

# Client-Side Programming



By Russ McGuire - [russ.mcguire@gmail.com](mailto:russ.mcguire@gmail.com)

Up until now I've focused on server-side technologies for our [online social network](#). [PHP and MySQL](#) (especially leveraging many of the [other platforms](#) and [web services](#) we've looked at) can deliver tremendous capabilities. But today's modern websites rely heavily on client-side programming capabilities to deliver a pleasing user experience.

---

## What is Client-Side Programming?

People accessing your website generally do so using a web browser – this is the client – and your website sits on a server in the network. PHP is a powerful server-side programming language that can make your website dynamic – showing the most up to date information. However, for your users to see any new information; their client (web browser) has to make a new request to the server and the entire page gets downloaded again and redrawn on their screen.

Client-side programming allows a program to run inside the web browser so that new information is presented in a portion of the browser window without the entire web page having to be reloaded, and often without the user having to take an explicit action.

Let me give a very simple example. It is very easy for me to write a PHP program on the server which will embed the current time into a web page. I could even get fancy and represent it as a clock face. However, the clock won't update at all until the web browser requests an update and the server sends the whole web page again with an updated clock. Using client-side programming, I can update the clock in the window every second if I like, without impacting anything else on the web page and without sending any additional traffic over the network.

## What are Client-Side Programming Languages?

The vast majority of client-side programming today is either done in JavaScript or in Flash. Microsoft's Silverlight is an emerging alternative to Flash, and HTML5 will introduce client-side programming capabilities as it becomes standardized. But for now, if you want to program in the client, you'll probably be using either Flash or JavaScript.

Flash is a very powerful tool for creating impressive web interactivity. However, there are issues with Flash that have caused me to steer clear of it. First, Flash is a proprietary technology of Adobe, so to take full advantage of Flash, I need to buy development tools from Adobe, and my users will generally need to have the Flash plugin installed in their browser, which leads to other challenges. Flash plugins haven't generally been available for mobile devices, and even if they are, Flash is a bit of a resource hog, so performance on lower end systems (and mobile devices) can be painful. Some security experts have even [recommended](#) that users disable Flash because "bad guys" could use Flash's power to do "bad things".

JavaScript doesn't perfectly overcome all of the challenges of Flash. There are [security concerns](#) with JavaScript (although fewer than Flash), JavaScript does

require more processing power than just plain HTML (but generally less than Flash), and not all browsers support JavaScript, especially on non-smartphone mobile devices. But, JavaScript doesn't require a browser plugin, and there are many good JavaScript development tools available for free.

### JavaScript's Strengths are Its Weaknesses

Because JavaScript (unlike Flash) runs natively in the browser, it is susceptible to variances amongst browsers in how the script is interpreted. I'm sure this is nothing new to you since web developers already have to deal with differences in how Internet Explorer, Firefox, Safari, Chrome, etc, interpret CSS or even some basic HTML tags. But it's a pain, none the less.

Thankfully, open source developers have created JavaScript frameworks that not only handle the cross-browser idiosyncrasies of JavaScript, but also make it much easier to tap into the full power of the language. This is where JavaScript's second strength becomes a challenge – unlike Flash (where you basically have to go to Adobe for good tools), there are so many JavaScript frameworks to choose between – how do you choose?

Here are a few of the most popular JavaScript frameworks:

- [jQuery](#)
- [Prototype](#)
- [YUI](#)
- [Google Web Toolkit](#)
- [Dojo](#)

### How Do Frameworks Help?

I've found that there are three basic concepts that demonstrate the value of JavaScript frameworks.

The first concept is selectors. Everything on a web page can be considered an object that can be used by JavaScript. That means every header, every paragraph, every list, every list element, every image, every link, every form element, and every button – basically anything that has an HTML tag – can be selected by JavaScript for action. The technical name for this is the Document Object Model (DOM). JavaScript can respond to the user's interaction with those elements – such as mousing over a header, clicking in a form field, or checking a checkbox. However, from a programmer's perspective, being able to identify that element and set up the appropriate

**TicketU™**

# Don't let December sneak up and bite ya'!

**Avoid the last-minute rush. Sign up now to sell your Christmas performance tickets online – and **SAVE!****

**TicketU** can dramatically boost ticket sales to your holiday events.

Sign up to use TicketU for online ticket sales by 8/31/10 and receive your first **300 tickets FREE** (that's a \$300 savings!)  
Initial setup fee required.  
Offer available to new TicketU clients only.

Sign up to sell tickets online and save!  
[www.ServiceU.com/christmas](http://www.ServiceU.com/christmas) 866.709.7120

SERVICEU™

actions for it, can be a real pain – especially since it works slightly differently in different browser versions. JavaScript frameworks generally make it simple to pull out a single element. Frameworks especially make a big difference in the area of DOM traversal –for example, being able to easily identify the 3<sup>rd</sup> item in the second ordered list on the page.

The second concept is DOM manipulation. Each element on the page has attributes that can be modified by JavaScript. That includes visibility, color, background color, border, location, text, link, etc. JavaScript frameworks make it easy to modify these attributes in very sophisticated ways. For example, an element can gradually change from one color to another when the mouse is placed over it. JavaScript frameworks even provide a way to create new elements on the page. For example, when you're filling out a form, each time you complete the last text box in a list, another text box can appear in case you have another item to add to the list.

The third concept is user interface elements, or widgets. JavaScript frameworks, either natively, or with plugins, make it really easy to add sophisticated user interface elements to your page. For example, when the user clicks on a date field, a small calendar widget may appear making it easy to select a date. This widget is easily implemented using a JavaScript framework.

### Choosing a Framework

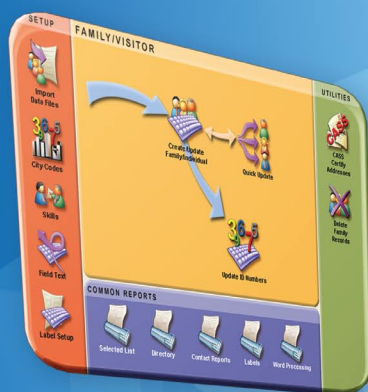
We've only tried two JavaScript frameworks. We first used YUI and were hooked on the concept, but then struggled with a particular implementation and searched for an alternative. We came across jQuery and have been using it ever since. One difference between the two is that YUI is focused on making Yahoo's user interface elements available for other developers, while jQuery is a more typical open source project focused on harnessing the innovation of a large

community of developers. To further explore the differences between the framework options, I would recommend [Wikipedia's comparison table](#) and IBM's article titled "[Compare JavaScript frameworks](#)."

As you go through your decision process, you very well may reach a different conclusion, but I hope hearing our approach has been helpful to you.

*Russ McGuire is an executive for a Fortune 100 company and the founder/co-founder of three technology start-ups. His latest entrepreneurial venture is Hschooler.net (<http://hschooler.net>), a social network for Christian families (especially homeschooled students) which is being built and run by three homeschooled students under Russ' direction.*

## EXPAND YOUR BOUNDARIES WITH PREMIUM MINISTRY TOOLS



**Church Management**  
**Fund Accounting**  
**Event/Facility Scheduler**  
**School Management**  
**Donor Management**  
**Ministry Scheduler**  
**Group Ministry Management**  
**Online Manager - Giving**  
**Updates**  
**Registration**  
**Direct Deposit**



For a Free Intro Tour CD  
 Call  
 1-800-266-3311  
 Or Visit  
[www.logoscms.com/intros](http://www.logoscms.com/intros)



**LOGOS**  
 Management Software INC.