Using User Context for Accessing IT Resources

Lars Krüger
VLBA Lab - ITI
Otto-von-Guericke-University
P.O. Box 4120, 39016 Magdeburg, Germany
+49 391 67 54832
lars.krueger@ovgu.de

Bastian Grabski
VLBA Lab - ITI
Otto-von-Guericke-University
P.O. Box 4120, 39016 Magdeburg, Germany
+49 391 67 54832
bastian.grabski@ovgu.de

ABSTRACT
Individualization is an enhancement of existing role concepts by subjective information demand. Role concepts, which belong to *personalization*, grant access to IT resources. This paper prepares the ground for a context-based approach that provides *individual* – as opposed to *personalized* – access to IT resources in heterogeneous system landscapes. A central part of such an approach is the definition of user context. Here, we provide such a definition, derived from the state of the art in this field, along with a UML class model. Our definition of user context is validated by relating the UML context model to the authorization concepts of SAP R/3, AIX and Solaris.

Categories and Subject Descriptors
H.1.2 User/Machine Systems: *Human factors*; D.2.11 Software Architectures: *Domain-specific architectures*; K.6.5 Security and Protection (D.4.6, K.4.2): *Authentication*

General Terms:
Management, Design, Security, Human Factors, Languages.

Keywords
Individualization, Personalization, Authorization, Human Factors, Context-awareness, Domain-specific language.

1. INTRODUCTION
The idea of individualization and opportunities for its realization are increasingly discussed. Being originally a topic of social sciences [5, 9], individualization now catches on in business informatics [8, 16, 24, 14] and in interdisciplinary projects dealing with cognitive technical systems [22]. Individualization goes beyond traditional personalization (such as manual access grants in particular applications or role orientation) as it does include the user as an individual with his or her specific requirements.

The need for individualization appears in different forms and is illustrated by the following use case: A user has to access specific IT resources in order to accomplish a task in a business process, but the required privileges are not assigned to the role the user takes on. In general, a *role* is either a formal (*business role*) or an informal (*functional role*) organizational item that represents the user's tasks or a formal technical item (*technical role*) that correspond to a user's access privileges. A *business role* represents only organizational responsibility, not a person's identity [20]; thus, it cannot be used for individual access to IT resources. *Business roles* include a person's tasks (process roles in process organization) and position (combination of organizational unit and permanent post) [25]. Although many business processes are technically supported, *application and provisioning of privileges* must be done manually, which makes it difficult to have roles changed in case of, e.g., vacation or illness. Additionally, the manual processes of setting up accounts, granting access privileges and assigning roles increases the effort to administrate system landscapes [19]. Despite automatic provisioning of IT resources (for example with GRC techniques, but only for SAP systems [19]), there is no end-to-end process for accessing heterogeneous IT resources that is user-driven and technically supported. As a result of vendor-specifics and high security requirements in heterogeneous system landscapes, each IT system includes its own user administration and security functions (e.g., for the assignment of access privileges) [10]. Hence, the complexity of authorization management increases in line with the size of a system landscape [18, 11, 21]. On the other side, there exist *management strategies* that don't apply sanctions against deviant behaviour, but even support creative employees. These management strategies make use of the flexibility and creativity of specialists and managers, who work in a responsible and independent way and are often in charge of their own budget, to increase economic value [5]. However, to apply these strategies, individualization of software is crucial. We achieve individualization by involving a user's context.

In this article, the foundation of an approach is introduced that allows a user to have an individual (possibly automatic) access to IT resources in a heterogeneous systems landscape. Role concepts provide the foundation for modeling a user's information demand. Augmented by individual demand, the resulting extended role concepts are used as special context information. In general, *context* includes information that can be used to characterize the situation of an interaction partner [4]. Here, the information of the interaction partner *user* is depicted in a *user context model. User context* is used to represent a user's information demand by using role concepts from the business and the individual perspective. If this information was described by a domain specific language (DSL), whose (abstract) syntax is based on the user context model, our approach could be used for granting access privileges automatically. However, the DSL is not discussed here. The challenges of the presented research consist in reducing the complex-

CASTA'09, August 24, 2009, Amsterdam, The Netherlands.

ity of administrating authorizations and in synchronizing the organizational information demand (business and functional roles) with technical authorization concepts (technical roles). This can be achieved by an appropriate modeling of user context, which is suggested in the next Section.

2. STATE OF THE ART

Personalization aims at granting users (which are explicitly modelled [2]) access to IT resources such as data and functions of IT systems via formal business and technical roles that are both compliant. *Individualization* includes not only formal, but also informal role orientation, namely rights, duties and adaptation of information selection and presentation to requirements, preferences and knowledge of a user [14]. Additionally, here, the informal role orientation aims at the compliance of functional and technical roles. Usually, individualization requires a continuous process, in which feedback of a user leads to adjustments according to a given situation [1]. In practice, the predominant method for assigning privileges in complex application systems is the use of roles. IT systems include authorization concepts, such as access privileges and technical roles, as a set of rules that determine a user's access to functions and data. *Access privileges* represent interrelations between subjects, objects and access activities; during authorization, they are verified and confirmed. Technical roles in heterogeneous system landscapes are object of interest in the field of identity management, e.g., as 'Identity as a Service' [15, 3, 17], but the scientific world has previously paid little attention to this issue (e.g., in [24, 26]). In a particular *individual situation,*

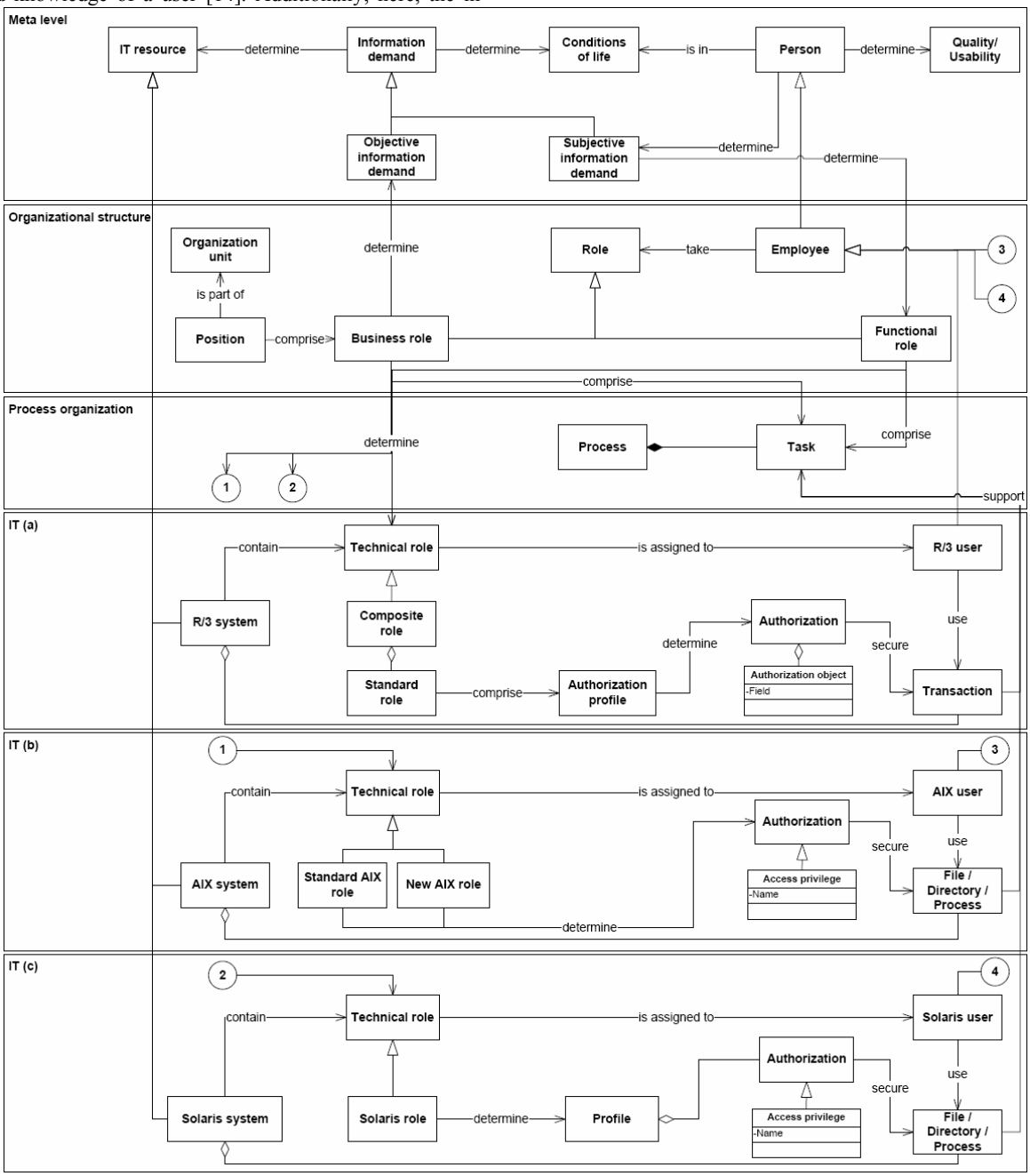

Figure 1. User Context Model

the information demand may change so that a user needs access privileges that are not granted so far [27]. Kuhlmann et al. [12] call this an assignment of *functional roles* that extend business roles by additional user-specific functions or tasks. In [16], this is called *extra role behavior*, where the user takes a new role in a creative or even contradictory way. The *situation-based* approach is an appropriate way to describe a person's specific circumstances or conditions of life [15, 16, 13]. Therefore, context models are the most frequently discussed approach as their design is sensitive to situational changes [4, 27]. Hence, situation orientation is context awareness.

3. MODELING USER CONTEXT

3.1 User Context

According to recommendations for context description [27], a user's information demand should be represented using (1) a general *definition* in natural language, (2) a formal representation as *user context model,* and (3) an operational utilization of this model by designing an appropriate (machine-readable) language. This article deals with aspect (1) and (2), while (3) is subject of ongoing work. Based on Section 2, the terms related to a user's context are pulled together here for the first time (see Fig. 1, 'Meta level') to realize individualization on IT level. User context uses organizational formal or informal role concepts (see Fig.1, level 'Organizational structure'), which in turn include job or task descriptions (see Fig. 1, level 'Process organization').

Defining *User Context* (see Fig. 1)**:** The user's information demand, i.e. his (or her) need for IT resources, arises out of his task and can be classified into objective and subjective information demand. Source for *objective information demand* is the task description according to his job characteristics described in a *business role*. The set of IT resources a user needs for the accomplishment of his task (*process role*) that is carried out in his position (*business role*) is called *technical role* or *IT role*. Subjective information demand contains requests for IT resources that extend the user's job description and thus lead to a *functional role*. The discrepancy between objective and subjective information demand is called a special situation within the user life cycle (i.e., a condition of life). All things considered, *user context* represents objectively as well as subjectively required IT resources for *all* tasks that arise from a user's specific *situation* and is expressed by business and functional roles. These roles have to be synchronized with the set of all needed IT resources that are represented by IT roles.

3.2 User Context and IT Roles

To enable context-aware and automatic granting of access privileges, *user context* must be mapped to *technical roles (IT roles)*. To basically validate the definition provided in Section 3.1., authorization concepts of SAP R/3[1] and UNIX (AIX, SOLARIS) were investigated. Later we will investigate authorization concepts of other systems as well in order to make the user context model as generic as possible. The model depicted in Fig. 1 contains the authorization concept of SAP R/3 [10, 6], AIX [7] and Solaris [7] in the different views of 'IT' and is shortly explained in the following.

Functions and data of a SAP system, which are accessed by *transactions*, are secured by means of *authorization objects*. These objects consist of fields whose values describe particular *access privileges* in detail. The combination of different authorizations is called *authorization profile*. An authorization profile is assigned to only one *standard role*. *Composite roles* are collections of standard roles and are assigned to SAP R/3 users – according to their business roles.

In general, authorizations in UNIX, which are actually *access privileges*, are described by names and support a user's task. Solaris resources are also accessed via profiles that determine authorizations. Here, profiles are used as a collection of access privileges as opposed to SAP systems, where (automatically generated) profiles are used for authorization checks and are a prerequisite for assigning technical roles to a user. In AIX systems, authorizations are directly assigned to roles, which in turn assigned to an AIX user.

There are several methods for context modeling [23]. In this article, UML class diagrams have been used. Fig. 1 depicts the resulting *user context model* and includes all terms discussed above. The figure clarifies the synchronization of different role types (business and functional as well as technical roles). By showing the compliance between organizational and technical roles it is validated that individualization on IT level can be achieved by applying our definition of 'user context'.

4. CONCLUSION

The paper has discussed terms and approaches to synchronize distinct roles, which are defined from the perspective of organizations, individuals and IT systems. To practically achieve such a synchronisation, *user context* was defined and modelled by UML class diagrams. The design of an *authorization model* extends the user context model. Context awareness includes the adaptation of the user context model to changing situations or information demand, respectively. In this way, the approach allows an individual provision of IT.

The automatic provisioning of IT resources is part of ongoing work and was delineated with the construction of a DSL. In fact, there already exist opportunities for accessing heterogeneous systems automatically on the basis of existing roles, e.g., in the area of identity management. These approaches are not situation-aware: Roles are created once and then remain static.

Goal of our future work is the design of a comprehensive authorization model, which represents the diversity of authorization concepts, and the integration of this model in the *user context model*. Both authorization model and user context model shall provide information for DSL construction in order to operationalize user context awareness within a systems landscape.

5. REFERENCES

[1] Adomavicius, G., Tuzhilin, A. 2005. Personalization Technologies: A Process-Oriented Perspective. Communications of the ACM (CACM) 48, 10 (October 2005), 83-90. DOI= http://doi.acm.org/10.1145/1089107.1089109.

[1] Authorization concepts of many software manufacturers base on the authorization concept of SAP R/3. This, considering SAP R/3 in Figure 1 ensures a broad application range.

[2] Bodendorf, F. 1992. Benutzermodelle – ein konzeptioneller Überblick. WIRTSCHAFTSINFORMATIK 34, 2 (April 1992), 233-245.

[3] Fink, U. 2009. SAP ersetzt Benutzerverwaltung. In Computer Zeitung 13 (March 2009). 19.

[4] Dey, A. 2001. Understanding and Using Context. http://www.cc.gatech.edu/fce/ctk/pubs/PeTe5-1.pdf.

[5] Deeg, J., Weibler, J. 2008. Die Integration von Individuum und Organisation. VS Verlag.

[6] Esch, M., Junold, A. 2008. Berechtigungen in SAP® ERP HCM. Galileo Press.

[7] Frisch, A. 2002. Essential System Administration: Tools and Techniques for Linux and Unix Administration. 3. ed., O'Reilly.

[8] Fischbach, K., Schoder, D., Gloor, P. 2009. Analysis of Informal Communication Networks. Business & Information Systems Engineering 1, 2 (April 2009), 140-149.

[9] Green, S. M. 2004. Individualisierung und Wissensarbeit. Individualisierungsprozesse im Unternehmen und ihre Auswirkungen am Beispiel der Personalorganisation. DUV.

[10] IBM Business Consulting Services 2003. SAP® Berechtigungswesen. Design und Realisierung von Berechtigungskonzepten für SAP R/3 und SAP Enterprise Portal. Galileo Press.

[11] Kern, A., Kuhlmann, M., Schaad, A., Moffett, J. 2002. Observations on the Role Life-Cycle in the Context of Enterprise Security Management. In Proceedings of the 7th ACM symposium on Access Control Models and Technologies (Monterey, California, USA, June 3-4, 2002). SACMAT'02. ACM Press, New York, NY, 43-51. DOI= http://doi.acm.org/10.1145/507711.507718.

[12] Kuhlmann, M., Shohat, D., Schimpf, G. 2003. Role Mining – Revealing Business Roles for Security Administration using Data Mining Technology. In Proceedings of the 8th ACM Symposium on Access Control Models and Technologies (Como, Italy, June 2-3, 2003). SACMAT'03. ACM Press, New York, NY, 179-186. DOI= http://doi.acm.org/10.1145/775412.775435.

[13] Levashova, T., Lundquvist, M., Pashkin, M. 2006. Moving Towards Automatic Generation of Information Demand Contexts: An Approach Based on Enterprise Models and Ontology Slicing. In OTM 2006, LNCS 4275, R. Meersmann, Z. Tari, Eds. Springer, Berlin, 1012-1019.

[14] Mertens, P., Stößlein, M., Zeller, T. 2004. Personalisierung und Benutzermodellierung in der betrieblichen Informationsverarbeitung – Stand und Entwicklungsmöglichkeiten. Technical Report No. 2/2004, Universität Erlangen-Nürnberg.

[15] Müller-Corbach, K. 2008. Unternehmensweites Rollenmanagement. http://www.doag.org/pub/docs/Publikationen/DOAGNews/2008/2008-4/2008-04-News-Mueller-Corbach-Rollen_Management.pdf.

[16] Meier, M. C., Winkler, V., Buhl, H. U. 2007. Ansätze zur Gestaltung situierter und individualisierter Anwendungssysteme. WIRTSCHAFTSINFORMATIK 49, Sonderheft (Februar 2007), 39-49.

[17] Reckeweg, A. 2008. Identitäten in einer Service orientierten Welt – Identity as a Service. http://www.sun.com/bigadmin/hubs/multilingual/deutsch/content/id_as_service.jsp.

[18] Rupprecht, J. 2002. Datensicherheit im Data Warehousing. Technical Report, Institut für Wirtschaftsinformatik, Universität St. Gallen.

[19] SecurIntegration 2008. GRC in SAP-Umgebungen. mitp.

[20] Samarati, P., De Capitani de Vimercati, S. 2001. Access Control: Policies, Models, and Mechanisms. In FOSAD 2000, LNCS 2171, R. Focardi, R. Gorrieri, Eds. Springer, Berlin, 137–196.

[21] Siedersleben, J. 2007. SOA revisited: Komponentenorientierung bei Systemlandschaften. WIRTSCHAFTSINFORMATIK 49, Sonderheft (Februar 2007), 110-117.

[22] SFB/Transregio 62 2009. Eine Companion-Technologie für kognitive technische Systeme. http://www.informatik.uni-ulm.de/ki/sfb-trr-62/.

[23] Strang, T., Linnhoff-Popien, C. 2004. A Context Modeling Survey. In Workshop on Advanced Context Modelling, Reasoning and Management (Nottingham, England, September 10-11, 2004). UbiComp 2004 – The Sixth International Conference on Ubiquitous Computing. http://elib.dlr.de/7444/01/Ubicomp2004ContextWSCameraReadyVersion.pdf.

[24] Walther, I. 2005. Rollen- und Situationsmodellierung bei betrieblichen Dispositions- und Planungssystemen. Doctoral Thesis, Universität Erlangen-Nürnberg.

[25] Wedde, H. F., Lischka, M. 2004. Modular Authorization and Administration. In: ACM Transactions on Information and System Security 7, 3 (August 2004), 363-391. DOI= http://doi.acm.org/10.1145/1015040.1015042.

[26] Wortmann, F. 2006. Entwicklung einer Methode für die unternehmensweite Autorisierung. Doctoral Thesis, Universität St. Gallen.

[27] Zimmermann, A., Lorenz, A., Oppermann, R. 2007. An Operational Definition of Context. In Modeling and Using Context, B. Kokinov, D. Richardson, T. Roth-Berghofer, L. Vieu, Eds. Springer, Berlin.