Kynetx Report

The Advent of Next Generation Browsing

Introducing the Structured Web

Phillip J. Windley, Ph.D.
Chief Technology Officer
Kynetx
January, 2009

kynetx
where opportunity meets desire

Introduction

We are mired in a tangle of architectural legacies that make today's Web browsing experience uncomfortable, confusing, and tiresome for many users. In particular, the lack of Web site independent identity has hampered the ability of the browser to effectively intermediate the Web on the user's behalf. But change is coming and we are about to witness a significant improvement in the nature of Web browsing—indeed, the nature of the browsing experience is about to change forever.

The current disaggregated browsing experience is untenable and must change. Significant trends are combining to make that possible.

In the early days of the Internet, companies sought to give users the benefit of a consistent experience by building portals that integrated multiple activities. Portals are now mostly a thing of the past; a few large examples such as Yahoo! and MSN still exist, but by and large they have fallen victim to what must be the most important law of the Web: a different site is only a click away. As users sought out the best sites for any given purpose, the browsing experience fractured and became ad hoc. As a result, disaggregation of Web sites and services is now the norm.

An example from the world of ecommerce illustrates this. Shoppers use search engines like Google or Yahoo! to find a product and choose an online retailer from the search results. Before buying they might research products at independent review sites like Epinions.com and Viewpoint.com. They might discuss those same products on myriad blogs, Twitter, and social networking sites like Facebook.

As we'll see, disaggregation causes users to manage too much of the Web experience themselves. This situation is untenable and must change. Fortunately, three major technology trends are creating the needed opening for improving the browsing experience:

1. The browser becoming a viable platform
2. Cloud computing
3. The advent of Internet identity

But, as significant as these three trends are, they do not in themselves solve the problem. We will review the history that has led us to the current browsing paradigm, explore these current significant technology trends, and examine how both old and new players are leveraging this historical moment in time. Then, we will introduce the notion of a new browsing experience that will change how you do things on the Internet.

Kynetx is the first vendor to take advantage of this opportunity to invent what we call "structured browsing." Kynetx generates the infrastructure for structured browsing.

A Brief History of the Web

The Web is built upon three technologies:

- **HTML**, the markup language that specifies page format and, most importantly, allows links to be specified,
- **Uniform Resource Locators (URLs)**, the addresses that specify to which resource (usually a page) a link on the Web points, and
- **HTTP**, the transport protocol that retrieves, creates, and updates resources.

These, in turn, were built on protocols and concepts that came before, such as the Internet Protocol (IP), IP addresses, and the Domain Name Service (DNS). In this world, the machine was king and early Web sites and usage concentrated on this machine centric view.

As more and more people began to use the Web for things people are interested in, like shopping, it became clear that additional changes were needed. Specifically two changes happened in relatively short order:

- A mechanism was introduced using cookies for creating sessions where transactional context could be stored.
- A mechanism for protecting the data transmitted over the wire was created using Transport Layer Security, or TLS (also called SSL).

Because of privacy concerns, cookies were designed so that one Web site could not read the cookies from another site. This created a situation where the user's activities were secure and relatively private at the expense of allowing sites to cooperate easily to create a good customer experience.

Beyond cookies, Web identity has been largely an afterthought. The only digital identity mechanism built into HTTP—HTTP User Authentication—is quite primitive in its UI elements and so most sites have avoided using it in favor of ad hoc mechanisms for identifying users. The protocol layers below HTTP have no way of handling identity information.

The technical underpinnings of the current Web are insufficient to create a context-rich, structured browsing experience.

As a result, the only identity information a Web site usually has is one that identifies the machine or the browsing session. Further, any additional identity information that the Web site develops can be used on that Web site alone. Thus, today's browsing experience is ad hoc. The Web site has no way of playing intelligently inside a broader, cross-site context for the browsing session.

Current trends in computing and browser technology are providing a scenario where this ad hoc browsing experience can be shifted to a more advanced session that is based on much richer context information under the control of the user. This new experience is known as structured browsing and represents a significant shift in the way things work on the Web.

A change of this magnitude could not occur, of course, if it in any way disrupted the existing browsing metaphor or technology base. The only way to move to a next generation browsing experience is to allow customers to move at their own pace, and without requiring significant changes to the browser, Web server, or baseline methodology.

This is a tall order and explains why it has taken 14 years to ensue.

To further understand this, let's look at the current technology trends and their significance.

Recent Trends on the Web

The Browser as a Platform

The Web browser has gone through a significant growth and stabilization process since its inception. As a result many computer users find that they are spending more and more of their time in the browser.

Over time, browser vendors have learned that creating a development ecosystem around their browser products is critical. Each vendor has developed an ecosystem that supports and encourages third parties to develop applications that are specific to the browser. These applications span the gamut from changing the layout and color of the browser to FTP clients and security components.

The browser has emerged as a viable platform for developers seeking to create new applications. Users love customizing their browsers to meet their individual needs.

The arrival of the browser as a platform is creating an opportunity for many software developers to add significant value to the browser and the browsing experience.

To give you some idea just how healthy this ecosystem is, consider a few statistics provided at a recent browser conference hosted by Microsoft and Mozilla:

- According to Justin Scott of Mozilla[1], over 62,000 add-ons are downloaded every hour for Firefox.
- Since 2005, over one billion add-ons have been downloaded.

At the same conference, Microsoft indicated that it too has a significant, dynamic ecosystem for browser extensions.

This is significant support of the browser as a platform. For the foreseeable future, this trend is likely to continue. Indeed, as we shall see, several companies have collaborated in developing significant software infrastructure that runs on the browser platform that is essential in bringing about the advent of structured browsing.

The browser platform architecture allows for infrastructure to be added in an organic way that gives the customer a choice about when to add the functionality and when to use it. At the same time, this approach lets the existing ad hoc browsing model to continue operating just as it always has.

Online Identity is Going Internet Scale

At a talk at the University of Utah in 2005, Vint Cerf[2], one of the inventors of the Internet and the developer of the TCP/IP protocol was asked what he would have changed about the design of this key protocol if he could do it over. He responded that he wished that identity had been built into the protocol from the start. As a consequence of that decision, identity on the

Information cards provide—for the first time—a way for people to safely, securely, and conveniently manage their online identities independent of any particular Web site.

Internet has always been an afterthought and difficult to do. Pseudonymous activity is the norm because of the architecture of the Web.

Web sites have always struggled with how and when to identify their customers. Registration systems allow repeat visitors to use Web sites in a more convenient manner, but initial registration requirements introduce significant friction into the process, resulting in lost transactions.

Recently, new developments in Internet scale identity systems like Information Cards[3] are changing the landscape for Web sites, credit card issuers, and most importantly Web users.

Information Cards are the product of an effort by Microsoft to solve the Internet identity problem. Based on open standards and available from several vendors, Information Card technology provides a convenient, safe, secure way for people to selectively reveal their identifying information, preferences, traits, and relationships to a Web site. The architecture of the Information Card system protects people's privacy by putting them in the driver's seat.

Information Cards provide the user with a familiar metaphor in the form of a wallet-like application called a "selector." Figure 1 shows Microsoft's Information Card selector. When Web sites ask for identity information, users reach into their selector and present the Web site with an appropriate card—just like they do offline.

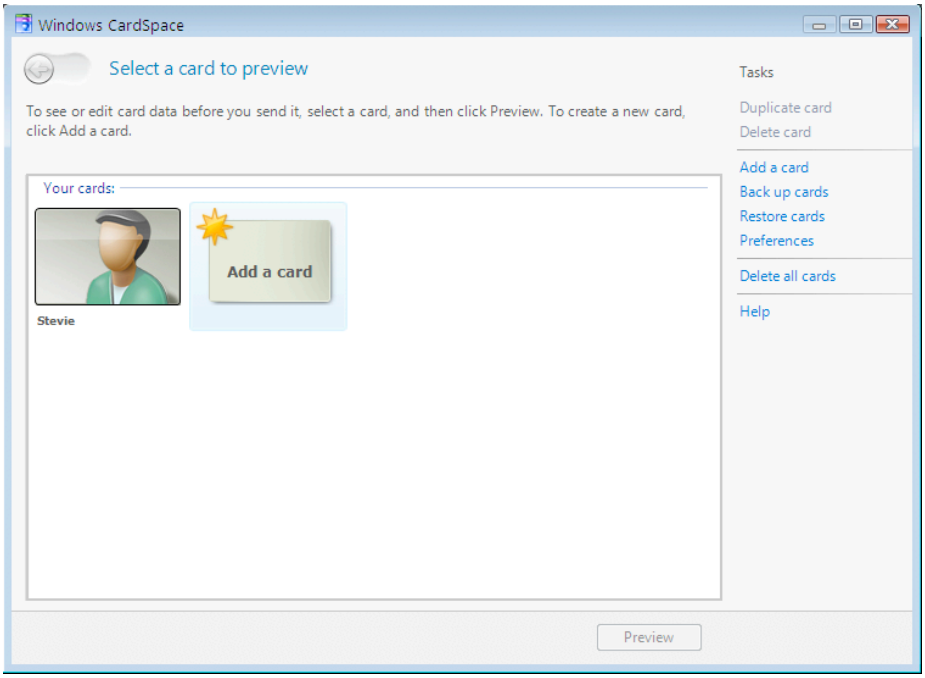

Figure 1. Microsoft's Information Card selector

A number of large companies, including Microsoft, Google, and PayPal are members of the Information Card Foundation, the group standardizing Information Card technology. The New York Times said[4] "The presence of PayPal, which is owned by eBay, in the group is the most significant: PayPal, with its direct access to our checking accounts, will naturally be inclined to

be conservative. If it becomes convinced that these cards are more secure than passwords, we should listen."

Information Card compatible systems are available from several vendors including Microsoft, Novell, IBM, CA, and Parity. There are also open source initiatives for Information Card infrastructure including the Bandit Project, the Higgins Project, and the Pamela Project.

Cloud Computing

The cloud has long served as a metaphor for the Internet. In recent years, more and more companies have begun delivering their software services over the Internet rather than by delivering software that is run on their customer's computer equipment.

Cloud computing offers a model where services can be delivered immediately with no upfront capital costs.

The advantage of cloud computing is that users get a diverse, scalable, elastic, and reliable service without incurring the upfront, capital intensive costs associated with buying hardware, housing it in a data center, and licensing software to run on it. Instead, users pay only for what they use, when they use it.

A number of companies, including Amazon, Microsoft, and Google offer access to their products "in the cloud." In addition to these major players, many companies make "as a service" delivery the core of their business model. Perhaps no company is better known for that than Salesforce.com.

As we'll see below, cloud based services are particularly powerful when combined with browser add-ons since they make the add-on simpler to build; new functionality can be added without the user having to upgrade the add-on itself. Cloud based services combined with add-ons represent a disruptive force because new customized experiences will be easily delivered to Web users on demand and in context.

Introducing Structured Browsing

Most browsing episodes are ad hoc. Web page content is delivered however the provider decides and in whatever order they want. The order, placement, and features of many Web sites are arranged without regard to convention. Moreover, the functionality of most Web sites is available for machines to manipulate, but difficult for people to interact with except through the mechanisms that the site designer has created in the user interface. This is changing with the advent of structured browsing.

Episode Context

One way to keep children entertained during a long drive is connect the dot pictures like this one:

The chore of managing the context of a browsing episode usually falls to users, not their computers.

We can imagine visits to individual Web sites in a browsing episode being like the dots in this picture. Only by connecting the dots does the image appear[5].

When people go from site to site in a browsing episode with the goal of accomplishing some task, they must connect the dots in their heads. We call the lines that connect the dots of individual Web site visits "context."

Context is all of the information relevant to a given browsing episode and can include the traits, preferences, and attributes of the user. The primary challenge of an ad hoc browsing experience is that the user must manage much of the relevant context rather than being able to turn this chore over to the machine. Browsers provide very little help with this task.

In many cases, browser add-ons are performing some part of the context management task. For example, consider the case of a simple add-on that overlays search results with information on whether or not the sites displayed are known phishing risks. In this example, the add-on is managing part of the security context that users would otherwise have to manage themselves.

Browsing in Context

As we have seen, architectural decisions made long ago about Internet and Web architecture created an environment where machines, IP addresses, and resources (as identified by URLs) were primary and people were secondary. These decisions, which form the technical underpinnings of the current browsing experience, leave a Web that provides only minor support for browsing in context.

Contrasted with this ad hoc experience is one where rich context, built on a foundation of Web site independent identity, can be used to give people enhanced browsing experiences and relationship-based interactions while at the same time protecting their identity, privacy, and security.

People don't usually install add-ons merely to customize their browser. They are customizing their browser for a purpose: to impose order, form, control, and classification to the otherwise ad hoc flow of information that a stock browser provides.

The browsing experience is greatly enhanced when the browser can take advantage of rich context, built on a foundation of Web site independent identity.

In contrast to ad hoc, structured browsing experiences are:

- Browser-aided – The browser, with the added functionality provided by structuring devices like add-ons, serves as a tool for helping the user manage context and take action based on that context.
- Individual – Context is dependent on what is relevant to a person. Some people, for example, may care about security and others about saving money. In the case of the security-minded person, the safety of sites is relevant context. In the case of the frugal shopper, the availability of discounts is relevant context. Any system for managing context has to allow for customization at an individual level without introducing layers of burden for the provider.
- Based on identity – Individual customization is not enough. With an underlying foundation of an identity system that is designed to protect personally identifying information and secure data, structured experiences bolster the privacy and security of users.
- Relationship supportive – By creating an individualized, private and secure experience, structured browsing supports an atmosphere of trust wherein relationships between service providers and their customers can flourish. Relationships based on trust create loyal, satisfied visitors.
- Verifiable – When the status and reputation of both service providers and their customers are verifiable, transactions can happen in an environment of trust. This can only happen when built upon the foundation of secure and private identity.

Kynetx Network Services—Structured Browsing Infrastructure

Kynetx Network Services (KNS) is designed to make it easy to create context-aware, structured browsing experiences.

KNS provides a way for Web sites to respond to visitor context and deliver better, more customized experiences to users. KNS provides an abstraction layer on the Internet for creating cross-site transactional context in support of structured browsing experiences.

KNS has several important features that make structured browsing easy and effective:

- KNS is Web site independent. KNS is designed from the ground up to work with almost any Web site.
- KNS works across Web sites. KNS responds to user context, even when that context includes information the user collected at another site.
- KNS is browser independent. The KNS system works with all modern browsers.
- KNS is context-aware. KNS makes use of permissioned, personal information that the user supplies as well as ambient data about current and past browsing episodes.
- KNS is real time. KNS is constantly working in the background to provide context-aware interactions that are customized to user, page, and moment.
- KNS is secure. Because KNS is based on Information Cards, KNS provides a secure, private environment where context is shared under user control.

Using KNS, Web sites can easily respond to visitor context and provide a customized, structured experience.

KNS provides better, personalized experiences to users and at the same time allows Web sites to be more responsive to user needs. This is the basis for a relationship between visitor and Web site that engenders trust and loyalty. KNS does not mediate that relationship, but enables it by providing a richer foundation upon which mutual reputation, trust, and loyalty can be built.

How Kynetx Works

KNS depends on three inter-related technologies to accomplish its work.

- **Kynetx Information Cards** (KIX) provide Web site independent identity. Specific cards, through the claims they contain, activate functionality inside the browser.
- **Kynetx Rule Language** (KRL) is a domain specific language that provides an abstract, linguistic means of specifying structured experiences through customization of browser content.
- **Kynetx Rule Engine** (KRE) evaluates KRL rules in response to requests from the browser and responds with custom JavaScript code to be executed in the browser.

The interaction of these three technologies provides a means for companies, developers, and even individuals to create structured browsing experiences on the Web.

As shown in Figure 2, before KNS, users only interacted with the Web site (1). Kynetx adds two critical components to generate structured browsing: the KIX in the information card selector (2) and the cloud service, KNS, that customizes the page based on the user context (3).

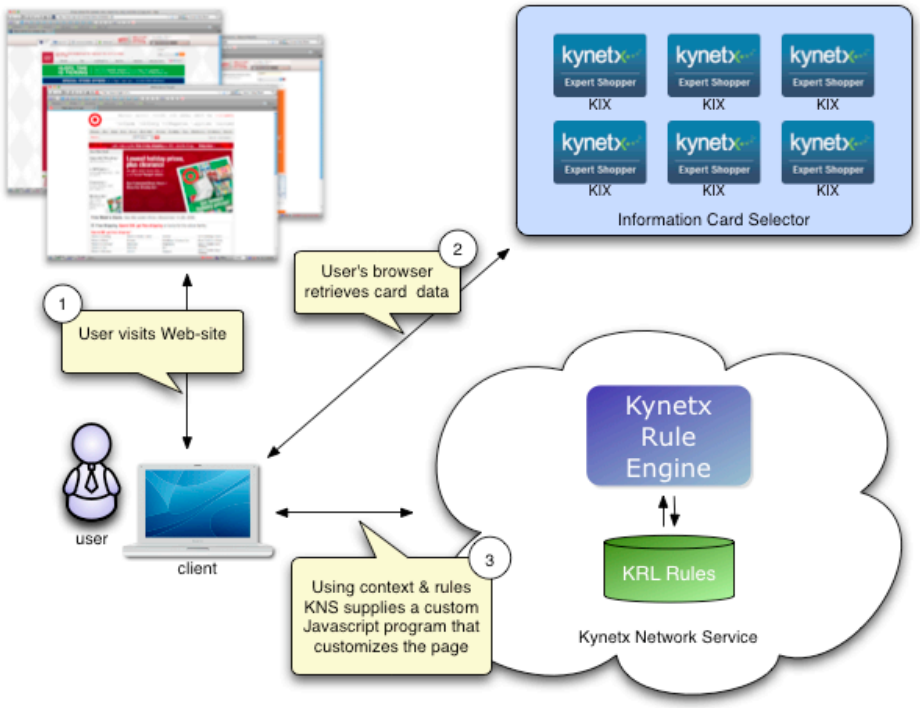

Figure 2. KNS customizes the page the user is viewing

Kynetx Information Cards

KIX are an abstract representation of desired functionality. When a person chooses a KIX, the functionality associated with that card is enabled in their Web browser.

For example, a user concerned about protection against phishing scams might choose a KIX that provides a ratings overlay of search results and message boxes alerting them to dangerous sites if they happen to end up on one.

Another user interested in maximizing the discounts they are entitled to might choose a KIX for a reminder service that overlays search results with icons showing relationships they have that entitle them to discounts at certain merchants (e.g. AAA, AARP, UPromise, etc.) as shown in Figure 3.

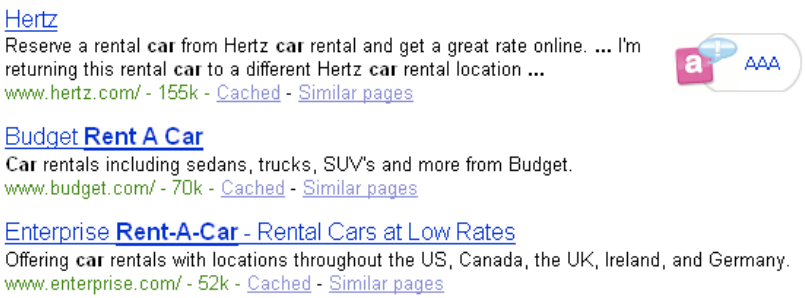

Figure 3. The Azigo RemindMe service highlights the user's relationship with AAA in Google using KNS.

KIX represent particular functionality and allow users to easily and safely manage their personal information along with their Web experience.

Before a card can affect the browser, the user will download and install a card selector (if one isn't already included with their OS) and an add-on that extends the browser so that it is responsive to the card selector and the Kynetx Network Service. That one installation of a card selector provides the user with what amounts to a universal add-on. After that single install, users gain additional functionality by putting small (< 3k) cards into the selector—a simple, secure, and friction-free experience.

The card selector allows the user to easily manage their KIX, including sorting, searching, arranging, enabling, disabling, and deleting cards. Disabling or deleting a card is easy and instantly removes the functionality associated with it from future browsing experiences.

Kynetx provides a directory of available cards that have been certified by Kynetx called the KIX Directory. The certification process ensures that the functionality associated with a KIX is safe, private, and secure. Users can browse or search for cards that provide specific functionality to structure browsing experiences. KIX might be offered for free or for sale depending on the desire of the developer

Because KIX are based on the Information Card standard, supporting infrastructure is available from a number of vendors and their safety and security is based upon 15 years of research and development.

Kynetx Rule Engine

KRE provides an API[6] that allows Kynetx rulesets to be stored, retrieved, analyzed and, most importantly, evaluated.

When a request to evaluate a ruleset is made to KRE, that request contains information such as which cards are installed, what claims they contain, the current state of the browser, and other state information. Based on this request, KRE responds with a small JavaScript program customized to the request that modifies the browsing experience. That JavaScript program, running in the browser, customizes the user's experience.

KRE includes a callback mechanism by which rules can communicate success and failure messages back to KNS. KRE provides A/B testing for rules and an analytics subsystem for monitoring rule actions and activity. A/B testing provides developers with the means for testing rules to determine their effectiveness and fitness for purpose. Analytics allows rule developers to see detailed data about how rules are being used and track which rules work and which do not.

The primary delivery mechanism for KRE is a cloud based software infrastructure service that responds to ruleset evaluation requests. Kynetx charges for ruleset evaluations on a CPM basis. Kynetx is also willing to license KRE to companies wishing to run a private or semi-private KNS system behind their firewall or for their customers.

Kynetx Rule Language

KRL is the heart of the Kynetx system and the source of its tremendous flexibility and power.

KRL, as its name implies, is a rule language that specifies what actions should be taken when a specific set of conditions is met[7]. A ruleset is a collection of rules meant to provide a particular structured experience. The KIX that are in the user's card selector, along with the Web page that the user is viewing, determine which rulesets are used. Specific rules fire based on the context contained in the request.

The Kynetx Rule Language provides a powerful abstraction for creating and delivering structured experiences on the Web.

KRL gives any developer the power to deliver structured experiences. This gives developers incredible leverage that will drastically reduce their development effort in several ways:

First, because KNS provides a kind of universal add-on that works across operating systems and browsers, developers are freed from writing complex code that represents the great bulk of the current add-on development effort. Instead, they can concentrate on what they really care about: building the functionality they want to deliver.

Second, because KRL generates JavaScript code that has been tested to work in all modern browsers, developers can largely ignore browser compatibility—one of the great obstacles to writing a widely used add-on.

Third, because of the power of its abstractions and its design for the specific task of building structured experiences on the Web, KRL gives developers a compact and easily used notation that provides real intellectual leverage. In short, one line of code can replace dozens of lines in a more general-purpose language.

Kynetx provides a simple interactive development environment (IDE) called Intersekt for writing and publishing rulesets. Kynetx anticipates that others will also build tools that support KRL.

A simple provisioning process allows developers to create a new ruleset and associated KIX. Then, using Intersekt, developers add rules to the ruleset to create the structured experience. As mentioned previously, after developers have their ruleset certified, they can place it in the KIX directory, making it available to users.

The Benefits of KNS

KNS reduces risk and increases leverage for businesses wishing to offer their customers a context-rich, structured browsing experience; users seeking better, more contextual browsing; and developers customizing the browser experience.

The following lists some specific benefits:

- KIX, KRL, and KRE provide an abstract means of adding functionality to the browser giving developers tremendous leverage.
- A Kynetx structured experience is cross-platform and multi-browser without effort by the developer.
- A Kynetx structured experience is more secure than one delivered by a typical add-on. The only software running on the user's

machine—with access to the user's private data—is the card selector and a single, simple, easily analyzed browser add-on.
- The architecture of Information Cards is designed to put users in a position where they can control the release of their private data including any claims that are associated with their KIX.
- Security and privacy are built-in to the underlying technology.
- Structured experiences provide a trusted, secure environment where high value relationships can be built and maintained at lower cost.
- Users are freed from managing episode context themselves and can turn much of that chore over to the browser.
- Users can customize and enhance their browsing experience using small, easy to manage tokens in the form of KIX rather than though large, bulky code-based add-ons.

Summary

People who experience structured browsing will wonder why it hasn't always been this way.

Reshaping the browsing experience to allow people to use the Web in a way that preserves episode context across multiple sites will transform the Web beyond what we can imagine. As we've discussed, three important trends are making this possible:

1. The browser becoming a viable platform
2. Cloud computing
3. The advent of Internet identity

By themselves, these trends will not change the Web. But when they are combined so that their individual strengths play off each other and wrapped in an abstraction like KNS, they promise to change the way people browse.

Structured browsing is a compelling shift in the world of computing. People who experience structured browsing compared to the traditional ad hoc experience will wonder why it hasn't always been this way. As an individualized, real-time service that operates across multiple Web sites, structured browsing offers people freedom from the tedious chore of connecting the "context dots" between multiple sites and, as a consequence, streamlines browsing episodes.

KNS significantly reduces the risk that businesses must undertake to provide structured experiences for their customers.

As a platform, KNS provides an abstraction layer on the Web, making it easy to create context-aware, structured browsing experiences. Our vision is one that allows existing methods of using the Web to co-exist with structured browsing, increasing opportunity and decreasing risk for all participants.

KNS allows businesses to provide structured experiences for their Web visitors by leveraging the context of individual browsing episodes. KNS gives developers a low-risk avenue for extending the browser with context-aware services that interplay with Web sites.

Kynetx offers a low-friction way for users, businesses, and developers to participate in this coming transformation of the Web.

Lexicon

ad hoc browsing—browsing without context.

browser add-on—a small program that is run inside the browser to modify the browsing experience in some way.

browsing episode—the browsing activities performed by a user in connection with accomplishing a given task.

claims—data elements stored in an information card.

context—all of the information relevant to a given browsing episode.

information card—a small cryptographically-enabled data file that represents an entity's identity.

Intersekt—the interactive development environment (IDE) that developers use to create KRL rulesets.

KIX directory—the directory of KIX that are available. The directory allows users to find cards that provide the functionality they desire.

Kynetx Information Cards (KIX)—information cards used inside KNS.

Kynetx Network Service (KNS)—the cloud service presented by the KRE executing KRL rulesets in response to browser requests.

Kynetx Rule Language (KRL)—a domain specific language that provides an abstract, linguistic means of specifying structured experiences through customization of browser content.

Kynetx Rule Engine (KRE)—evaluator for KRL rules that accepts requests from the browser and responds with custom JavaScript code that is executed to customize the experience.

selector—a container or wallet-like software application that runs on the user machine; used to manage information cards.

structured browsing—browsing in context using KNS.

Contact Information

Kynetx, Inc.
World Headquarters
3098 Executive Parkway
Suite 275
Lehi, UT 84043
(801) 649-4601
www.kynetx.com

Acknowledgements

Steve Fulling, Craig Burton, Judith Burton, and Kristen Knight were instrumental in developing the content and character of this white paper. The author gratefully acknowledges their contributions.

Endnotes

[1] One Billion Add-On Downloads, Justin Scott, Mozilla, http://blog.fligtar.com/2008/11/19/1-billion-add-on-downloads/
[2] Internet Challenges, April 19, 2005, http://www.windley.com/archives/2005/04/vint_cerf_on_in.shtml
[3] Information Card Foundation, http://informationcard.net/
[4] Goodbye, Passwords. You Aren't a Good Defense, New York Times, 9 Aug 2008, http://www.nytimes.com/2008/08/10/technology/10digi.html
[5] If you're wondering, this is a frog.
[6] The API is described online: http://wiki.kynetx.com/pages/Kynetx_Network_Services_(KNS)_API
[7] KRL is documented online: http://wiki.kynetx.com/pages/Kynetx_Rule_Language_(KRL)_Documentation