language, Microsoft® .NET, Ruby on Rails, and ColdFusion.

If you're using the Ext JS framework with other JavaScript libraries, the file INCLUDE_ORDER.txt, which is installed in the root installation directory, indicates the order in which you should include the libraries in your application.

UI elements

The heart of the Ext JS framework is the multitude of rich UI elements provided. These elements include forms, dialog boxes, tabs, trees, and grids.

Forms

Ext JS provides you with a rich set of tools to create interactive forms. [Figure 1](#) shows an example form. [Listing 2](#) shows its associated implementation.

Figure 1. Example Ext JS form

Multi Column, Nested Layouts and Anchoring

First Name:

Last Name:

Company:

Email:

Biography:

B I U A ab [List Icons]

Listing 2. Example Ext JS form source code

```
var top = new Ext.FormPanel({
    labelAlign: 'top',
    frame:true,
    title: 'Multi Column, Nested Layouts and Anchoring',
    bodyStyle:'padding:5px 5px 0',
    width: 600,
    items: [{
        layout:'column',
        items:[{
            columnWidth:.5,
            layout: 'form',
            items: [{
                xtype:'textfield',
                fieldLabel: 'First Name',
                name: 'first',
                anchor:'95%'
            }, {
                xtype:'textfield',
                fieldLabel: 'Company',
                name: 'company',
                anchor:'95%'
            }
        ]
    }]
})
```

```

    }, {
      columnWidth: .5,
      layout: 'form',
      items: [{
        xtype: 'textfield',
        fieldLabel: 'Last Name',
        name: 'last',
        anchor: '95%'
      }, {
        xtype: 'textfield',
        fieldLabel: 'Email',
        name: 'email',
        vtype: 'email',
        anchor: '95%'
      }
    ]
  }, {
    xtype: 'htmleditor',
    id: 'bio',
    fieldLabel: 'Biography',
    height: 200,
    anchor: '98%'
  }],
  buttons: [{
    text: 'Save'
  }, {
    text: 'Cancel'
  }
],
});

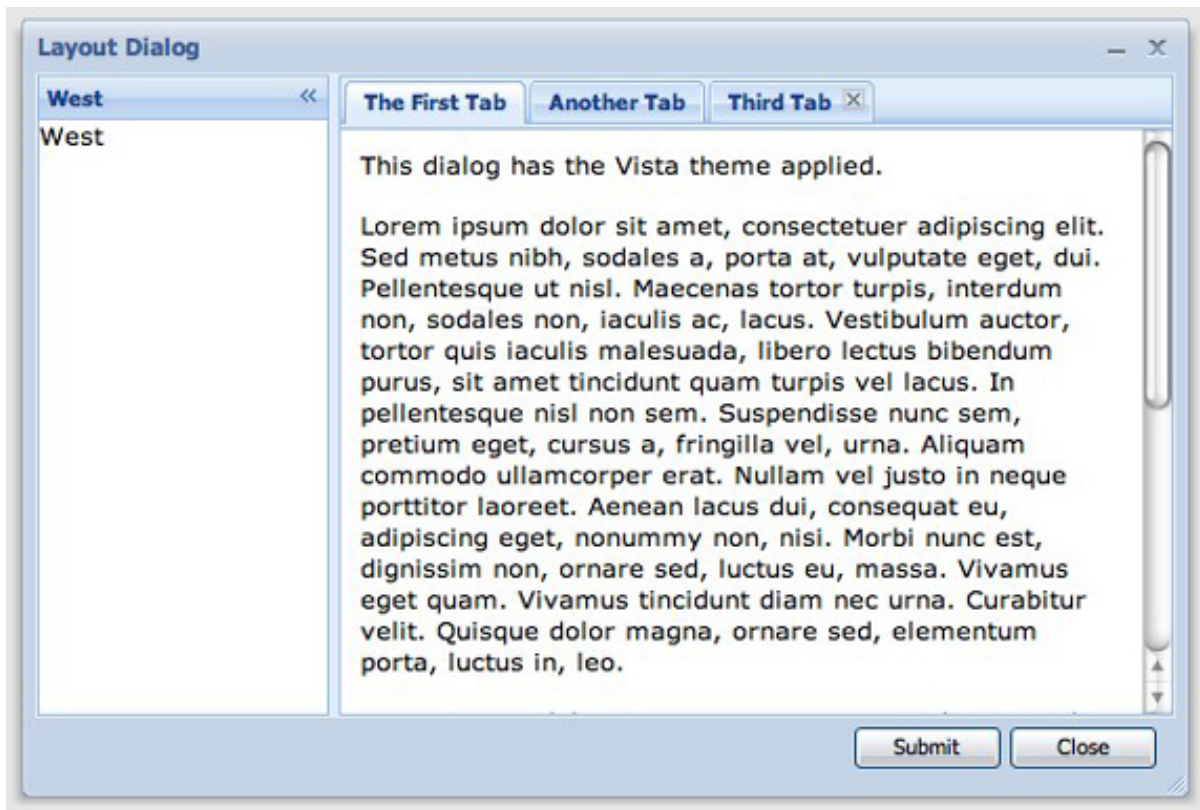
top.render(document.body);

```

Dialog boxes and tabs

As shown in [Figure 2](#), Ext JS provides you with the ability to create modal dialog boxes for user input as well as the ability to implement a tabbed UI to maximize screen real estate. The source code for the dialog box in [Figure 2](#) is shown in [Listing 3](#).

Figure 2. Ext JS modal dialog box with tabs



Listing 3. Ext JS modal dialog box with tabs source code

```
var LayoutExample = function(){
// everything in this space is private and only accessible in the HelloWorld block

// define some private variables
var dialog, showBtn;

var toggleTheme = function(){
    Ext.get(document.body, true).toggleClass('xtheme-gray');
};
// return a public interface
return {
    init : function(){
        showBtn = Ext.get('show-dialog-btn');
        // attach to click event
        showBtn.on('click', this.showDialog, this);
    },

    showDialog : function(){
        if(!dialog){ // lazy initialize the dialog and only create it once
            dialog = new Ext.LayoutDialog("hello-dlg", {
                modal:true,
                width:600,
                height:400,
                shadow:true,
                minWidth:300,
                minHeight:300,
                proxyDrag: true,
                west: {
                    split:true,
                    initialSize: 150,
                    minSize: 100,
```

```

        maxSize: 250,
        titlebar: true,
        collapsible: true,
        animate: true
    },
    center: {
        autoScroll: true,
        tabPosition: 'top',
        closeOnTab: true,
        alwaysShowTabs: true
    }
});
dialog.addKeyListener(27, dialog.hide, dialog);
dialog.addButton('Submit', dialog.hide, dialog);
dialog.addButton('Close', dialog.hide, dialog);

var layout = dialog.getLayout();
layout.beginUpdate();
layout.add('west', new Ext.ContentPanel('west', {title: 'West'}));
layout.add('center', new Ext.ContentPanel('center', {title: 'The First Tab'}));
// generate some other tabs
layout.add('center', new Ext.ContentPanel(Ext.id(), {
    autoCreate: true, title: 'Another Tab', background: true}));
layout.add('center', new Ext.ContentPanel(Ext.id(), {
    autoCreate: true, title: 'Third Tab', closable: true, background: true}));
layout.endUpdate();
}
dialog.show(showBtn.dom);
}
};
}();

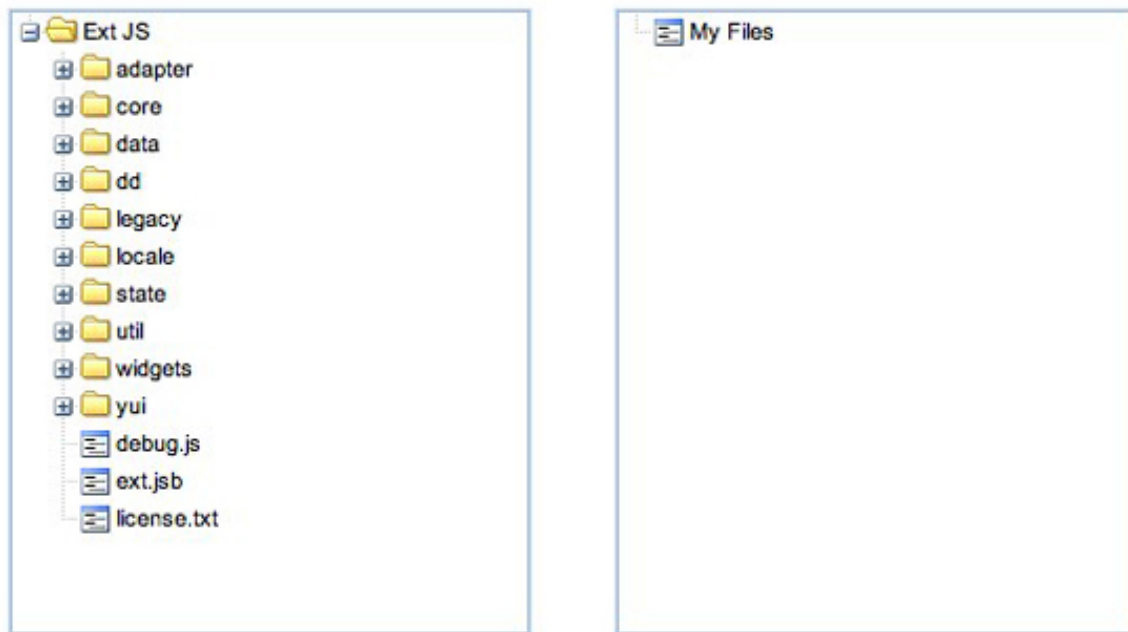
// using onDocumentReady instead of window.onload initializes the application
// when the DOM is ready, without waiting for images and other resources to load
Ext.EventManager.onDocumentReady(LayoutExample.init, LayoutExample, true);

```

Creating trees

As shown in [Figure 3](#), Ext JS also provides tree controls, which provide users with a familiar file system-like view. Ext JS tree controls support full drag-and-drop functionality. The source code for the tree controls shown in [Figure 3](#) appears in [Listing 4](#).

Figure 3. Ext JS tree controls



Listing 4. Ext JS tree controls source code

```
var TreeTest = function(){
// shorthand
var Tree = Ext.tree;

return {
  init : function(){
    // yui-ext tree
    var tree = new Tree.TreePanel({
      el:'tree',
      animate:true,
      autoScroll:true,
      loader: new Tree.TreeLoader({dataUrl:'get-nodes.php'}),
      enableDD:true,
      containerScroll: true,
      dropConfig: {appendOnly:true}
    });

    // add a tree sorter in folder mode
    new Tree.TreeSorter(tree, {folderSort:true});

    // set the root node
    var root = new Tree.AsyncTreeNode({
      text: 'Ext JS',
      draggable:false, // disable root node dragging
      id:'source'
    });
    tree.setRootNode(root);

    // render the tree
    tree.render();

    root.expand(false, /*no anim*/ false);

    //-----
```



```

// YUI tree
var tree2 = new Tree.TreePanel({
    el:'tree2',
    animate:true,
    autoScroll:true,
    loader: new Ext.tree.TreeLoader({
        dataUrl:'get-nodes.php',
        baseParams: {lib:'yui'} // custom http params
    }),
    containerScroll: true,
    enableDD:true,
    dropConfig: {appendOnly:true}
});

// add a tree sorter in folder mode
new Tree.TreeSorter(tree2, {folderSort:true});

// add the root node
var root2 = new Tree.AsyncTreeNode({
    text: 'My Files',
    draggable:false,
    id:'yui'
});
tree2.setRootNode(root2);
tree2.render();

root2.expand(false, /*no anim*/ false);
}
};
}();

Ext.EventManager.onDocumentReady(TreeTest.init, TreeTest, true);

```

Grids

Perhaps one of the most powerful Ext JS UI elements is the grid control. It allows you to display data from a back-end data source in addition to other structured data such as XML and arrays. As shown in [Figure 4](#), Ext JS grids can implement paging and column sorting. This example highlights the Ajax capabilities of the Ext JS framework, retrieving the latest ExtJS.com forum topics. The source code for the grid in Figure 4 appears in [Listing 5](#).

Figure 4. Ext JS grid control

ExtJS.com - Browse Forums		
Topic	Replies	Last Post
How to deal with single quote symbol inside a name in a JSON String Premium Help Forum I have a business logic inside a pageflow (action class) which creates a JSON object and sets in the form. So I retrieve the JSON Object (String) from the form to populate a data grid on the jsp. But in the data, there are fields where there's a chance of single quotes (" ") inside it, say name...	0	Feb 18, 2008, 3:36 pm by jayghodasara
response has no properties Help Forum handler: function() { frmContact.form.submit({ url: 'mypage', success: function(frm, response) { var result = Ext.util.JSON.decode(response.responseText); var vlcon = Ext.MessageBox.INFO; Ext.MessageBox.show({title: 'Contact', msg: result.response,...	3	Feb 18, 2008, 3:35 pm by devnull
Is the Ext desktop useful Open Discussion Forum I've been developing Ext for a short while now, and so far I've steered clear of the Desktop. I mean, it's pretty and all that, but other than showing off what can be done, it is really useful in a business application. It won't replace the user's desktop, and it would be sheer murder to place...	0	Feb 18, 2008, 3:33 pm by crafter
applyTo overwrites "id" ? Help Forum Hello! I'm trying to get ExtJS to create a Panel over some existing html content. Here's an example: As you can see, I am assigning an id to that markup already. Here's the ExtJS code: var mainwindow_panel = new Ext.Panel({ applyTo: 'mainwindow_div', //id: 'mainw...	0	Feb 18, 2008, 3:21 pm by moose_
Accordion layout behaviour Premium Help Forum Dear community, I've designed a viewport with an accordion panel in the west part. Each item of the accordion panel contains a tree panel with tree nodes. Is there any convenient way to maximize the height of the expanded (clicked) item in the accordion panel? The behaviour would be the same ...	7	Feb 18, 2008, 3:20 pm by ECU

Listing 5. Ext JS grid control source code

```

Ext.onReady(function(){
    // create the Data Store
    var store = new Ext.data.Store({
        // load using script tags for cross domain, if the data in on the same domain as
        // this page, an HttpProxy would be better
        proxy: new Ext.data.ScriptTagProxy({
            url: 'http://extjs.com/forum/topics-browse-remote.php'
        }),

        // create reader that reads the Topic records
        reader: new Ext.data.JsonReader({
            root: 'topics',
            totalProperty: 'totalCount',
            id: 'threadid',
            fields: [
                'title', 'forumtitle', 'forumid', 'author',
                {name: 'replycount', type: 'int'},
                {name: 'lastpost', mapping: 'lastpost', type: 'date',
                    dateFormat: 'timestamp'},
                'lastposter', 'excerpt'
            ]
        }),

        // turn on remote sorting
        remoteSort: true
    });
    store.setDefaultSort('lastpost', 'desc');

```

```

// pluggable renders
function renderTopic(value, p, record){
    return String.format(
        '<b><a href="http://extjs.com/forum/showthread.php?t={2}"
        target="_blank">{0}</a></b>
        <a href="http://extjs.com/forum/forumdisplay.php?f={3}"
        target="_blank">{1} Forum</a>',
        value, record.data.forumtitle, record.id, record.data.forumid);
    }
function renderLast(value, p, r){
    return String.format('{0}<br/>by {1}', value.dateFormat('M j, Y, g:i a'),
        r.data['lastposter']);
    }

// the column model has information about grid columns
// dataIndex maps the column to the specific data field in
// the data store
var cm = new Ext.grid.ColumnModel([
    {
        id: 'topic',
        header: "Topic",
        dataIndex: 'title',
        width: 420,
        renderer: renderTopic
    },
    {
        header: "Author",
        dataIndex: 'author',
        width: 100,
        hidden: true
    },
    {
        header: "Replies",
        dataIndex: 'replycount',
        width: 70,
        align: 'right'
    },
    {
        id: 'last',
        header: "Last Post",
        dataIndex: 'lastpost',
        width: 150,
        renderer: renderLast
    }
]);

// by default columns are sortable
cm.defaultSortable = true;

var grid = new Ext.grid.GridPanel({
    el: 'topic-grid',
    width: 700,
    height: 500,
    title: 'ExtJS.com - Browse Forums',
    store: store,
    cm: cm,
    trackMouseOver: false,
    sm: new Ext.grid.RowSelectionModel({selectRow: Ext.emptyFn}),
    loadMask: true,
    viewConfig: {
        forceFit: true,
        enableRowBody: true,
        showPreview: true,
        getRowClass: function(record, rowIndex, p, store){
            if(this.showPreview){
                p.body = '<p>'+record.data.excerpt+'</p>';
                return 'x-grid3-row-expanded';
            }
            return 'x-grid3-row-collapsed';
        }
    },
    bbar: new Ext.PagingToolbar({
        pageSize: 25,

```

```

    store: store,
    displayInfo: true,
    displayMsg: 'Displaying topics {0} - {1} of {2}',
    emptyMsg: "No topics to display",
    items: [
        '-', {
            pressed: true,
            enableToggle: true,
            text: 'Show Preview',
            cls: 'x-btn-text-icon details',
            toggleHandler: toggleDetails
        }
    ]
  });
});

// render it
grid.render();

// trigger the data store load
store.load({params:{start:0, limit:25}});

function toggleDetails(btn, pressed) {
    var view = grid.getView();
    view.showPreview = pressed;
    view.refresh();
}
});

```

Ext JS and Ajax

The Ext JS framework includes support for Ajax implementations. Typically, a common feature of Ajax applications is for an application to asynchronously respond to user input by updating the UI without redisplaying the entire Web page. [Listing 6](#) shows a typical Ext JS Ajax implementation: an HTML text field and button element that posts data in the text field to a Web server when the button is clicked.

Listing 6. Include the Ext JS framework

```

<script type="text/javascript">
Ext.onReady(function(){
    Ext.get('okButton').on('click', function(){
        var msg = Ext.get("msg");
        msg.load({
            url: [server url], // <-- replace with your url
            params: "name=" + Ext.get('name').dom.value,
            text: "Updating..."
        });
        msg.show();
    });
});
</script>

<div id="msg" style="visibility: hidden"></div>
Name: <input type="text" id="name" /><br />
<input type="button" id="okButton" value="OK" />

```

When the user clicks **OK**, the Ext JS `UpdateManager` class is used to make the Ajax call, which is a great simplification of a typical Ajax `HttpRequest` call.

Ext JS integration with other Web server frameworks

You can use Ext JS with other common Web development server-side frameworks, including PHP, the Java language, Microsoft .NET, Ruby on Rails, and ColdFusion. For integration specifics for each of these frameworks, see [Resources](#).

Ext JS development tools

You can integrate Ext JS framework development into several popular integrated development environments (IDEs), including Eclipse, Aptana, and Komodo. For information about including Ext JS development support in your IDE of choice, see [Resources](#).

The developerWorks Ajax resource center

Check out the [Ajax resource center](#), your one-stop shop for free tools, code, and information on developing Ajax applications. The [active Ajax community forum](#), hosted by Ajax expert Jack Herrington, will connect you with peers who might just have the answers you're looking for right now.

Conclusion

Web development frameworks often promise to simplify and speed application development, but many fall short of that goal. Ext JS keeps its promise with an easy-to-use development model. The latest release of Ext JS —version 2.0— shows that it is committed to evolving and remaining a cornerstone of RIA development.

This article introduced the major features of the Ext JS framework, but there is certainly much more. Now, it's your turn to explore more deeply, starting at the ExtJS.com Web site and the [interactive examples](#)!

Resources

Learn

- [Tutorial: Introduction to Ext](#): This step-by-step guide helps get you started with Ext JS.
- [Ext JS documentation](#): Read the version 2.0 API documentation.
- Get more information on the [commercial licenses](#) available for Ext JS.
- "[Designing composite applications: Design patterns](#)": Discover more about design patterns. (Jo Grant and Craig Wolpert, developerWorks, February 2008).
- [Head First Design Patterns](#): Read the book by Eric Freeman, et al., that influenced Ext JS version 2.0 (O'Reilly, 2004).
- [Mastering Ajax](#): Read this developerWorks series for a comprehensive overview of Ajax.
- [Server-side frameworks for Ext JS](#): Discover integration specifics for frameworks such as Python, PHP, and Microsoft .NET.
- [Technology bookstore](#): Browse for books on these and other technical topics.
- [IBM technical events and webcasts](#): Stay current with developerWorks' Technical events and webcasts.

Get products and technologies

- [Ext JS version 2.0 framework](#): Download the framework.
- [IBM® trial software](#): Innovate your next open source development project with IBM trial software, available for download or on DVD.

Discuss

- [Ext JS forums](#): Participate in the Ext JS community.
- [Ext JS blogs](#): The blogs should be your first stop for the latest news and information about Ext JS.
- [developerWorks blogs](#): Participate in developerWorks blogs and get involved in the developerWorks community.

About the author

John W. Fronckowiak

John Fronckowiak is president and founder of [IDC Consulting Inc.](#), and he is also a Clinical Assistant Professor in Information Systems at [Medaille College, at the School](#)

[of Adult and Graduate Education](#). He is the author of several books and articles about Web application development, programming, database design and development, and networking.