The Personal Web

Smart Internet for Me

Joanna Ng
IBM Canada Laboratory, Toronto, Ontario, Canada

Abstract

This position paper proposes the ***Personal Web*** as the next instantiation of the *Smart Internet Research Initiative.* Zooming in the problem scope of Smart Internet, the Personal Web focuses on the person *ME* as the center of gravity of web integration. The objective of the *Personal Web* is to empower *ME,* as a common internet user of generally limited technical skills, the autonomy and ease of control in assembling and aggregating integrate-able web elements across the web for a particular *sphere of context* of my concern. This should be so simple to do that becomes frequent everyday tasks. Typical integrate-able web elements are made available as extensions to current web sites by web domains' software programmers, to enable end users to drive and operate web integration by themselves. This user-sovereign web integration results in the generation of dynamic and high degree of personalized web artifacts for visualizations and interactions, synthesized together in a manner that is cognitively supportive; melding silos of web domains into a unified web platform to support the individual user's situational and/or persisted and repeated need for contextually relevant data and services without any programming requirement. Currently, the most popular solution towards such goal is mashups of various nuances which focuses on web sites, information and services web integration. The Personal Web proposes a people-centric integration of information, services and web content as a significantly different approach of web integration. This position paper asserts that this approach offers much better potential to realize the set goals of the Personal Web than the currently prominent mashups approach. The Personal Web is fundamentally built upon Resource Description Framework (RDF) based Linked Data of the semantic web by Tim Berners-Lee, with several significant adaptations and extensions of linked data proposed in the paper to resolve the current technical and practical limitations of mashups, in order to realize the Personal Web goals and objectives. Adaptations and extensions discussed in this paper include ***personal linked data; linked services*** and linked services scoped into ***personal linked services.*** In addition, a three-layer meta-model of the Personal Web is also proposed in order to enable more opportunities for the inference and discovery of semantic relationships between the user's sphere of context and the participating integrate-able web elements. With a defined meta-model of the Personal Web, simple user operations for web integrations can be defined and made available to users to control web integration for themselves without additional tools and widgets. I believe that the strategy set by the Personal Web will bring forth a significant step to improve users' lives by putting control in their hands in pulling together data, services and web content feeds from around the internet.

1 Introduction

Enabling general internet users, who typically have limited technical skills, the ability to pull data; services and web content together, from across the web as an integrated platform for the users' situational and/or persisted-repeated need, has been a long standing technological objective of recent web research and Web 2.0 development. There are various approaches to bring this into reality. The most recent and prominent approach towards such set goal is Mashups. Grammel et al. defines mashups as an end user driven recombination of web based data and functionality [11]. Mashups takes the approach of data, services and web presentation centric integration to solve user's need for information and services not originally anticipated by the web programmers [17]. However, constructing mashups typically involves the use of mashup tools that require deep programming skills that general internet users do not typically have, resulting in a very high barrier for adoption. To lower the barrier for adoption, widget user model for mashups is being developed recently as an alternative. Yahoo pipes and Popfly are some examples. While the widget user model claims that not a single line of code is required to be programmed, nevertheless,

these mashups widgets are abstractions of necessary programmatic constructs and are not easily understood by general users unless they have prior understanding of these programming concepts such as data flow, loops etc.. Therefore general internet users still find these mashups widgets complex to understand and use [20]. Newer approach such as Mashups with programming-by-example tries to give users more direct manipulations. However, the operations in terms of what users can do with the aggregated data are very limited [17]. The limitations and perceived complexity from general web users in the various nuances of mashups are inhibitors for general internet users to perform web integration by themselves as frequent everyday tasks. Even if users manage to survive the mashup creation phase, the data and services within each participating unit of integrate-able web elements are of very low degree of personalization, making mashups inadequate in realizing the set goals of the Personal Web.

Analyzing social networking platform from the perspective as a web integration platform as an alternative to mashup provides a promising progress towards mashups' limitations that is worth noting here. For example, Facebook started with a ***social graph***, a term coined by Zuckerberg of Facebook, to capture relationships among Facebook users. Building on top of this, in April 2010, the Facebook social platform launched an Open Graph protocol as a graph for "real-world things" such as movies; restaurants etc., which are, in essence, ***integrate-able web elements*** abstracted and extended from the source web sites. Web sites can be enabled for Open Graph by defining the subset objects from source web pages and abstracted them as "real-word things" into Open Graph objects through the prescribed Open Graph protocol. By adding XML tags to the site's source HTML, these "real-word things" as Open Graph objects can become stream feeds that can be integrated into Facebook's pages such as 'wall'. Facebook provides one user operation to enable user-sovereign web integration without programming requirement, which is the "Like" operation, implemented as a "Like" button with the associated social plugins. Facebook users can use the "Like" button to create a "Like" relationship between the Open Graph object as a stream feed and the user. Facebook user can also use the "Like" button from a web page outside of Facebook, such as the sample in Figure 1. As a result, users integrate web pages themselves by using the Facebook's "Like" button, resulting in the "real-world things" represented as Open Graph objects, imbedded as subset web elements from a web page of a web site to become a part of the Facebook page. It has domino impact cascaded down through the previously established social links of the Facebook's social graph. Open Graph is implemented as Resource Description Framework in Attribute (RDFa). According to Mark Zuckerberg, this is an important step towards the creation of a more semantically aware web [24].

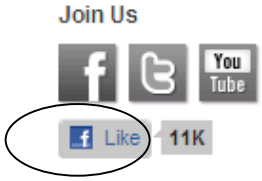


Figure 1 Sample Web Page with a Facebook "Like" Button

What is more significant, as per Zuckerberg's statement made in April, 2010 at the Facebook F8 Developer Conference, is that "My identity is first with Facebook and then is defined by things all over the web" [23]. This is the first closest claim towards a personal web implementation.

The notion of re-purposing social networking platform as a platform of web integration is still at its infancy stage, but with surprisingly tremendous success in terms of adoption rate. Despite of its short history, there are a few critical observations that are significant to note for its next iteration.

The first observation is that ***less is more in the meta-model of real-world things*** of integrate-able web elements. The meta-model of Facebook's Open Graph Objects and its data type is an extremely simple graph-based taxonomy. There were only a few categories of "real-world objects" such as *people; places; products and entertainment etc.,* with a very shallow taxonomy. There was only one link relationship in this graph: the "Like" link. Facebook users click on the "Like" button to specify "real-world objects" from web pages that they like. The user model cannot get simpler than this. User-sovereign web integration is achieved, yet so simple to use for general internet users that can now be done as frequent, everyday tasks. There is no programming requirement or complex mashup widgets. It drives the first massive real life adoption of an apparent semantic web site of pivotal significance. In this new paradigm of web integration, the social cascading effect because of the links established in the social graph is a critical novice element that the mashups approach cannot compare. All users from the Facebook social graph who have direct or inferred relationships with ME, as the Facebook user, are now made aware of these Open Graph objects because "I", the user, "Like" it. Cascading effect through the "Like" links can further ripple through the social network by my "friends".

On the contrary, open standards such as Web Ontology Language (OWL) are highly open and powerful in expression and rich in features and capabilities such as classes, instances with many link types. But the price of openness and expressiveness is complexity. OWL requires deep technical exper-

tise to use and therefore results in a high barrier for adoption by everyday users. While powerful, the user does not know what to do with it, inhibiting the goal towards end-user programming [16]. When such complexity of meta-model is exposed to the user, it tremendously inhibits adoption by every day users.

The second observation is that ***defining a universal, common set of integrate-able web elements with a simple meta-model is necessary.*** It is unrealistic to expect original web pages to participate in user driven web integration with its entirety. A subset of web elements has to be abstracted from its source web site and be normalized into a set of integrate-able web elements as units of participation in user driven web integration. Such integrate-able web element in Facebook's Open Graph is Open Graph Objects. Many web domains have already followed the prescriptive steps according to the Open Graph Protocol API to abstract subset web entities from their web pages into Open Graph Objects as integrate-able web elements. There is clarity and simplicity in the prescription of such process of abstraction and normalization that existing web sites know what to do in order to contribute their sites' subset web entities into integrate-able web elements. This is also critical for quick and wide adoption.

Almost all mashup solutions do not require the participating web domains in mashups to abstract subset web entities from their source web pages in order to participate. However, all the complexity of mashing, including the abstraction and normalization from source sites, is left to be dealt by end users at the mashups creation phase. Alternatively, in the proposed personal web approach, a current web domain abstracts a subset of data, services and web contents out of its source domain in order to create integrate-able web elements to participate in the Personal Web. User can control web integration without programming requirements by using simple operations to create web integration by assembling these integrate-able web elements from across the web into a cohesive entity.

The third key observation is that ***a simple, well defined meta-model enables system generation of visualization and interactions of web integration results.*** The well defined meta-model of Open Graph Objects enables system computed web pages into a synthetic entity for the visualization and further interactions of the integration result. The Facebook "wall" with all imbedded "liked" Open Graph Objects is a good example. In the same token, the proposed basic infrastructure of the Personal Web, using RDF based Linked Data, provides much better opportunities to automate system generated web artifacts for end users to visualize and interact with the results of the user-sovereighn web integration. This is pivotal in enabling a simple user model.

The fourth major observation is that an ***RDF-based, graph oriented meta-model is designed for integration and offers much better chances for open web integration with more depth and breadth***. The RDF-based Linked Data is by nature designed for open integration. With all integrate-able web elements of the Personal Web normalized into a common RDF representation, more open and dynamic aggregation across the web is now possible.

Harvesting from these observations, the Personal Web intends to provide "ME", as a general internet user, the power and control over situational or persisted and repeatable web integration that is relevant and appropriate to MY specific requirements of content, information and services, as per my current sphere of context, so simple to use that they can become effortless, frequent everyday tasks. Like Facebook, the Personal Web proposes a people-centric integration of content, information, services. In the case of Facebook, all web integrations are done and owned on the Facebook server side. Personal Web, on the other hand, started with "ME" and my current "***sphere of context***" that orientates the web integration, integration can happen on the users' ground.

The rest of this paper is organized as followed: Section 2 covers a proposed scenario of the operations of the Personal Web. Section 3 discusses related work in the context of assessing the distance from the goal of Personal Web. Section 4 calls out the key conceptual and technical principles. Section 5 describes the proposed extensions of the RDF based Linked Data in order to form the based infrastructure for the Personal Web. Section 6 highlights the future and a set of problems and research challenges. This chapter ends off with a discussion on the potential significance of the Personal Web.

2 A Scenario of Operation

To demonstrate the Personal Web concepts and to illustrate how the Personal Web may work, a proposed scenario of operations of the Personal Web is described in this section, both from the perspective of the participating web domains and the perspective of the end users. The scenario is still a sketch and will evolve as the Personal Web continues to mature. As concepts are refined through more studies; validations of research results and user feedback from implementations, more polished version will be developed.

"Personal-Web-Enabling" an Existing Web Domain

For each web domain to participate in this user-sovereign web integration as the Personal Web prescribes, the participating web domain first needs to identify and abstract the subset web elements from the current site, such as subset units of information; subset units of services and subset web

content as feeds, and normalize the abstracted subset into a set of integrate-able web elements as per the prescription defined by the meta-model and protocol of the Personal Web. When this process is completed, such web domain is considered "personal web enabled". A set of RDF based Linked Data Graphs that capture the set of integrate-able elements into a set of machine process-able form is created and made available. The resulting integrate-able web elements are to be used by the Personal Web users to drive web integrations conducted by them, integrating with integrate-able web elements from other personal-web-enabled web domains, in a manner that may not be anticipated by the originating web domains.

A Proposed Scenario of a User-Sovereign Web Integration: In the following proposed scenario of what an user-sovereign web integration may look like, the Personal Web users articulate *a matter of concern* as a "*sphere*" of context. Around this articulated context of matter of concern, a ***Personal Web Sphere*** for that concern is formed from the finite set of normalized integrate-able web elements that come from each member-URL-s, which have been personal web enabled. The finite set of member-URL-s is defined by the user or by affirming the system's recommendation based on the computed semantic relevance of personal-web-enabled URL-s to the stated matter of concern.

For example, a user, Jo, defines a ***sphere*** for "online shopping" in order to synthesized all her online activities across the web into one integrated entity. Jo only shops from two sites online, both sites have been personal-web-enabled. After defining a personal web sphere, called "Jo's online shopping", she added two online shops' URL as member URLs for this sphere. Figure 3 shows an instance of user interface of what this may be like.

Data and Service Retrieval:
Each time, when a member URL of a set personal web sphere has been added, the corresponding member-URL's data schema subset; the user's share of data instances; and the subset of web services, and a set of web feeds from that source web domain, which have all been normalized into a set of integrate-able web elements by the source web domains as per prescription, captured in a machine process-able form in RDF based Linked Data, are loaded into and made available for the user's personal web sphere.

Data and Services Integration:
All Linked Data graphs from all member-URL-s are further integrated into one graph at the scope of the 'sphere', by relationship discovery through inferences or other means, resulting in a macro view of all integrated member as one cohesive entity.

Data and Services Cleansing and Classification
User can cleanse data or services that one does not want. User can also add semantic relationships to make the integration more relevant to him and to his situation with a mixed initiative approach.

Figure 2 A Sphere of "Online Shopping"

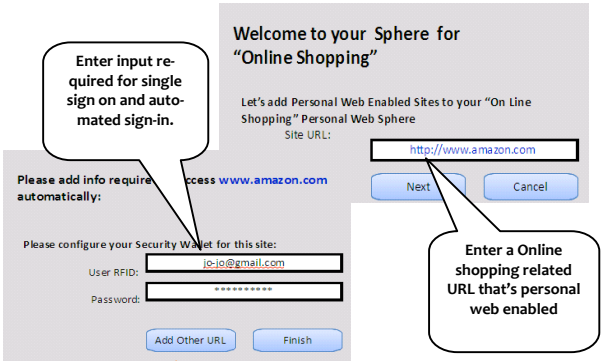


Figure 3 Adding personal-web enabled URLs to online shopping sphere

Data and Services Visualization and Interactions
Visualization and web interfaces of the merged RDF graphs from all member sites are be generated dynamically. There has been research work done in visualizing generated RDF graphs such as Haystack [13]. This allows the user to view and operate on the integrated data and services as one aggregated entity.

This proposed scenario of operations is a bookmark-like user model in the creation of personal web space as a way of user controlled web integration. While mashups has been viewed as a form end user programming of the web to enable web integration, the Personal Web proposed to take out the concept of programming from the control of general internet users in web integration and maintain the objective of user sovereign web integration with no programming requirement. It is asserted that this is significantly simpler than the mashup

web integration alternative which typically requires additional mashup tool or mashup widgets.

3 Related Work

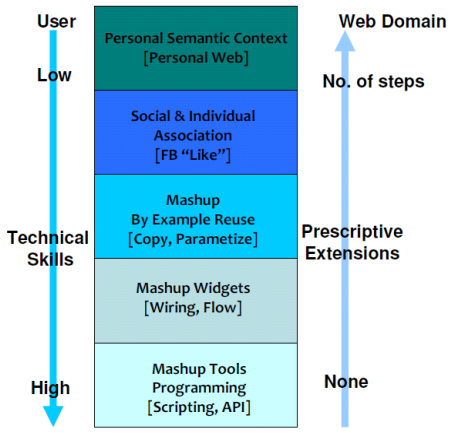
In the last five years, there has been rapid development in technologies towards the goal of enabling end users to aggregate web content; data and functions from multiple websites into an integrated unit, for a given purpose or to solve a specific problem. What makes this exciting and inspiring is the boundless innovative possibilities when the power and control over the web is put in the hands of end users, to create what may not be previously possible and anticipated. The survey and discussion in this section attempts to assess to what extent has such stated objective a reality for general non-technical internet users, to the point that they can perform such web integration with very minimum effort as frequent everyday tasks without any programming requirement.

2.1 *Mashups*

Mashups are the most popular tools of such enablement. Mashup is generally referred to as a web application that integrates data and sources from multiple sources to provide a unique service [20]. Studies have been conducted on the different kinds mashups designed for different functions and purposes. As a result, a set of abstracted mashup patterns have been identified [22]. In surveying the various models of mashups available today, Figure 4 shows the relationship between end user's skill requirement and the extension requirement of the participating web domains. While the potential of mashups to help users to solve their everyday problems is great, yet to this day, real world adoption for end user web integration as everyday activities is still not a reality.

Figure 4:

Progression of End User Enablement in Web Integration



The early offerings of mashups often require technical and programming skills beyond what one could expect from general internet users. For example, for Mash Maker to extract data from multiple web sites to be aggregated, an extractor has to be first created for such aggregation. The creation of extractor requires some programming skills [16]. Microsoft Popfly's involves a visual dataflow language. Google Mashup Editor uses extended HTML and Javascript heavily [11]. All these illustrate a high barrier for general internet users to freely and frequently construct data aggregation across web sites as their everyday activity.

The widget approach of mashup intends to eliminate the need of coding. Yahoo's pipes, Popfly are examples of this widget approach. Mashups widgets represent different web sites. Widgets are to be connected together through wiring. But such approach, while an improvement over the programming required for Mashup development tools, they have not yet able to offer satisfactory alternative. The number of existing widgets is usually just too many to be manageable and they are difficult to find. While it is true that with mashup widgets, users are not require to code or write scripts, however, mashup widgets are still too difficult for general internet users unless they have some understanding of basic programming concepts (such as loops; data types etc.).

Mashup by example is the latest development. It attempts to lower the deep technical skills required in the widget approach of mashup. System such as Karma enables its users to extract data from a web site into a data table through demonstration, and uses XPath generation to find similar data and copies into the table as well. However, such integration is only limited to data integration and is good only for ad hoc integration that does not require persistence. The requirement of service integration is not being addressed. System such as Vegemite is implemented spreadsheet like user interface to provide users direct manipulation [17], but has the same limitations in persistence and lack of support for service integration.

One can conclude that to this day, there has not been one single mashup model that is made available to general users, easy enough to be used for frequent, every day web integration activities, that can support robust and deep integration of data, services and web content feeds, all within one mashup model with high degree of personalization.

Facebook Open Graph

Recent research has proposed to leverage technologies established in semantic web to create solution that can relax the current limitations of mashup previously described [3]. The Facebook instance provides an interesting case study of using

a social networking platform also as a web integration platform. Facebook uses RDFa based "Open Graph Protocol". The Open Graph Protocol prescribes how a given web domain should "tag" the HTML of their site in order to enable the web page with "Like" button that their user can select. The site can also follow the Open Graph Object tags to create stream feeds with a "Like" button to participate as an imbedded object of face pages. When a web domain has followed the steps of prescriptive extensions of their sites according to what the Facebook Open Graph APIs have prescribed, Facebook provides its users a simple point-and-click user model to establish relationships between "real world things" and people of the social graph and network. With the connections established in the Facebook's social graph, a cascading effect of social impact is generated with the thing that one given Facebook user "Likes". The degree of technical skills required from the user of this web integration approach is very low.

Facebook is an interesting experimentation of merging web 2.0 with semantic web and it is not without its limitations and shortcomings. For examples, the available categories of Open Graph Objects are very limited in terms of comprehensiveness in representation of real-life objects. Also, the current meta-model of Open Graph Objects is limited to define a fragment of web feed by listing the set of references of images; text descriptions etc. from current web pages. The Open Graph meta-model falls short in defining objects for services and backend data such as data from relational databases to participate in this user-sovereign web integration. Scaling up, the requirement of security, such as authentication and fine grain authorization of Open Graph Objects and issues of privacy must also be addressed. At its infancy, critics of Facebook Open Graph are quick to point out that "Open" is not open because all aggregations have to be conducted by users of Facebook and must place on a Facebook server. While the simplicity of user operation of the "Like" button drives fast adoption, there are more relationships in the real world than the one and only "Like" relationship.

It is also important to point out that the mashups approach focuses on aggregation of integrate-able entities such as data, services and web feeds as different mashups types. For the longest time, technologists have attempted to define ontology that best models the *relationships between entity and entity*. The social networking approach, however, focuses on simple *abstraction of relationship between "a person" and his "integrate-able entities"*. The "Like" relationship is an answer to the question: how a user typically relates to this category of "things" to fulfill the purpose and reason of the user's web integration. In the case of Facebook, one of the purposes of web integration is to create an integrated page of all things that the user likes across the web. This realization provides invaluable insight in the consideration of adapting the semantic modeling to a model of *relationship between people and entities* instead of just focus on relationship modeling between entities and entities. All these lessons learned help to shape the Personal Web to realize its set strategy and goal.

4 The Personal Web Principles

This section attempts to capture an initial set of characteristics core to the Personal Web, both at the conceptual and technical perspectives. These principles are to be validated and adapted as the research and development of the Personal Web evolves.

4.1 The Conceptual Principles of the Personal Web

#1. The user "I" is the center of gravity for web integration

Facebook's web integration is oriented around the users of the social graph, for the purpose of aggregating social impact of all real-world objects liked by an user as an object of the social graph, cascading through the established "a friend of" relation through out the social graph. The Personal Web, on the other hand, orientates web integration around the user as the center of gravity for web integration. All data about "ME" across the web, such as all user IDs and passwords, all account numbers across the web that belongs to ME such as frequent flyer numbers, can be aggregated into an integrated data entity represented by one consolidated Linked Data GraphThis "ME" object with the practical benefit of sparing ME as an user, from repeatedly entering static data entries such as user ID and password; account information etc., every time, resulting more pleasant user interactions and experience [19].

#2. My Context Shapes the Scope and Semantics of Web Integration

Dey defined context as any information that can be used to characterize the situation of a person, place or object, in order to bring the most relevant content and services, for the user's benefit of the situation [9]. The Personal Web performs web integration around a user, and the user's personal semantic context, called "Personal Web Sphere". A little semantic goes a long way [14] for the discovery of semantically relevant data; services and web feeds for a specific context that the user specified.

#3. The user "I" is completely sovereign in controlling the web integration

Building on the Web 2.0 assertion of the Web is a platform, this principle of personal web takes a step further to see the web as one concerted platform to support MY every day situ-

ational or persistent and repeated need of data and services, that I, as the general non-technical users of the internet can do as frequent, everyday tasks. Mashups is sometimes being referenced as end user programming of the Web. The programming concept is to be removed from the Personal Web and instead, it asserts that the user model of the Personal Web has to be so simple that frequent everyday web integration totally operated and controlled by general internet end users is made possible without any programming requirement.

#4. Let the Web works on my behalf

The cognitive load reduction is one of the major goals of the Smart Internet. More advanced features of the Personal Web will be developed to include these concepts such as enabling an individual user to persist some previously persisted form of web integration as scheduled tasks or events as prospective memory tasks; or to set a certain persistent and repeated task sequence in automated into autopilot mode, running in the web platform as a batch processes with the user plays the role of a supervisory controller [1].

#5. It has to be a social web after all

It is important not to lose sight of the important fact that the special user "I" is also a part of the social network. While there is no concrete validation of user requirement, but hypothetically, sharing my personal web spheres with my friends on the social network, or my friends invite me to participate in his personal web sphere are some possible, futuristic scenario. Asking my friends what URL-s are most revelant to my Sphere is another practical scenario to add. The domino impact of the social network in influencing the adaptation of the most optimal and effective web integration for a given semantic context for a particular profile of users is too important to ignore.

4.2 The Technical Principles of the Personal Web

This section calls out the technical characteristics of the Personal Web architecture and design. These technical principles are intended to drive design and implementation decisions. Abiding to these technical principles in all components of Personal Web implementation is pivotal to its real world adoption.

#1. The Programming Goal of the Personal Web is to abstract out the programming complexity for its users so as to Pass Control from the programmers to the End Users

The goal of any API protocol and programming model of the Personal Web is to produce intermediary web artifacts, usually in machine readable and process-able form, for the sake of enabling the Personal Web users the control to aggregate these parts into something of a bigger context as the user so desires. The design goal of the Personal Web components and intermediary artifacts is to produce independent units of web composition which are open for integration by design of the software engineers who produce them. By putting these units of composition in the hands of general users, with a simple set of user operations defined for user sovereign web integration, end users have actual control over how the web as a platform can work for them, which is not possible with today's monolithic style of web applications.

2. Less is More in meta-model for semantics and user Operations for web integration

This principle can be applied in different components of the architecture of the Personal Web. For example, general internet users cannot handle the many great features of Web Ontology Web (OWL) to express relationships. Even high skill technical users need some time to compute if a given OWL instance as an 'individual' of a certain member class is "equivalent to" another class but "disjoint" with some other classes. This is a drastic contrast to Facebook's one and only relationship of "Like", from which simple user operation of the "Like" button can be derived that general users with limited technical skills can perform with minimum effort.

#3. Design for Open Integration

It is also important to abide by the principle of openness and standards. The intermediary of web artifacts of the Personal Web is designed and architected to be a part of something else that was not originally known and anticipated. This critical requirement of openness for integration is the major reason that the RDF Linked Data is the selected candidate of technology infrastructure instead of the mashup widgets of many nuances as the foundation of choice for internal representation.

#4. Model NOT relationship among things, but model people's relationship with things

Current usage of ontology seems to focus on the modeling accuracy of concepts and objects relationships in representing knowledge of the real world. Ontology language such as OWL has been, for decades, focused on modeling the real world things and knowledge with high degree of accuracy but completely lost sight of the importance of ***why*** and ***how*** people relate to things in real life. The "Like" link of Facebook is profound in the sense that it models how people relate to things in real life according to the user's reasons of web integration, instead of how "things" relate to each other. For example, while it is true in knowledge representation that shop carte "is equivalent to" shop bag, it is more important to assert that I, the user, can "join" them as one semantically for my purpose of seeing all my shopping entries online as an

integrated entity. In this example, the "Join" relationship that models the purpose of my web integration has more significance than the knowledge of shop bag and shop carte are equivalent in semantics. More research needs to be done in this area in people-and-entity relationships in order to define well-abstracted and useful user operations for web integration.

5 The Basic Enabling Infrastructure of the Personal Web

The basic enabling infrastructure of the Personal Web is ***linked data***, whose objective is to make the Web a "***Global Data Space***" by enabling data published from diverse sources of web domains to be connected with formally typed connections, in order to make the web a more intelligent source of information for its users [7]. The original goal of linked data is to "stitch together the world's structured information and knowledge to answer semantically rich queries" [4].

Linked Data is built upon ***Resource Description Framework (RDF),*** developed with the motivation to provide an open information model that can interwork and combine data from multiple disjoint sources in order to derive new information that can be processed automatically at the internet scale [21]. The underlying structure of any expression in RDF is a collection of triples. Each RDF triple consists of a ***subject, a predicate and an object.*** Each triple represents a statement of a relationship, which can be represented by a directed graph. The meaning of a given RDF graph is the conjunction of all assertions, represented by all triples it contains. As a result, it provides a generic, graph-based data model to describe things in the world [15]. Linked Data builds on top of RDF by defining formal ***typed links*** as connections between data from diverse domains using the construct of ***predicate*** from RDF in order to link arbitrary things in the world to create "the Web of Data" [7]. The subject, object and predicate of the RDF triple are all URI-s. Each identifies a resource or a string. Predicate's URI specifies how the subject and object are related.

The notion of developing personal information systems consisting of focused subsets of information, highly relevant to a particular user in order to deal with information overload of the user from the web platform, dates back at least as far as the personal web space vision of Abrams et al [1] in 1998. While Abrams et al focused on personal bookmarking, others looked for enhanced forms of search, but still within a linked data approach. However, the current view of the Personal Web goes beyond that earlier linked data approach to consider linked services that support not just information exploration and management, but many other tasks as well.

This paper proposes to extend the ***typed links*** of ***Linked Data*** in order to a build an infrastructure that instantiates the strategy of the Personal Web into reality. The adaptations and extensions proposed can be summarized into the following parts: #1 Re-purpose the global linked data into a personal scope of linked data called ***the Personal Linked Data***. #2-a. Extend the participating object nodes of the RDF to include Restful services that this paper terms ***Linked Services***. #2-b. Re-purpose the global scope linked services into the ***Personal Linked Services.*** #3. Integrate the *personal linked data* and the *personal linked ser*vices into one RDF linked data graph as one ***Personal Web Link***. #4. Provide an initial ***meta model*** to define and normalize these integrate-able web entities of the Personal Web into an open, universal model, prescriptive for current web domains to follow in order to extend the current site to take part in the Personal Web.

5.1 The Personal Linked Data

It is important to note that the Personal Web has a different scope and objective than that of the original Linked Data. The original concept of Linked Data is to connect local data for the global web. Personal Linked Data of the Personal Web connects user's global data for and about the user, as an individual. The vision, goal and objective of the Personal Web, from a pure data perspective, is a ***gathering of global data about me, for me and relevant to me across the global web into a synthetic, integrated data entity of ME across the global web.*** We called this the "***Personal Linked Data***".
From the scope and perspective of a given web domain, the web domain's *Personal Linked Data* for a given user is the proper subset of data entities that belongs to the user and/or public data subset relevant to the user, including structured and unstructured data, as well as web content feed, from the source web site, normalized into one RDF Linked Data graph in representation. For a web domain to be personal-web-enabled, the web domain needs to define the mapping for a given user's proper subset of data from the web domain. All user's proper subset of data, including structured, unstructured and web content feeds that the web domain owner defined for the users will be normalized into a linked data RDF graph for that user.

From the user's perspective, the user can gather Personal Linked Data RDF graphs from several web domains and group them together into one semantic context. In this way, multiple Personal Linked Data RDF graphs can further discover and infer relationships and links across the graphs, and also how they relate to the semantic context, which captures the user's purpose and intent. Joining these multiple Personal Linked Data RDF graphs, from across web domains but se-

mantically grouped within a set context, enable simple but user driven web integration.

5.2 Linked Services

Another extension from the original Linked Data proposed; is to extend the original concept of linked data beyond the scope of data and information but transposed to and applied in the scope of ***web services*** *as units of web functions.* In other words, the ***subjects; objects and predicate*** of the RDF triple and its typed links are used to express Restful service. Such extension of the Linked Data for the representation of Restful web services is being termed as ***"Linked Services"*** in this paper. Fielding's architectural design of restful services [10] in 2000 constrained each service as an URL, making restful services fit architecturally to be integrated and to participate in the linked data graphs of the Personal Web. Recent research has made progress in defining different methodologies to normalize different kinds of web services into Restful services, with the representational URI as resources. As a result, "services" as user tasks, once being normalized into restful services, can be transformed into important participants of the Personal Web. Theoretically, like its linked data counter part, "linked services" is default to the global web scope.

5.3 The Personal Linked Services

In reality, having a global scope of services is not practical or realistic. This paper asserts that services are often invoked and executed by and for a web user within a very well defined context; and the therefore, there is a much higher value proposition when the concept of "***linked services***" is applied to an individual personal web scope of ME as a web user instead of to the global scope of services. Such transformation is being termed "***personal linked services***".

From the scope and perspective of a given web domain, the web domain's *Personal Linked Services* for a given user is a proper subset of services entities that the web domain wants to make available to its users as integrate-able units of functions and tasks, normalized into one RDF Linked Services graph in representation, according to the meta-model of the Personal Web. What makes the linked services personal? Not only the choice and combination of services can be individualized, but the input resources, because the user is known, can be pre-set as per the identity and the data of the set user.

From the user's perspective, the user can gather Personal Linked Services RDF graphs from several web domains and group them together into one semantic context. There is great potential for future work to explore how to enable, and what user operations of the user model for the user to express the sequencing of these services across web domains, resulting in an extremely simple, user-sovereign process management web tool without extra tool widgets.

5.4 The Personal Web Sphere

The semantic context, which represents the reason and purpose for a user's need to perform web integration, driven and control by end user, is called a "sphere", as a sphere of context for integration. In other word, the "***sphere***" is the center of gravity that the user creates, as a unique individual, as the gravitational factors for web integration. The semantic links and relationships related to the 'sphere' drive the inference and discovery of related integrate-able web elements, resulting in a much deeper integration in web resources that yield a high degree of personalization that common user can handle as frequent, everyday tasks.

5.5 The Three Layered Meta Model

It is important to establish a meta-model for the Personal Web, upon which a universal and open framework can be developed. This is pivotal in enabling current web domains to extend their current web sites in a prescriptive manner in order for them to take part in the Personal Web. With a model to normalize to, end users can now operate with integrate-able web elements which are designed to be a part of something that may or may not be previously anticipated. In addition, with this defined meta-model, automatic generations of web artifacts for user visualization and interaction of the integration outcome can be developed. Automatic discovery of links and relationships through inferences and other means are then made possible.

This paper calls out three primary categories of integrate-able web elements for the Personal Web as participating units in this user-sovereign web integration, designed to be open and integrate-able with each other. They are, namely, (i) units of data, both structured and unstructured; (ii) units of web content in the form of feeds, (iii) units of services in terms of functions and capability. All three categories are represented as RDF Linked data objects in its machine process-able form. The strategy is to define one set of simple universal user operations for the Personal Web users to synthesize instances of all categories of web elements into something novice as a new web integration approach that end users can do without any programming requirements.

Facebook has a preliminary deployment with its Object Graph Objects, which in essence are units of web content in the form of feeds, being imbedded with other objects as an integrated Facebook page. The Personal Web proposes to add the categories of units of data and units of services to take this web integration approach to the next level of maturity. In

addition, a simple system to associate semantics and meaning to these integrate-able web elements is also proposed, with such, opportunities and accuracy in links and relationships discovery within and across RDF based linked data graphs will be greatly increased. A mixed initiative approach is suggested for such extension. Such web integration infrastructure has high potential to result in an increased level of depth, breath and openness of web integration, with a high degree of personalization, and a low barrier to general users that its mashups approach cannot compare.

This paper proposes a three-layer Personal Web meta-model by leveraging Chen's Entity Relationship (ER) model that was originally intended as a top-down database design tool. The ER model's three-levels of progressive perspectives of data, (namely, the "entity" level as the top conceptual level of abstraction; the "data structure" level as the concrete level of the previous layer, and the "data record" level as data instance level) is being adopted [8]. The Personal Web's ***Concept*** layer; ***Model*** layer and ***Instance*** layer come from such adoption (See Figure 5), with RDF nodes in each layer properly typed as ***concept-nodes; model-nodes and instance-nodes***. The relationships and links between RDF object nodes within each layer and across layers are expressed as RDF predicates.

5.5.1 The Model Layer and the Instance Layer

#1. Web Content as Units of Integrate-able Object Nodes
The Data models and schema of various units of web content should be captured as captured as ***web-content model object node***. The XML schema for the Open Graph Object is an example. Many instances of such Facebook feeds were created by the participating web domains, and should be captured as ***web-content instance object nodes*** within the meta-model system of the Personal Web.

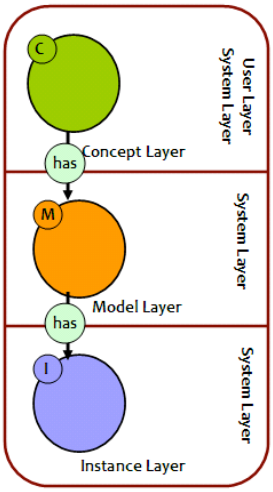


Figure 5 The Personal Web's Three-Layer Meta Model

#2. Data as Units of Integrate-able Object Nodes
The structured data schema from the source web domain (e.g. the data schema of a shopping carte from www.shopx.com) should be captured as ***data model nodes*** in the model layer. The actual data instances (e.g. the actual shop carte data records of purchases by user "Jo" from shopping site www.shopx.com) are captured as ***data instance object nodes***. All levels of traceability are maintained.

Future work needs to analyze and study real life interactions between people with these object models and instances in order to discover and abstract the most engaging and useful people-object relationships. Such findings will provide great insights for the definition of new link-types for simple user operations of creating purposeful web integration with no programming requirement. This paper asserts that these people-object relationships are ***real-world verbs*** (such as "Like" in Facebook) and can be represented as RDF-predicates**.**

#3. Services as Units of Integrate-able Object Nodes
The service meta-models should be captured as service models in the model layer. Two important service models should be called out, namely, the ***synchronous and asynchronous services***. The corresponding actual deployed service instances are captured in ***service instance object nodes***.

Future research work will refine and validate these service models for synchronous and asynchronous services. The initial sketch of a service model, kept in the model layer, for synchronous services includes a *resource node* and a *task node* with ***create, update, read and delete*** as verb-like predicates. The initial thought for an asynchronous service model is contains a *resource node* with a *task node* to be linked with ***publish and subscribe*** verb-like predicates. More real life deployment is needed to validate such service model in RDF representation.

5.5.2 The Concept Layer

Unlike Chen's ER model, the concept layer functions more as an ontology or taxonomy layer than the data entity abstraction of the original ER design [8]. Its main mission is to enable the linking of multiple model nuances in the model layer with one or more instances of a concept node in the concept layer. Since traceability is maintained across all three meta-model layers, algorithms can be designed to synthesize a unified data from disjoined silo web sources into integrated operations (e.g. "checkout" all shop carte for all member URL-s of my shopping sphere). Nuances of similar concepts can also be linked at the concept layer as well.

Ontology and data integration problems are well known issues. Each source web domain has its local conceptualization, ontology and 'tagging' system being handcrafted by database administrators and domain knowledge engineers. They are disconnected and disjoint from other web domains [12]. It is important to look into how semantic heterogeneity is being supported to achieve data integration practical to the users [5]. General user should also be given an opportunity to tag and map these concepts for one's personal matter of concern [18]. A mixed initiative approach in which both system automation and user participation is being proposed. Web domain may create and link some concept nodes with the corresponding data model object nodes to enhance user's web integration.

At this early stage, it is important to call out the concept node of "ME" and "SPHERE" are predefined and are special concept nodes for web integration. Future work is to be done in this and to be discussed with more detail.

Concept Nodes for the Real World Modeling: People, Places, Events, Tasks and Things

Modeling relationships between "things" only answers the question of how real world objects are related to each other. But it does not answer the question of how "people" relate to "objects" or "things. By carefully analyzing and abstracting relationships between people and things, and empirically answering the question of how/why "people" use/relate to "things", there is a better chance that more purposeful web integration for user's problem at hand can be enabled.

When people relate to things centers around a given context, the temporal and location dimensions are essential aspects. In this light, concept nodes are classified into concept nodes of the following category: People, Places, Events, Tasks and Things, Future work is needed to validate such taxonomy of concept nodes.

Types of Predicates as real world verbs

With what has been previously established, the RDF triple of subject-predicate-object can be applied to represent the ***people-action-thing pattern*** in answering the question of how/why people relate to things. The "Like" link of the Facebook Open Object Graph Object is a great example of such pattern. The predicates of the concept nodes are verbs as action words in modeling real world concepts.

6. Discussion and Future Work

This position paper is to set the vision, strategies and goals of the Personal Web and proposes to use RDF Linked Data as its technology base. Unlike the original strategy of Linked Data, which connects local source data in the global web, the Personal Web re-orients web integration and connects the global web data for and about ME in my personal web sphere to synthesize ME in the global web. Personal-web-enabled sites provide integrate-able web elements in a machine readable form of a RDF graph, with simple user operations that requires low technical skill level, that "I" the user can link with other RDF graphs from other silos web domains as a form of web integration in a manner that is distinctively different from the prominent mashups approach. While such novice web integration approach has great potential, there are lots of remaining challenges and open research issues related to the enabling of the Personal Web. The major open research areas are pointed out in the following. It is not claimed to be comprehensive.

6.1 Personal Linked Data

There are many open research challenges related to the concept of Personal Linked Data. The first challenge is to define a practical framework in order for existing web domains to be able to identify and capture the data model subset mapping of structured and unstructured data that will be of high value for its users' personal web sphere integration. The second challenge is that even if such data model subset is identified, what approach should be taken to identify the data instances subset mapping that captures only the data instances that belongs to a specific user. The third challenge is how to normalize the identified structured and unstructured data into RDF and yet preserves the original semantics that can promote links discovery. DBpedia has established a framework of such transformation [6]. However, the normalization loses some degree of items independence, resulting in the loss in semantics and opportunities to discover new relationships.

6.2 Linked Services and Personal Linked Services

Research in the relationship of RESTful services and Linked Data is still very new. Recent research proposes to describe RESTful resources as semantic resources as in RDF Linked Data, with the strategy to harvest the data that's already out there [2]. The strategy of Linked Services is to use RDF Linked Data construct to express the service models of synchronous and asynchronous RESTful services in a form that they are normalized into units of integrate-able web elements that end users can control.

This creates several interesting research challenges. First of all, independent from the goals of the Personal Web, normalizing web services of all models into one common RESTful

services model is by itself a significant challenge. Secondly, expressing RESTful services models in RDF Linked Data Graph, that can become unites of web integration such that end users can operate with, needs careful analysis and proper abstraction. However, if significant progress is made, user can control without programming requirements process flow such as sequences of services call, or to create script-based like agents for auto-pilot mode execution. How to make a well represented RESTful service in RDF Linked Data personal? This should be studied, not only to cover individual user's personal choice in the list and combination of services, it should also cover how an service instances can be preloaded with the user's personal data in order to improve the simplicity of service calls.

6.3 Lightweight but Expressive Semantic Infrastructure

Learning from the Facebook deployment example, an extremely simple data model and semantic infrastructure is pivotal in real world adoption. A lightweight but expressive semantic infrastructure is critical to the Personal Web in order to build associations between concepts, such as online banking, with the integrate-able web entities, including web feeds; data; and services.

One possible technology choice is Bergamaschi's "Common Thesaurus", constructed by analyzing Object Descriptive Language (ODL) and generalized into a very simple relationship model. There are three types of relationships in Common Thesaurus: "Synonym-of" and "Related-term" are the two symmetric relationships. "Broader-term" is the non-symmetrical relationship [5]. Adding mapping rules and attributes to this simple relationship model enables query and visualization of data source across different web domains. Other work done in this area is ConceptNet, It has a more elaborate relationship model built on WordNet with about twenty relationships [18]. More studies need to be done to define a practical semantic infrastructure that is simple to use, like the Common Thesaurus, yet expressive, like the ConceptNet.

6.4 Visualization and Interaction of User Sovereign Web Integration

Visualization of a complex integrated RDF graph by applying visualization techniques that is semantically accurate but slick in the eyes of the user is a key to general adoption and is an important area of research. Developing a framework to parse machine readable RDF and transform a graph based representation into a user friendly visualization artifacts that users can interact with is an important problem to solve. Future research work cannot ignore the studying of adaptability based on factors of the context such as location, device type.

6.5 Practical Issues

It is also important not to lose sight of the non-technical but practical issues such as the trust of the user: will general user trust a Web integrated ME to reside on the server side or worrying about losing identity if it lives on the user's device? There are also privacy considerations. How much identity of me could be stolen if I lose this integrated ME object? How much trust factor in the user to depend on the generated integration results? It is desirable to have a life-deployed instance of the Personal Web to provide some empirical, real life data of usage and feedback to make progress.

7 Conclusion

The Personal Web intends to provide "ME", as a general internet user, the power and control over situational or persisted/repeated web integration that is relevant and appropriate to the user's concern of the moment. The set goal is that it should be so simple to do that become general internet users' frequent, everyday tasks. We all know that this is not a reality yet. One has to conclude that the barrier of adoption is high due to its complexity. An entirely different paradigm of a social network oriented approach in web integration, using Facebook as an example, has great adoption with simplicity in user model. This paper proposes to learn from the case study of Facebook in order to propose yet a newer paradigm of web integration. Instead of a web integration oriented primarily around a social network, the Personal Web proposes a paradigm of web integration primarily orientated around ME as the user and my semantic context as a "Sphere" that scopes the integration gravity. While ME as a user is also a part of the social network, it comes second in priority after ME as the web integration driver. The Personal Web technology infrastructure is based on RDF Linked Data. The openness of such infrastructure offers tremendous opportunities with promising potential that this proposed strategy would result in web integration with more depth, breadth and degree personalization that users can do as a frequent every day tasks that the mashup approach cannot compare. Mashups today can be categorized into three typical categories: web content feed, information and services. There is no mashup today that integrate all three categories of web integration in one cohesive system to be put in the user's hand. The Personal Web claims to provide all three categories integratedly, which current mashup approaches cannot compare..

Acknowledgments

I would like to give my special thanks to Dr. Mark Chignell, Dr. James R. Cordy and Dr. Yelena Yesha for their great contributions in many ways to evolve Smart Internet into its next instantiation of the Personal Web. Your dedication and enthusiasm in pushing new ground in these new and exciting areas of research is energizing and inspiring. I would also like to thank Anatol Kark and Emilia Tung for their tremendous support. I thank the CAS Technology Incubation Lab. team members and CAS Research Staff Members, Jimmy Lo and Alex Lau, for their technical input and excellence in this project.

About the Author

Joanna Ng is currently the Head of Research at IBM Canada Software Laboratories, Center for Advanced Studies. She is also a Senior Technical Staff Member of IBM Software Group. She has held various senior management and architect positions in product development and software strategy within IBM. Joanna is an IBM Master Inventor with a long track record of profitable innovations. She has been granted over twenty five patents from various countries in research areas related to Smart Internet such as mobile commerce, voice-enabled portal, commerce portal, retail industry solutions; service-oriented architecture (SOA); asset repository; semantic and web technologies.

References

[1] Abrams, D., Baecker, R. and Chignell, M., "Information Archiving with Bookmarks: Personal Web Space Construction and Organization", CHI Conference, 1998.

[2] Alarcon, R. and Wilde, E., "Linking Data from RESTful Services", LDOW, April, 2010.

[3] Ankolekar, A., Krotzsch, M., Tran, T., Vrandecic, D., "The Two Cultures", ACM WWW 2007.

[4] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z. "A Nucleus of a Web of Open Data", 2007.

[5] Bergamaschi, S., Castano, S., Vincini, M., "Semantic Integration of Semistructured and Structured Data Sources", ACM Sigmond Record, 1999.

[6] Bizer, C., "DBpedia – a Crystallization Point for the Web of Data" Web Semantics: Science, Services and Agents on the World Wide Web, Volume 7, Issue 3 p. 154-165, September 2009.

[7] Bizer, C., Heath, T., Berners-Lee, T., "Linked Data: The Story So Far", International Journal on Semantic Web and Information Systems (IJSWIS), 2009.

[8]Chen, P.P.S., "The Entity-Relationship Model – Toward a Unified View of Data", ACM Transactions on Database Systems, Vol. 1, No. 1, p. 9-36, March 1976.

[9] Dey, A.K., "Understanding and using context", Personal and Ubiquitous Computing, 5(1), 2001, pp. 4-7.

[10] Fielding, R.T., "Architectural Styles and the Design of Network-based Software Architectures", Doctoral dissertation, University of California, Irvine, 2000.

[11] Grammel, L., Storey, M. , "A Survey of Mashup Development Environments", University of Victory, 2010.

[12] Guarino, N., "Formal Ontology and Information Systems", Proceedings of FOIS'98, Trento, Italy, IOS Press, pp. 3-15, June 1998.

[13] Huynh, D., Karger, D., Quan, D, "A Platform for creating, organizing and visualizing information using RDF", Semantic Web Workshop, 2002.

[14] Hendler, J., Keynote of International Semantic Web Conference, 2003

[15] Klyne, G., Carroll, J.J., McBride, B., "Resource Description Framework (RDF): Concepts and Abstract Syntax", W3C Recommendation, Retrieved http://www.w3.org/TR/rdfconcepts/

[16] Ko A.J., Myers, B.A., Aung, H.H., "Six Learning Barriers in End-User Programming Systems", VLHCC'04: Proceedings of the 2004 IEEE Symposium on Visual Languages – Human Centric Computing, pages 199-206, IEEE Computer Society, 2004.

[17] Lin, J., Wong, J., Nichols, J., Cypher, A., Lau, T.A. "End-User Programming of Mashups with Vegemite", ACM IUI'09.

[18] Liu, H., Singh, P., "ConceptNet: a Practical Commonsense Reasoning Tool-kit", BT Technology Journal – Vol 22 No 4, October, 2004.

[19] Ng, J., Nigul, L., Litani, E, Lau, D., "End user controlled web interaction flow using service oriented architecture model", The 2nd IEEE Workshop on Enabling the Future SOA Internet (EFSOW), IEEE Globecom, 2008.

[20] Tuchinda, R., Szekely, P., Knoblock, C.A., "Building Mashups By Example", ACM IUI'08.

[21] W3C, "Resource Description Framework (RDF): Concepts and Abstract Syntax, W3C Recommendation, 10 February, 2004.

[22] Wong, J., Hong, J. “What Do We ‘Mashup’ When We Make Mashups?”, ACM WEUSE IV’08.

[23] Zuckerberg, M., Facebook F8 Developer Conference, April 2010.

[24] Zuckerberg, M., Taylor, B. “The Open Graph Protocol”, www.opengraphprotocol.org, April 2010.