Managing Dynamic Context to Enable User-Driven Web Integration in the Personal Web

Norha M. Villegas, Hausi A. Müller
University of Victoria, Victoria, British Columbia, Canada

Joanna Ng, Alex Lau
IBM Canada Lab, Toronto, Ontario, Canada

October 15, 2010

Abstract

The *Personal Web* is the people-centric instantiation of the Smart Internet where information systems, services and web content are articulated by users according to their matters of concern. To realize the vision of the Personal Web, the Smart Internet requires infrastructure to support the user in the integration of personal data and the composition of personal services within a highly dynamic context that constitutes the user's *Personal Web Sphere*. To address these requirements, we propose a user-driven context management framework, built on the top of the basic enabling infrastructure of the Personal Web, to support users in the run-time modification of *personal context models*. The core of our proposal is the management of monitoring concerns by implementing feedback loops, where the user acts as the planner of the controller to adapt the monitoring strategy by means of using web interactions to modify the personal context models. These context models, deployed at three different levels of abstraction, represent monitoring concerns by defining abstract types of contextual entities, the relationships among them and the interactions that the user can instantiate to drive web integration.

1 Introduction

The *Personal Web* is a concrete realization of the Smart Internet that focuses on the user as the center of web integration [7]. Thus, Smart Interactions are required to support, from a people-centric perspective, the discovery, aggregation and delivery of resources from the Internet. Moreover, Smart Services must provide the infrastructure required by these interactions to assist users in web integration [8]. The vision of the Personal Web is to enable regular web users (i.e., people with non-specific technical skills) to control the integration of web resources according to their personal matters of concern. The semantics that define relevant types of web resources, the relationships among them, and the way how the user interacts with these entities is defined as the *Personal Web Sphere*. In this way, the user's personal web sphere defines the environmental context that affects the web integration performed by the user.

Due to the complex, volatile and transient nature of context, the Personal Web requires innovative approaches to assist the user in managing the information that can affect the interactions with web resources. To address this complexity, we exploit feedback loops and context-awareness techniques to propose a context management framework that supports and empowers the user in integrating web resources (web context entities) according to personal matters of concern. Our approach, built on the top of the Personal Web's enabling infrastructure proposed by Ng [7], supports the user in the modification of personal context models to drive the integration of web resources at run-time. These personal context models, modified by the user through simple but meaningful web interactions, feed our context management infrastructure back to discover personal web-enabled context instances that can be integrated to the user's personal context models, feeding back the context management system.

In this paper, we propose a user-driven context management framework to discover personal context entities and enable the user in the dynamic integration of these entities to her personal web sphere. With our approach, we support four of the five conceptual principles and the whole set of technical principles defined by the vision of the Personal Web [7]. With respect to the conceptual principles, (i) the user is the center of the web integration acting as the planner of our context management controller; (ii) the scope and semantics of the web integration is controlled by our context management framework according to personal context models (the user's personal web sphere); (iii) context-aware interactions between the user and entities of the real world are controlled by the user to accommodate context management concerns according to every day situations, by means of adapting personal context models; and (iv) these context management concerns are persistent in the form of personal context models that are re-used every time the user interacts with the corresponding personal web sphere. Regarding the technical principles, (i) having personal web-enabled context entities (web resources), the interactions defined by personal context models enable the user to compose desired entities into her personal web sphere readily; (ii) as our framework is based on the basic enabling infrastructure of the Personal Web, characteristics of design for open integration and simplicity of operations for web integration are preserved; and (iii) our modeling approach, deployed on the Personal Web's three layered meta-model, supports the instantiation of interactions between the user and instances of context entities, as well as the relationships among these entities that are relevant for the user's interactions.

The remainder of this position paper is organized as follows. Section 2 describes the scenario used throughout the paper to explain the components of our framework. This scenario will drive our validation. Section 3 explains our context management framework, as well as the integration of the basic enabling infrastructure of the Personal Web and its three-layered meta-model into our proposal. Finally, Section 4 discusses important aspects of enabling user-driven web integration by means of dynamic context management.

2 A Motivating Scenario

Imagine a user who is working on her web-based calendar. Her current concerns are related to two particular events, the anniversary dinner with her husband and her next business trip. Her *personal web* is required to assist the user in different context-aware, on-line activities concerning each event. Whenever she works on a particular event, the *smart Internet* supported by our *dynamic context management infrastructure* ought to suggest personal web-enabled data and services according to her matters of concern. These personal web-enabled elements, which we call *personal context entities*, are displayed on her web either because they are already within her personal web or because they are related to her current situation. Moreover, personal context entities previously existing in her personal web have a defined relationship with the user. On the contrary, personal entities suggested according to her matters of concern only will become part of her personal web when the user defines a relationship with them. These relationships are defined by the interactions between the user and the personal web enabled context entities discovered by our context management infrastructure at run-time.

Imagine the user is working on the anniversary dinner event. Her personal web displays linked data such as the invitation email sent by her husband to her. This email is a personal context entity (already existing in her personal web) that contains context information such as the restaurant where the dinner will take place. Based on her *personal web context models* enabled by our context management infrastructure, it is possible to suggest web services such as on-line shopping services according to the user's preferences and situation. Suppose she interacts with her favorite designer's fashion boutique web service. As the woman is already registered in the boutique's system, this web service previously linked to her personal web displays a custom dress catalog taking into account the current user's matter of concern (e.g., context information related to the dinner, location and clothing preferences). Then, she places an on-line order buying that new fancy dress she wants to wear for the anniversary dinner. As she pays using her credit card, the on-line banking service is also added to her personal web for future on-line shopping activities. Finally, while the user browses the boutique's web site, she interacts with some of the boutique's partner web services to order shoes and accessories suitable for her nice new dress. These URL resources are also added to her personal web as new *personal linked services*.

Because users' concerns vary according to current situations, one user can have more than one personal web instance. Suppose our user gets a calendar alert concerning her next business trip. Using her personal web models for traveling and shopping concerns, the smart Internet infrastructure suggests personal web services to buy air tickets, book a hotel and rent a car for this trip. As the woman browses the web, new personal context entities are added to her personal shopping web. Nevertheless, existing personal linked services such as her credit card payment service must be integrated to the current shopping activity. It is important to point out that shopping concerns differ for each event. Thus, a personal web infrastructure supported by the smart Internet is required to assist the user in her web experience taking into account her current situation.

In this motivating scenario the user acts as the main controller in the definition of her personal web. Supported by our context management infrastructure built as part of the smart Internet infrastructure, the user will be able to define new relationships with personal web-enabled context entities at run-time. This motivating scenario is used throughout the following sections to explain how, exploiting feedback loops and context awareness techniques, we can manage dynamic context to assist users in driving web integration as proposed by the Personal Web.

3 Dynamic Context Management for the Personal Web

Figure 1 illustrates the dynamic context management framework we propose to assist users in driving web integration in the Personal Web. Our approach is based on the feature-oriented reference framework we proposed for evaluating and implementing context management infrastructure, as required by the Smart Internet, and on the context management infrastructure we proposed for realizing dynamic monitoring in SOA governance [12, 11]. As we discussed in [12], the first big challenge related to context-awareness in the Smart Internet is the *identification of relevant context* for a specific set of interactions or user's matters of concern. The second challenge is then the *modeling* of these requirements in such a way that the representation of context is able to adapt itself dynamically as the application's environment and the user's concerns change. Once relevant context and context management objectives are modeled, the third challenge is to *design and implement* a dynamic context management infrastructure able to support the gathering, handling, and exploitation of context according to the model.

To propose this framework, we hypothesize that web integration can be addressed by integrating the user into a context management feedback loop. In this feedback loop, the user acts as part of the controller, specifically as the planner, to modify models of the real world that support the run-time instantiation of existing or new types of context entities within the user's Personal Web Sphere. To realize context modeling, we integrate the context taxonomy, the feature-based models and the context meta-model we proposed in previous works [11, 12] with the three layered meta-model envisioned as part of the basic infrastructure of the Personal Web [7]. To realize dynamic context management, our user-centric feedback loop supports the discovery and monitoring of relevant entities, and the adaptation of personal context models according to the user's interactions with these entities.

3.1 Modeling Personal Context Entities

Context modeling is an important component of the context information life cycle in the Personal Web. Personal context models represent the relevant aspects of the interactions between the user and entities of the real world, as well as the relationships among these entities that can affect the user's interactions. In our approach, entities of the real world that are relevant for the user's personal web sphere are known as *personal context entities*. To represent personal context entities, we define context models on three different levels of abstraction based on the three-layered meta-model proposed for the Personal Web [7]. These three layers correspond to the concept layer, the model layer and the instance layer. According to Fig. 1, *the concept layer* is composed of a foundational feature-based meta-model that defines the types of context entities and the relationships among them according to the general context taxonomy we proposed for the Smart Internet [12]. This meta-model is complemented with domain specific ontologies to instantiate the domain specific personal context models that define *the model layer*. Finally, the actual instances of the personal context models, which are derived from the interaction between the user and personal web enabled context entities (web application layer), constitute the *instance layer*.

3.1.1 The Concept Layer: The Foundational Context Representation

The feature-based meta-model for dynamic context management and the context taxonomy we proposed in [11, 12] provide the abstractions for the instantiation of personal context models. This feature-based meta-model defines the abstract context data types and the relationship among them. Relationships among context entities support the instantiation of both the relationships between the user

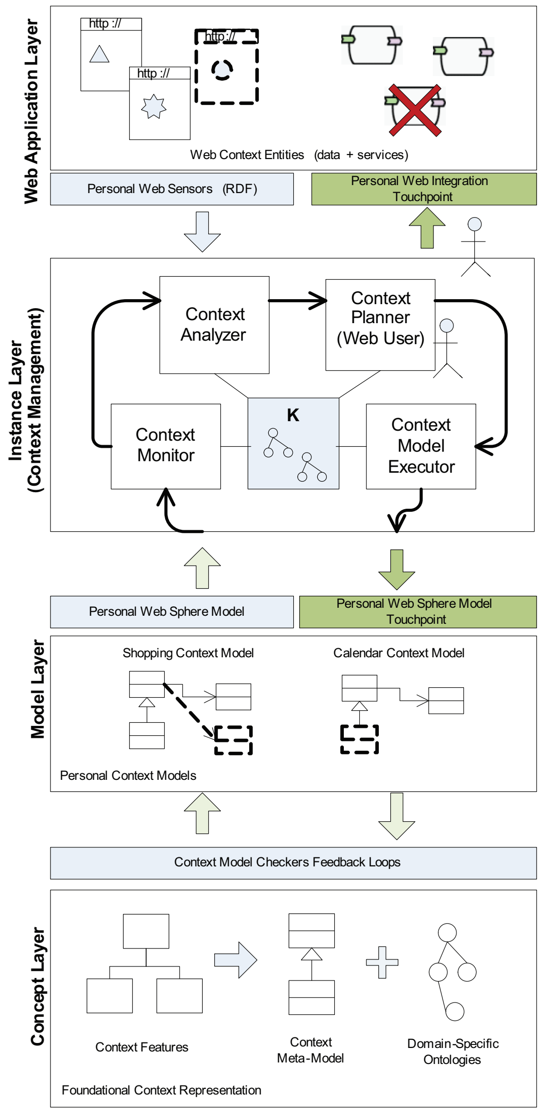

Figure 1: User-driven dynamic context monitoring framework for the Personal Web. The user acts as the planner of the feedback loop by modifying context models that drive the discovery and monitoring of personal data and personal web services according to the user's concerns. Personal context models are defined at three levels of abstraction as proposed by the Personal Web.

and entities of the real life, and the relationships among these entities. This meta-model also defines the abstractions that represent relevant properties of these entities. The abstract definition of the relationships between the user and context entities will enable the linking of personal data and personal services at the instance level. To realize the instantiation of personal web context models from this abstract representation, domain-specific ontologies complement the concept layer to express semantic dependencies between instances of context information in particular web domains [10, 3]. For our motivating scenario, the concept layer defines the abstract types required for the instantiation of the personal context models that represent the context entities relevant for the personal shopping web sphere and the personal calendar web sphere of the user. The meta-model together with domain-specific ontologies also defines the instantiation of valid interactions between the user and the relevant context entities. Instances of these interactions are a "debit" interaction with the "credit card" service entity, a "buy" interaction with a "dress" product entity through a "shopping car" service entity, and an "attend" interaction with an activity context entity such as the anniversary dinner or the business trip.

3.1.2 The Model Layer: Personal Context Models

From the abstract representation of context entities and the relation among them supported by web domain-specific ontologies in the concept layer, the model layer instantiates basic domain-specific personal context models required by the user's personal web sphere. These models are used to represent actual types of personal context entities that can be relevant for the user, the valid interactions between the user and these entities, and the relationships among these context entities required to support the user's interactions. Moreover, these concrete models define the information that characterizes the situation of entities the user interacts with. Personal context instances that define the user's personal web sphere in the instance layer are derived from personal context models of this layer. Revisiting our motivating scenario, a particular instance of personal context model for a shopping concern should include context entities such as "product", "on-line vendor", "financial institution", "credit card", "on-line shopping car", "calendar event"; user interactions such as "buy" a product, "attend" an event, "debit" from a credit card; and relationships among types of context entities such as "partnership" among different on-line vendors. Supported by our monitoring feedback loops, the user drives the adaptation of these personal context models according to her web integration concerns.

3.1.3 The Instance Layer: Personal Context Entities

As depicted in Fig. 1, the instance layer is composed of the context management feedback loop including the knowledge base. Personal context instances derived from the model layer are part of the knowledge base of the user's personal web sphere. Personal context instances represent entities of the real world and the way how the user interacts with them. They also model the relationships among these entities that are relevant for the user's interactions. Examples of context instances from our motivating scenario are the "dress" that the user is buying, the "fashion store" where she is placing the purchase order, her "financial institution", her "credit card", the "shopping car" where the dress, the shoes and the accessories are registered in, the "anniversary dinner" and the "business trip". Relationships and interactions are instantiated in the same way as proposed in the model layer but between concrete context entities and the user.

As our framework is based on the enabling infrastructure of the Personal Web [7], the identification personal web-enabled context instances, the relationships among them, and user's interactions are based on the Resource Description Framework (RDF) [1, 5]. Instances of actual implementations of RDF are the OpenLike protocol supported by Google and the Open Graph Protocol (OGP) proposed by Facebook. Both protocols focus on capturing relationships among users by defining "like" interactions with web entities [4, 13]. Context entities are represented as web object nodes in an RDF graph (e.g., data entities, services). In the same way, the edge between a pair of entity nodes can represent a semantic relationship between these two context entities. These relationships are driven by the user through personal interactions. New interactions link new data or new services to the user's personal web sphere. The semantics of these personal context instances are defined using RDF properties. Moreover, mechanisms based on RDF can be used by personal web sensors to discover relevant context entities according to the context definition of the user's personal web sphere.

3.2 User-Driven Monitoring Feedback Loops

By dynamically upgrading personal context models (cf. the model layer) that support the instantiation of context entities and user's interactions (cf. the instance layer), the user controls her personal web sphere to integrate new personal data and personal services. Whenever the user interacts with entities within a particular web domain, concrete context entities and interactions instantiate the personal context model to set up the user's personal web sphere accordingly.

Inspired by the autonomic element of autonomic computing and its application to the engineering of adaptive systems [2, 6], we propose a context management infrastructure where the user acts as the central part of the controller. By selecting interactions from the available set of interactions with entities of the real world, the user acts as the *planner* of the feedback loop. Interactions performed by the user are contextual events that triggers the modification of personal context models. These modifications are performed by the *context model executor* according to the user's interactions. When interactions imply the addition of new types of context entities to the personal model (cf. model layer), the *context model checkers* control the instantiation of the general meta-model (cf. concept layer). While the user is browsing the web, the *context monitor* uses the corresponding personal context model to discover, through "RDF sensors", potential personal web-enabled context entities according to her matters of concern (e.g., the anniversary dinner and the business trip). Then, the *analyzer* correlates information from context entities with properties of the corresponding personal context model to identify a set of relevant interactions between the user and these entities of the real world. The user can also act as part of the analyzer by selecting as relevant, interactions automatically identified as non-relevant.

4 Discussion and Future Work

Our ongoing research focuses on the validation of available technologies for the implementation of our context management framework. Empirical evaluation is also required to assess the impact of the proposed approach in supporting the vision of the Personal Web.

This paper proposes a dynamic context management approach to assist the user in the integration of personal data and personal services into her personal web sphere. This way, we contribute to the realization of a *dynamic Personal Web Link* by modifying, at run-time, the models that drive the discovery of personal web-enabled entities (personal context entities) and guide the interactions between the user and these context entities. With respect to modeling aspects of personal web domains, modeling artifacts are implemented at three different levels of abstraction as proposed for the Personal Web [7]. Feedback loops play an important role to guarantee consistency among these layers of abstractions. Between the two lowest layers, the model and the concept layer, context model checkers (implemented as feedback loops) monitor transformations in personal context models to check the properties of the resulting models with respect to the conceptual properties represented by general context-meta models and the corresponding domain-specific ontologies. Between the instance layer and the model layer, the relation is twofold. First, a top-down interaction enables the context model executor from the instance layer to trigger transformations of the personal context models defined by the model layer. Second, changes in personal context models modify the set of potential personal web-enabled entities to be monitored by the context monitor. Whenever a personal context model transformation is performed, context model checkers perform the validation against conceptual models defined by the bottom layer. It is important to point out that not all the interactions performed by the user will trigger transformations in personal context models. The user could integrate new personal data or services that are instances of existing types in the actual personal context models.

With respect to architectural concerns, every element of the framework should be designed independently to support distribution. Enabling infrastructure of the Personal Web is required to be deployed either at the server domain, at the client domain or distributed on different processing nodes. For supporting architectural decisions, control-based reference models and reference architectures provide valuable guidelines for the implementation of dynamic monitoring infrastructure [11].

Finally, more research on self-adaptive service-oriented systems is required to leverage the user experience in the Personal Web. For instance, dynamic sensing infrastructure for discovering smart services based on dynamic service selection must be integrated to provide the user with a richer set of resources from the Smart Internet, beyond personal web-enabled context entities discovered from web pages [11, 9].

Acknowledgements

This work grew out of collaboration with colleagues at IBM Toronto Center for Advanced Studies, CA Labs, and Carnegie Mellon Software Engineering Institute. We are deeply indebted to many of our friends and students, who contributed significantly to our appreciation for the personal web and understanding of context-awareness. In particular, we would like to thank Gabriel Tamura, Grace Lewis, Sowmya Balasubramanian, Ron Desmarais, Priyanka Gupta, Ishita Jain, Stephan Jou, Przemek Lach, Atousa Pahlevan, Dennis Smith, Qian Yang, and Qin Zhu for their continued collaboration and support. This work was funded in part by the National Sciences and Engineering Research Council (NSERC) of Canada (CRDPJ 320529-04 and CRDPJ 356154-07), IBM Corporation via the CSER Consortium, as well as Icesi University, Cali, Colombia and University of Victoria, British Columbia, Canada.

About the Authors

Norha M. Villegas is a CAS PhD student under the supervision of Prof. Hausi A. Müller. Together with her research groups at University of Victoria (Rigi Research Group - Canada) and Icesi University (DRISO Research Group - Colombia), she is working on the application of software engineering models, techniques and architectures to the development of self-adaptive and self-managing systems. In particular, she concentrates on the implementation of autonomic infrastructures able to support the dynamic adaptation and evolution of context aware applications. Research interests include feedback loops, autonomic computing, dynamic context management, context-awareness, Smart SOA, SOA governance and Smart Internet.

Hausi A. Müller is Associate Dean Research, Faculty of Engineering and Professor of Computer Science, University of Victoria, British Columbia, Canada. He was founding Director of the Bachelor of Software Engineering program. In collaboration with IBM, CA and SEI, his research group investigates methods, models, architectures, and techniques for self-managing and self-adaptive software-intensive systems. Dr. Müller's research interests include software engineering, software evolution, autonomic computing, diagnostics, SOA governance, software architecture, software reverse engineering, re-engineering, program comprehension, visualization, and software engineering tool evaluation. He is Program Co-Chair for IBM CASCON 2010. He was General Chair of VISSOFT 2009 and co-organizer of SEAMS 2006-10 and DEAS 2005. He was Workshops Co-Chair for ICSE 2008. He was General Chair for IWPC 2003 and ICSE 2001. Dr. Müller served twice on the Editorial Board of IEEE Transactions on Software Engineering

(TSE). He is Vice-Chair of IEEE Computer Society Technical Council on Software Engineering (TCSE).

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.

Alex Lau is a research staff member at IBM CAS Canada Research, where his research focuses on service oriented architecture, BPM tools, and business events. Prior to joining CAS, Alex was a software engineer in IBM Canada, where he worked on a number of Websphere BPM products including Websphere Integration Developer and Websphere Business Modeler. Alex holds a Bachelor's degree in Mathematics (Computer Science) and a Master's degree in Mathematics (Computer Science) from the University of Waterloo.

References

[1] T. Berners-Lee, W. Hall, J. A. Hendler, K. O'Hara, N. Shadbolt, and D. J. Weitzner. A framework for web science. *Foundations and Trends in Web Science*, 1(1):1–130, 2006.

[2] J. O. Kephart and D. M. Chess. The vision of autonomic computing. *Computer*, 36(1):41–50, 2003.

[3] R. Krummenacher and T. Strang. Ontology-based context modeling. In *Proceedings Third Workshop on Context-Aware Proactive Systems (CAPS 2007)*, June 2007.

[4] LikeProtocol Working Group. The OpenLike Protocol. http://openlike.org/, 2010.

[5] F. Manola, E. Miller, and B. McBride. RDF Premier. W3C Recommendation. http://www.w3.org/tr/rdf-primer/, 2004.

[6] H. A. Müller, H. M. Kienle, and U. Stege. Autonomic computing: Now you see it, now you don't—design and evolution of autonomic software systems. In A. De Lucia and F. Ferrucci, editors, *International Summer School on Software Engineering (ISSE) 2006-2008*, volume 5413 of *LNCS*, pages 32–54. Springer, 2009.

[7] J. Ng. The Personal Web: Smart Internet for Me. Technical report, IBM Center for Advanced Studies Lab, Toronto, Ontario, Canada, 2010.

[8] J. Ng, M. H. Chignell, and J. R. Cordy. The smart Internet: Transforming the web for the user. Technical report, IBM Canada Center for Advanced Studies, Technical Report, 2009.

[9] A. Pahlevan and H. A. Müller. Static-discovery dynamic-selection (sdds) approach to web service discovery. In *Proceedings of the 2009 World Conference on Services)*, pages 769–772. IEEE Computer Society, 2009.

[10] T. Strang, C. Linnhoff-Popien, and K. Frank. CoOL: a context ontology language to enable contextual interoperability. In *Proceedings 4th IFIP WG6.1 International Conference on Distributed Applications and Interoperable Systems (DAIS 2003)*, volume 2893 of *LNCS*, pages 236–247. Springer, Nov. 2003.

[11] N. M. Villegas and H. A. Müller. Context-driven adaptive monitoring for supporting soa governance. In *Proceedings of the 4th International Workshop on a Research Agenda for Maintenance and Evolution of Service-Oriented Systems (MESOA 2010)*. Carnegie Mellon University Software Engineering Institute, 2010. To appear.

[12] N. M. Villegas and H. A. Müller. Managing dynamic context to optimize smart

interactions and services. In *Proceedings of the Smart Internet Technologies Workshop 2009 (CASCON conference). Springer LNCS*, volume 6400, 2010. To appear.

[13] M. Zuckerberg and B. Taylor. The Open Graph Protocol. http://www.opengraphprotocol.org, 2010.