Over the past 15 years, JavaScript has evolved from a "toy" language to a mainstream programming one that powers Web browsers, games, and server applications. Perhaps the most significant development that led to JavaScript's ubiquitous use was the emergence of the jQuery library. In many development shops, jQuery is the core library upon which all other application code is developed.

I spoke with jQuery creator John Resig, currently of Khan Academy, which offers instructional videos and learning tools for math, science, computer programming, history, and more. You can see the entire interview with John at www.computer.org/computingconversations.

John Resig discusses the birth of the JavaScript library jQuery and how it came to play a key role in the browser software ecosystem.

BUILDING AN ELEGANT LIBRARY

John started developing what would become jQuery in 2005 as an undergraduate student in computer science at Rochester Institute of Technology. Like many undergraduate students, John was involved in several Web development projects that were exploring emerging browser-based technologies. One of the early libraries he encountered was Prototype, which was bundled in Ruby on Rails:

The most popular JavaScript library back then was Prototype. It was the first time I had seen a JavaScript library written in a nice, clean, object-oriented manner. Up to that point, I didn't realize that JavaScript code could be beautiful and elegant. Looking at Prototype inspired me to want to build something even better and add capabilities like manipulating HTML in the Document Object Model [DOM].

Over time, as John looked at emerging JavaScript libraries and started building his own, he felt it was important to simplify development in the browser:

See www.computer.org/computer-multimedia for multimedia content related to this article.
COMPUTING CONVERSATIONS

I saw a usability chasm, so I started to build different tools and libraries, and eventually these all glommed together into a single library, which I ended up calling jQuery. I was originally going to call it jSelect, but that domain was taken. I released it January 2006. At the time, I was in college working on all these different side projects. jQuery just spun out of these side projects—none of which exist anymore, except for jQuery.

Initially, John designed jQuery to achieve two goals. The first was to develop an elegant and powerful selector syntax that would allow programmers to access and modify the DOM:

I found the selector pattern in a library called css_selector written by Simon Willison. He created a method that allowed you to write a simple CSS [Cascading style Sheets] selector and find some elements. It was very primitive: you could only do the most basic of queries, and it didn’t have support for CSS2 and CSS3.

The second goal John wanted to achieve was for JavaScript developers to be able to add events to various places in the DOM:

When you’re trying to manage an open source project, the code is only a small portion of the total effort. You have to make sure you have a clear website and a very nice “Getting Started” guide. The very first person I added to the jQuery team was someone to help manage the community. I wanted to make sure that if anyone was having a problem with jQuery, their needs were going to be taken into account, and we could fix whatever issue they encountered.

In summer 2006, John and several of his friends were part of the seed accelerator Y Combinator and moved to Boston to turn their idea into a start-up. His work on jQuery was still just a side project. The start-up didn’t thrive, but John decided to work at the Mozilla Foundation as a JavaScript evangelist:

My job at the Mozilla Foundation was to promote JavaScript specification activities and improvements to JavaScript. I still wasn’t actually working on jQuery as part of my job. In my spare time, I would work on mailing lists and bug fixes in jQuery. It wasn’t until my last year at Mozilla [2010–2011] that I was allowed to work on jQuery full time.

John spent his last year at Mozilla strengthening the jQuery community:

During that time, I put a lot of effort into making sure there was infrastructure in place so that if I wasn’t there on a day-to-day basis, it could continue to exist. Part of that was setting up a nonprofit and making sure there were enough people working on every aspect of it. When I joined Khan Academy, I stepped down from technical involvement in the jQuery project, and everything has been running very well since then. I can just use jQuery now as a happy consumer.

Over the past decade, thanks to the work of John Resig and many others in building and evolving high-quality JavaScript libraries, moving application functionality into browsers is increasingly becoming a reality. The jQuery library provided a base level of functionality, almost like an operating system for browser applications that provided a solid foundation to move the entire industry forward. It will be interesting to see how the following generations of JavaScript libraries will build on this innovative work.