Constraint-Sensitive Privacy Management for Personalized Web-Based Systems[1]

Yang Wang

Donald Bren School of Information and Computer Sciences,
University of California, Irvine, U.S.A
yangwang@ics.uci.edu

Abstract. This research aims at reconciling web personalization with privacy constraints imposed by legal restrictions and by users' privacy preferences. We propose a software product line architecture approach, where our privacy-enabling user modeling architecture can dynamically select personalization methods that satisfy current privacy constraints to provide personalization services. A feasibility study is being carried out with the support of an existing user modeling server and a software architecture based development environment.

1 Introduction

The benefits of web-based personalization for both online customers and vendors have been challenged and counteracted by privacy concerns [1]. When privacy laws and regulations are in effect, they restrict not only the personal data that can be collected and manipulated by the personalized websites, but also the *methods* [2] that can be used to process the data. For instance, the German Teleservices Data Protection Act [3] that mandates personal data to be erased immediately after each session except for very limited purposes would preclude the employment of certain machine learning methods where the learning takes place over several sessions. On the other hand, though, alternate personalization methods can often provide the same or similar personalization services with possibly fewer privacy impacts but possibly also lesser quality [4]. In our example, a personalized website could use incremental machine learning (that discards all raw data after the end of a session) to provide personalization to web visitors from Germany[2], while it can use possibly better one-time machine learning with the data stored across several sessions to provide personalization to web visitors from the U.S. who are not subject to this constraint.

From a personalization point of view, we ask the research question: how can personalized web-based systems maximize the personalization benefits, while

[1] This research has been supported through NSF grant IIS 0308277. I would like to thank Alfred Kobsa, André van der Hoek and Eric Dashofy for their help in preparing this paper.

[2] This is not yet a complete solution though since the German Teleservices Data Protection Act also mandates that profiling requires the use of pseudonymous or the consent of the user.

L. Ardissono, P. Brna, and A. Mitrovic (Eds.): UM 2005, LNAI 3538, pp. 542 – 544, 2005.

respecting the privacy constraints that are currently applicable (such as privacy laws and regulations, and the user's privacy preferences)? [3]

2 Proposed Approach

Because of the high cost of personalization [5], we suggest to address this issue early in the design of a User Modeling Server (UMS) [6]. We propose a software architecture that can dynamically select methods to provide personalization services. To incarnate this idea, we choose the Software Product Line (SPL) approach from software architecture research. SPLs have been successfully introduced in industrial software development for improving productivity, software quality and time-to-market [7]. Product-line software development exploits commonalities between related products via a shared repository of carefully selected software artifacts, from which a particular product can be generated using built-in variability mechanisms [8]. The idea to treat software as a product line brings a new way of supporting "any-time" software variability (i.e. at design, invocation and run time) [9].

We conceive our UMS as an extensible SPL architecture where each of the different personalization methods is embedded in an individual component and new methods (components) can be easily plugged into the architecture. The software architecture can dynamically filter all components that violate the current privacy constraints and then, optionally, elect one or more of the remaining components to provide the personalization service based on their anticipated quality of service. Thereafter the SPL can instantiate a separate run-time system instance of the remaining or the selected components to serve the current user. In order to prevent the situation that too many run-time system instances degrade the overall performance of the UMS, the system can merge instances that have the same system configurations.

3 Feasibility Study

We are conducting a feasibility study which utilizes an existing LDAP-based UMS [10] and ArchStudio 3.0, a software architecture based development environment. The UMS is expressed in xADL 2.0 [11], the underlying XML-based architectural description language for ArchStudio 3.0 that supports architecture-level configuration management (such as versioning, diff and merge operations). Different personalization methods are treated as variant components guarded by Boolean expressions with privacy constraints in the architecture. If the Boolean expression of a variant component can be fully resolved to be TRUE or FALSE, the component is included or excluded in the architecture, and a new run-time system is instantiated for the current user that is consistent with the currently prevailing privacy constraints. If such a runtime system already exists, the user session will be assigned to this session instead.

We are currently developing a prototype system that we intend to evaluate against privacy laws from several countries and privacy attitudes that were solicited

[3] While there is no unanimous measure of personalization benefits, we utilize the anticipated quality of personalization methods being used as a quantitative indicator of these benefits.

from Internet users. This will help us verify whether these privacy constraints can indeed be expressed in our system and whether it is able to cater to the users in the expected manner. It will also give us an indication of the performance and scalability of our approach.

4 Conclusions

Enabling personalized websites to operate in a privacy-aware manner (both with respect to legal and user requirements) will allow users to utilize personalization services with less privacy concern. Our approach allows personalized websites to address the combinatorial complexity of privacy constraints in a systematic and flexible manner, building on state-of-the-art industry practice for managing software variants at runtime. We aim at exploring the feasibility of this approach using an existing user modeling server and empirically established privacy constraints.

References

1. Teltzrow, M. and A. Kobsa, *Impacts of User Privacy Preferences on Personalized Systems: a Comparative Study*, in C.M. Karat et al, ed.: *Designing Personalized User Experiences for eCommerce*, 2004, Dordrecht, Netherlands: Kluwer
2. Kobsa, A., J. Koenemann and W. Pohl, *Personalized Hypermedia Presentation Techniques for Improving Online Customer Relationships.* The Knowledge Engineering Review, 2001. **16**(2): p. 111-155
3. DE-TS, *German Teleservices Data Protection Act*. 1997.
4. Kobsa, A. *A Component Architecture for Dynamically Managing Privacy in Personalized Web-based Systems*. in R. Dingledine, ed.: *Privacy Enhancing Technologies: Third Intern'l Workshop*. 2003. Dresden, Germany: Springer. 177-188
5. Jupiter, *Beyond the Personalization Myth.* 2003. http://www.jupiterresearch.com/bin/item.pl/research:vision/79/id=94553,keywords1=personalization/
6. Kobsa, A., *Generic User Modeling Systems.* User Modeling and User-Adapted Interaction, 2001. **11**(1-2): p. 49-63.
7. Bosch, J., *Design and Use of Software Architectures: Adopting and Evolving a Product-Line Approach.* 2000, New York: Addison-Wesley
8. Paul Clements, Linda M. Northrop, *Software Product Lines: Practices and Patterns*. 2002, New York, New York: Addison-Wesley
9. Hoek, A.v.d., *Design-Time Product Line Architectures for Any-Time Variability.* Science of Computer Programming, **53**(30): p. 285-304
10. Fink, J., *User Modeling Servers: Requirements, Design, and Evaluation.* 2004, IOS Press, Netherlands (Infix)
11. Dashofy, E. M., A. van der Hoek, and R. N. Taylor. *A Highly-Extensible, XML-Based Architecture Description Language.* In *Proceedings of the Working IEEE/IFIP Conf. on Software Architecture*. 2001. Amsterdam, Netherlands.103-112