Harbinger Group

Five JavaScript Frameworks

A Point-by-point Comparison

A Harbinger Group White Paper

Harbinger Group

Your partner in technology innovation

About Harbinger Group: Harbinger Group (www.harbingergroup.com) is a leading provider of software engineering services to some of the world's best product companies.

This white paper is part of Harbinger's Web 2.0 practice. Harbinger's Web 2.0 practice provides software development and testing services for the rich user interface and collaboration needs of enterprise businesses and social web communities. Visit our Web 2.0 Practice page to learn more.

Visit our website to download or request our white papers on leading edge technologies and trends.

Contents

Introduction

There are a multitude of JavaScript frameworks available today for programming rich client-side interactions in web applications. With many such different options, it is important to choose a framework that not only serves your current requirement, but also serves you in the long run. In this paper, we draw up a visual comparison between Ext JS and four other popular JavaScript frameworks: Prototype, Dojo, JQuery and YUI. In addition to feature-set and technical comparisons, we also look at parameters such as community support, design principles and quality of documentation and samples to provide a sense of the framework's durability. We will then explore Ext JS in some detail and identify some of its salient features.

A Word on the Comparison Attributes

We chose the comparison attributes based on the key goals that JavaScript frameworks need to achieve in a web application. Following are some of the key attributes that we based our comparison on:

❖ Usability features that reduce user clicks and provide an intuitive interface

Needless to say, being client-side technologies, these frameworks need to model user interactions extremely well. For example, features such as layout management, look and feel, consistency across browsers, all contribute immensely to a successful web application and these frameworks need to stand up to the challenge.

❖ Quality and extensibility of widget library and framework objects

Having a rich library of widgets is important, as it indicates the framework's commitment to developers. However, few widgets of any complexity are used in applications exactly the way they exist in the library. Hence, the ability and ease of extending and customizing them is a critical factor in the framework's practical usefulness.

❖ Scripting capabilities including data modeling and ease of event and error handling

Frameworks and technologies that adhere to solid design principles and intuitive scripting constructs have a much better chance of going a long way than the ones that have awkward constructs or have design shortcuts. Good modeling ability and intuitive structures breed a community of developers. Most importantly, it also means that when the relevant part of the industry is ready for its next big upheaval, this framework is the one that it relies heavily on.

❖ Performance parameters such as speed and binary size

With the other promises that the frameworks have to live up to, tradeoffs such as ones between size and speed or design elegance and performance kick in and become a big challenge. The appropriate balance between these tradeoff elements again defines the practicality and acceptance of the framework.

❖ Integration capabilities, availability of support and developer network

A framework's interoperability with other technologies, and sometimes even competing frameworks, plays an important role in its success and acceptance. Elements like quality of documentation, availability of support and the nurturing of a developer community around the framework are all features that create immense value and provide an ecosystem for the framework to thrive in.

Five JavaScript Frameworks

Figure 1 shows the comparison of the five JavaScript frameworks that we researched along these important factors.

		Ext JS	Prototype	Dojo	JQuery	YUI
1	Event handling	✓	✓	✓	✓	✓
2	AJAX support	✓	✓	✓	✓	✓
3	Namespaced API and Namespacing	✓		✓	✓	✓
4	Cross-site scripting*	✓		✓	✓	✓
5	IFrame I/O is mostly used for file upload			✓		
6	Custom HTTP request headers	✓	✓	✓	✓	✓
7	Error handling	✓	✓	✓	✓	✓
8	Integration with other JS libraries†	✓			✓	
9	Synchronous AJAX calls		✓		✓	
		Ext JS	**Prototype**	**Dojo**	**JQuery**	**YUI**
10	Widget collection	****	**	****	**	***
11	Refined UI effect examples	****	***	***	**	***
12	Client side data model	****	*	***	*	***
13	Overall modeling of complex UI interactions	***	**	**	****	**
14	Performance (overall)	****	*	***	***	**
15	Minimal learning curve	***	***	**	****	***
16	Ease of use (API)	***	***	**	***	***
17	Browser support	***	**	***	***	***
18	Documentation	***	***	**	****	****
19	Developer community	***	***	**	**	***
20	File size (KB)	150 to 500	40 to 140	50 to 280	10 to 50	30 to 300

Framework feature comparisons are as of 26-Feb, 2009
* Cross-site scripting is possible in Prototype through plugins, but there are limitations to their use
† Integration with other JS libraries is much smoother in Ext JS and JQuery compared to the other three

Figure 1: Comparison of five JavaScript frameworks

Ext JS and JQuery come out the winners with respect to their support of basic technical features. Both are upcoming frameworks that show commitment towards robustness of design and conceptual modeling that is needed for serious web applications. Let's look at some of the salient features of Ext JS:

❖ Ext JS has a rich library of widgets that work right out of the box on different browsers. It comes with cross-browser DOM, due to which the widgets look exactly the same in most of the popular browsers. These widgets are also very extensible due to a robust component model and Ext JS's good use of design patterns.

❖ Ext JS and JQuery, both integrate with other JavaScript libraries through the use of appropriate adapters. This makes them good contenders for your existing applications built using another framework.

❖ It provides a "build your own Ext JS"* feature that allows you to build your own custom ext.js file with only the components you need. Since the size of Ext JS binaries can be higher than most other frameworks, this is especially useful. In addition, techniques like compression and Ext JS's Content Delivery Network† are also used to good effect in managing its binary size.

❖ Ext JS supports a robust client-side data model. This makes its design scalable and working with such things as JSON formats much easier. It enables developers to build serious applications, and is a vital aspect that few other JavaScript libraries, including JQuery, can claim.

EXT JS QUICK FACTS

✓ Rich library of cross-browser compatible widgets

✓ "Build your own Ext JS"

✓ Integrates with other JavaScript libraries

✓ Component model with support for extensions and plugins

✓ Over 20 design patterns, including Flyweight, Façade, Visitor and Observer

✓ Supports AJAX, DHTML and DOM scripting

✓ Hosted on CacheFly, a free Content Delivery Network (CDN)

✓ Commercial and open source licenses (under GPL v3 with FLOSS exception)

✓ Supports IE (6.0+), FireFox (1.5+), Safari (3.0+), Opera (9.0+) and Google Chrome

❖ On performance parameters such as speed and responsiveness, Ext JS emerges a clear winner among its peers.

❖ Ext JS has a strong developer community and example usages that ease the use of its rich widget library.

❖ Ext JS also fairs well in its overall modeling of complex user interactions, although JQuery scores somewhat higher in this respect.

---

* Build your own Ext JS - http://extjs.com/products/extjs/build/
† Ext JS is hosted on its own Content Delivery Network through a partnership with CacheFly (http://cachefly.com)

Your Choice of JavaScript Framework

Your initial choice of JavaScript framework is an important one and the differences between these frameworks can be subtle. Being client-side technologies, these frameworks handle multiple priorities like usability, scripting ease, support for browsers, performance and many others. Ext JS is emerging as an "industry-strength" framework and is being increasingly used in the enterprise. It is among the best performing frameworks. Although at first glance, the binary size of Ext JS comes out to be on the higher side, it is partly due to its rich library and can be effectively managed through intelligent use of compression, CDN and custom builds. Ext JS comes in both open source and commercial licensing models, and it is important to study the fine print to understand which one suits your need.

Visit https://www.harbinger-systems.com/resources/case-study/enhanced-learning-management-system-with-ext-js *for a case study on how Harbinger Systems helped a company with the choice of a JavaScript framework, and developed a Web 2.0 interface over their existing application.*

Learn More: *See how you can use Web 2.0 concepts for the makeover of enterprise software, in the Harbinger Systems white paper, HSTW-101 – The Enterprise Software Makeover Guide. Visit* https://www.harbinger-systems.com/resources/whitepaper/the-enterprise-software-makeover-guide *to request the paper.*

Sources

[1] http://www.extjs.com/

[2] http://www.techpresentations.org/JavaScript_Library_Overview

[3] http://designingwebinterfaces.com/

[4] http://www.ja-sig.org/wiki/display/UP3/Javascript+Toolkit+Comparison

[5] http://www.kauriproject.org/wiki/g1/g4/g2/69-kauri.html?layoutType=plain

[6] http://extjs.com/blog/2007/07/10/css-selectors-speed-myths/

[7] http://www.windley.com/archives/2008/10/prototype_framework_is_a_javascript_extension_not_just_a_library.shtml

[8] http://alex.dojotoolkit.org/2005/12/file-uploading-with-dojo/

[9] http://open.silverstripe.org/wiki/development/javascriptplatform/Research