

KAAZING: The Heart of the Living Web

COMPANY OVERVIEW WHITEPAPER

Introduction

The Web we take for granted today--real-time, immediate, collaborative, and mobile--is something the inventors of the Web never expected. And how could they?

Who could have imagined a Web where tweets fly back and forth among hundreds of millions of people, giving instant updates on every part of our lives? Or thought that by flicking on a map on your phone, you could instantly pinpoint the whereabouts of friends and colleagues? Or anticipated that one day, most of our time spent online would revolve around social networks powered by live status updates?

Today's Web is a *living Web*, an organism whose evolution and growth is fueled by the deluge of immediate, ever-changing, ever-expanding data we're constantly feeding into it. It couldn't be more different from the original Web, which after all was designed to transfer static documents.

Ironically, it's this transformation that poses one of the biggest challenges to the Web's progress. The underlying architecture simply can't keep up with a living Web that's becoming faster, more instantaneous, more collaborative, and more mobile.

The problem is the Web's original design. Created to exchange documents, the Web was architected to transfer documents using a patient request/respond method. But speed, constant connection, and collaboration define communication today. Smart phones, iPads, and iTVs are just a few of the devices turbocharging this change.

Because the original Web standards had not evolved much since 1995, the industry responded by cobbling together back-end fixes that, on the surface, allowed the Web to adapt to these new demands. But these kludges come with significant costs. They use unnecessary amounts of bandwidth, drive up development costs, gobble up energy, and keep the Web from delivering its full promise.

The Kaazing Platform, the only enterprise-class high-performance Web communication platform and a new approach to online connectivity, was created to remove these constraints. WebSocket is an industry standard that the founding team behind Kaazing helped develop to make the existing Web infrastructure cheaper, faster, and easier to manage. The WebSocket protocol creates a persistent network connection—a socket—that transforms today's HTTP request/response communication into a simultaneous two-way connection. It's like replacing the telegraph with the telephone.

The Kaazing Platform builds upon the WebSocket protocol, making it richer and more accessible, creating the basis for an ecosystem that meets the demands of enterprises looking for flexible, reliable solutions and programmers eager to build on top of the technology. Kaazing is at the heart of the Living Web, providing the foundation for cheaper, faster, truly real-time and always-on connectivity.

The Living Web

It's hard to imagine, looking at the Web today, that the word "static" could ever be applied to it. Yet, the Web was initially designed to transfer static documents, static text.

It was a modest beginning for something that's become so dynamic. Today's Web is an organism that is constantly growing and evolving based on the data fed into it and the applications built upon it. We're staying connected in real time. Powered by tweets, status updates, instantly uploaded video clips, location alerts from mobile phones and other new streams of rich data, the Web reflects every aspect of people's lives.

We're creating a Web of things--smart, analytical networks of sensors that will eventually number in the trillions. From railway systems to water authorities, organizations are using sensors to track activities and predict problems before they happen. This data is ushering in new ways of thinking about the world.

We're collaborating in new ways. Novel applications powered by the always-on Web lets anyone, anywhere, anytime work or play with others simultaneously. Tapping into this collective expertise is unleashing new approaches to innovation and solving problems.

But as much as new Web technologies enable faster transmission of documents, the Web isn't meeting its real promise. Because of the way the Web is designed, each time we make a leap forward, we introduce new costs and trade-offs.

Today's Web

The problem is fundamental. We think that what's happening online is the height of instantaneous, fast, cheap access to data. But it's not.

What we have right now—the social interaction, the business opportunities, the analytics—is a pale reflection of what would be possible were the Web truly live, always on, and capable of handling fast, simultaneous connections easily and cost effectively. The Web was designed to deliver static pages one request at a time. It performs well in the document and search oriented Internet of the past. But it doesn't meet the demands of the Living Web for a few key reasons:

1. High Latency

Latency is caused by the HTTP's request-response design.

With HTTP, browsers request bits of data, such as the page of an article or a photo from a Web site's server. But each time the browser requests information, it has to wait until the server responds before it can ask for something else.

Communicating on the Web is more like sending telegrams than talking with someone on the phone. Rather than a conversation where people speak at the same time, you send a message and then wait for them to read it and respond before you can continue your discussion.

2. Heavy Bandwidth Consumption

Every time an HTTP message is sent, it is accompanied by a set of header data that contains information for transmitting the data. These headers are large -- often larger than the payload or data being delivered -- and can end up consuming most of the network bandwidth used during the communication.

3. Complicated Translation

Browsers only speak HTTP; back-end business systems don't. Because there are many different enterprise protocols, it takes a lot of translation for data to go to and from the server and browser.

4. High Scale

As the legacy Web evolved into the Living Web, we came up with fixes to simulate a real live data experience. But they create their own headaches.

To speed up the Web experience and make it seem more interactive, workaround techniques including Ajax and Comet were developed that simulate real-time connectivity by pushing data from servers to browsers. These techniques work by constantly polling a Web server to check for updates—often needlessly, since most polls go unanswered. Yet the header data that accompanies each of these useless request/response cycles wastes a huge amount of bandwidth and CPU power. The explosion of mobile devices exacerbates the problem. People expect the same real-time, interactive experience they get with laptops, increasing the strain on networks and energy resources.

Companies are forced to pour huge amounts of money into back-end and application architectures to work around the inefficiencies of HTTP. They also have to invest in building and maintaining different architectures for many platforms and devices.

Despite all this work and money, the Web still isn't truly real time or interactive. That puts the brakes on industries that need live connectivity to thrive, including financial services, retail, online gaming, and logistics. For instance, for financial traders, a delay of a few milliseconds can add up to tens of millions of dollars in lost profits annually. Or consider retailers: as much as they want to provide real-time suggestions and customer service to individuals shopping online, all they can do is to track users' browsing behavior and open a pop-up box inviting them to chat.

WebSocket: Evolution in Web Communication

The WebSocket protocol was created to tackle the shortcomings of the legacy Web and let the Living Web reach its potential.

The costs of patching existing systems keep piling up. The more that applications, devices, and Web sites try to respond to customer demands for instant communication, the greater the demands on datacenter and network resources become.

WebSocket solves these problems. Designed for efficient, simultaneous two-way communication, WebSocket doesn't replace the existing infrastructure, it works with it. WebSocket's big leap forward is that, while it's compatible with HTTP, unlike HTTP, it creates a full-duplex data channel. It transforms today's request/response communication into a constant two-way connection in which both sides can send information to each other, even at the same time.

WebSocket initiates a handshake over HTTP, and then upgrades the connection to an always-on socket. A client and server (or server and server) can send information through the connection without the recipient having to wait for a request.

The result is true, interactive connections that replace today's illusion of always-on communication. This innovation does away with the need for fat pipes, replacing dumb repetitive connections with smart thin ones. It removes the need for much of the middleware and work-around technologies that drive up the cost of delivering fast, interactive connectivity.

WebSocket on its own, though, is simply a socket; an empty pipe. To transform it into a solution that's robust, reliable, and flexible enough for enterprises to depend on, Kaazing developed the Kaazing Platform, the only enterprise-class high-performance Web communication platform on which this Living Web ecosystem can flourish.

The Kaazing Platform makes the formerly impossible possible. For instance, Kaazing finally lets retailers cater to customers live and in real time. Capturing the click-stream from consumers in real time, a merchant can offer instant deals to motivate them to buy or to buy more. Meantime, worries over losing customers because of slow-loading pages disappear at services such as concert ticket or airline sites.

Because Kaazing Platform is based on the HTML5 family of Web standards, developers can build applications that do not require client downloads or plug-ins. That can put an end to fiddling with application downloads before starting an online Web conference or stock trading session.

TV could finally be truly interactive, dishing up players' real-time stats on a tablet or smartphone during a baseball game, serving real-time merchandising deals, and letting viewers interact with other fans watching the same game. In countries where gambling

is allowed, intra-game gambling would also be supported; for instance, after a player is awarded a penalty kick in a soccer game, viewers could bet on the outcome.

Kaazing: The Heart of the Living Web

Kaazing was founded to end the Web's connectivity gridlock. The Web unleashed a communication revolution, but this next phase of its development—live, interactive, collaborative—can't live up to its true potential until the inefficiencies and costs associated with adapting to the legacy Web are erased.

Kaazing's founders know what it takes to attack these constraints. As HTML5 was being developed, they saw the chance to do something bolder and developed the HTML5 WebSocket specification to turn the Web's request/response mode of communication into simultaneous, two-way connections; a telephone rather than a telegraph.

Kaazing's founders contributed their work to open standards committees so the WebSocket protocol could become pervasive. And indeed, the major browsers have adopted or are in the process of adding the protocol.

But WebSocket alone isn't sufficient to end the gridlock. Kaazing's founders also recognized the need for a whole new universe of development tools, flexible services, easy-to-use solutions, and heavy-duty enterprise offerings that respond to the broad demands of the market, whether from established companies or startups, professional developers or tinkerers.

That's where the Kaazing Platform comes in. The first product in the platform is the Kaazing WebSocket Gateway, the world's only enterprise solution for full-duplex, high-performance communication over the Web using the HTML5 WebSocket standard.

By making connectivity faster and more efficient, by eliminating the need to build and maintain different back-end architectures for different platform and devices, the Kaazing Platform's approach is already slashing middleware and bandwidth costs and re-invigorating the evolution of the Living Web.

Summary

As the Living Web continues its dramatic evolution, it is fast resetting our expectations of what it means to be connected. Given what we achieved in spite of the constraints and limitations of the legacy Web, it is incredible to imagine what an unfettered Living Web could bring.

By slashing overhead, eliminating artificial constraints, and creating an infrastructure that's leaner and more efficient, Kaazing is making it possible to unleash a truly interactive, real-time, dynamic Web – a *Living Web*.