Managing University Identity Management Systems: A Design Science Approach

Gunnar Dietz

Projekt eCampus
Universität Hamburg
Schlüterstr. 70
20146 Hamburg
gunnar.dietz@uni-hamburg.de

Martin Juhrisch

Projekt MIRO
Universität Münster
Röntgenstr. 9 – 13
48149 Münster
juhrisch@uni-muenster.de

Abstract: Especially in the university context the configuration of identity management systems, the management of roles and their assignment to users is a highly complex task. To simplify this we propose an automated linking of the identity management system to business process models which contain organizational responsibilities. Based on the E^{3}plus approach [JW08] we introduce a modelling grammar and a method to access electronically stored enterprise models by Web services. The usefulness of our approach was tested within a pilot study conducted at the University of Münster and the universities of Hamburg (Germany).

1 Introduction

Within a modernization process many German universities currently implement identity management systems. These projects often are embedded in larger information management projects and come with efforts to realize a service oriented architecture where web services play a major role. The project MIRO[1] of the University of Münster (Germany) and the joint project eCampus of the universities of Hamburg (Germany) fit into this description.

While identity management software (the IBM Tivoli Identity Manager (ITIM) is used in Münster and the Novell Identity Manager is used in Hamburg) offer solutions for managing and consolidating identity data, the assignment of roles and entitlements and the provisioning of target systems with account data the main questions remain: How to implement the identity management system functionality to cope with the requirements of the departments and faculties, the central institutes, the administration of the universities and of course the users? How to support and automatize the many workflows within a university and how to close the gap between user data management and rights management?

[1] **M**ünster **I**nformation System for **R**esearch and **O**rganization

Central part for answering the last question is a well-configured role management together with policies that are able to equip a user with all rights needed for fulfilling his duties. The RBAC[2] approach (existing in many shapes) offers a theoretical solution supported by most identity management systems. Again, however, the following questions remain: How to determine a useful set of roles for this and how to implement the policies and how to assign them correctly to the users? To answer this questions a detailed process analysis is needed which must result in a design concept for the (configuration of the) IDMS[3]. This is normally an enormous effort especially since the number of roles in a university context is extremely high. Even the analysis of the organizational structure is often much more difficult than it should and responsibilities are often unclear. Often some kind of models exist but with limited availability and significance. It is difficult enough to consolidate this information and to generate design models from it, but it is much more difficult to cope with changes of requirements.

Therefore, an approach to facilitate the configuration of an IDMS based on modelled requirements is desirable. In this paper we present such a model driven approach. In this approach an automatic binding between electronically stored enterprise models and the IDMS software is suggested. Enterprise models can be enriched by metadata (especially role data) which can be accessed via Web services to configure the IDMS. In this way changes in the enterprise models can be monitored and result automatically in a change of the configuration of the IDMS. As a result the configuration of an IDMS will become much more comfortable and more transparent – which emphasises one major role of IDM: to increase security and privacy by transparency of data flow and the assignment of rights.

Our approach embeds into the more general E^{3}plus approach [JW08] which introduces a framework of patterns (object patterns (OP), attribute patterns (AP) and attribute value patterns (AVP)) and a way to integrate meta-information into enterprise models based on these patterns. Furthermore it uses the E^3+WS method [WJE06] to describe the Web services used to access the enterprise models.

The paper is structured as follows. Section 2 introduces a framework showing how enterprise and web-service models can control university's systems functionalities. Section 3 exposes the concept of the implemented approach. The discussion on Section 4 summarizes consequences, recapitulates the proposed ideas and exposes open questions regarding the realization of the application integration.

[2] RBAC = **R**ole **B**ased **A**ccess **C**ontrol
[3] IDM = **Id**entity **M**anagement; IDMS = **Id**entity **M**anagement **S**ystem

2 Integration Framework

Our approach focuses on a direct linking between addressable software components and business context[4] information (e.g. role information) represented in enterprise models. As no existing modelling language supports integration and sharing of model information with IT applications we achieve the necessary transparency through the extension of conceptual modelling languages, e.g. ARIS [Sc01] with language constructs that aim at the description of model information as web-services.

The property of web-services is to be easily specifiable with formal design languages. This leads to an evaluation of mapping possibilities from formal concepts of web-service design to the meta-level of conceptual modelling. We propose a semantically enriched meta-model for conceptual modelling. The genericity of these web-services enables a flexible integration between infrastructure and enterprise models. This requires additionally a model of web-services, describing the services which are used by the applications.

The E^3plus approach, as introduced in [JW08] offers the ability to store some kind of meta-data in enterprise models by adding pattern-constructs (namely object patterns (OP), attribute patterns (AP) and attribute value patterns (AVP)) to existing modelling languages as ARIS. This is done in several steps using the existing E^3-model [Gr04]: adding patterns as language constructs on meta-meta-model level, adding concrete pattern on meta-model level and using instances of these patterns on the normal modelling level. In contrast to the original idea – using patterns to describe business objects with natural language constructs – we focus here on AVPs (and their instances on model-level) to describe concrete values of certain meta-data needed in enterprise models.

Furthermore, the E^3+WS method [WJE06] – also based on E^3 – can be used to model Web services to access electronically stored enterprise models, see also [JWD07]. Combining this method with the E^3+ approach we can model (and build) Web services which are able to access meta-data stored in pattern instances used in models.

3 Implementing the approach

In the concrete identity management scenario we have done the following: Using E^3plus we extended ARIS and created – among others – a "role" OP together with corresponding APs ("name", "description", …). With AVPs we can either create concrete roles ("student", "employee", …) on meta-model level or allow the possibility to create arbitrary roles (as AVP instances) on model level – or a combination of both. Furthermore we created Web services, using E^3+WS, to access this role data. Therefore we can create EPC diagrams which can hold role information for each process, which is accessible via a Web service.

[4] Information passed between business process activities in an enterprise model together with the information provided by the modeller of the business process is called the context of a process flow (Leymann, Roller and Schmidt, 2002).

To use this data, the IDMS must access the Web service, read the role information for each process, translate this information to account data and propagate this information to the application or user management that is responsible for the process in question. A proof of concept exists for the ITIM (a modified workflow for account provisioning that accesses the model-data Web service) and a first implementation was done for the Novell Identity Manager (in this case a driver that accesses the Web service to synchronize the role information with the identity vault and another driver to translate this information into entitlements to the services in question, see Figure 2).

Now, if role information in an EPC diagram changes (see Figure 1), the IDMS may react immediately to this changes and change the rights for the services in question.

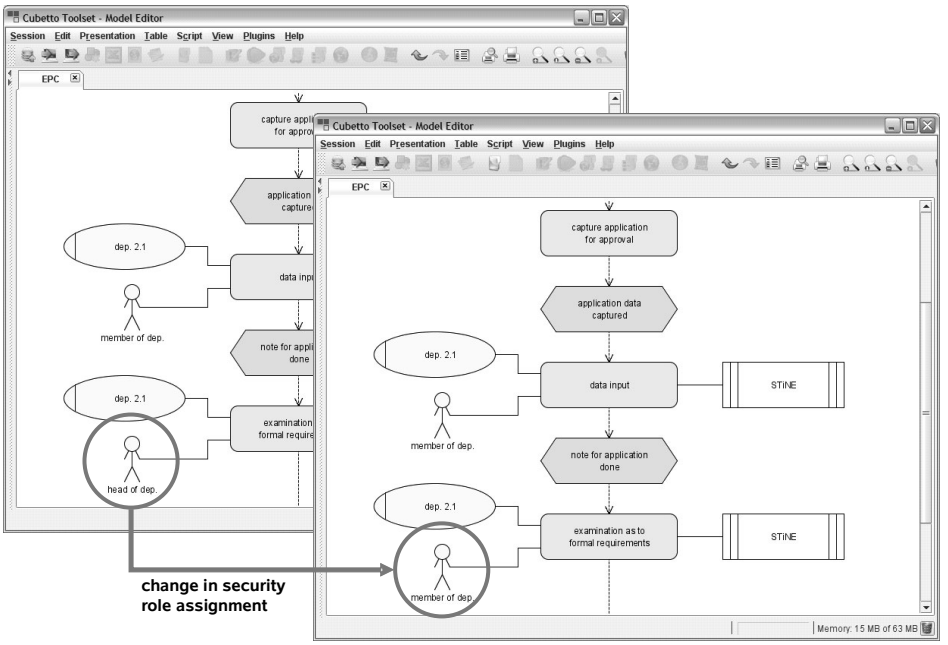

Figure 1. Changed role association to business processes indicate changed access rights

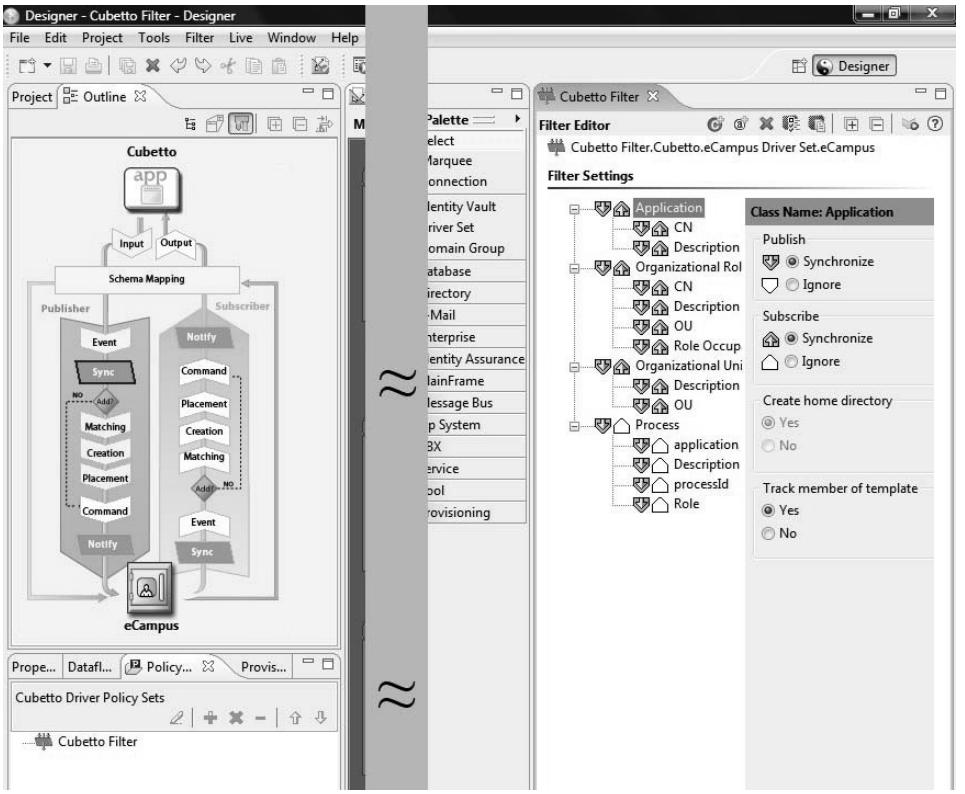

Figure 2. The driver (Novell IDM) for accessing the role data within the Meta-CASE tool Cubetto

4 Conclusion and further research

The presented approach allows us to create web-service descriptions that enable access to the information of the documented processes. Thus, we have created the foundation for a flexible architecture to control and manage rights in the identity management system through existing enterprise models. The reuse of enterprise models by using them to manage application architectures comes up to a major challenge to increasing the use of enterprise analysis methods in businesses and organizations [TC06]. Bringing model information out of the CASE tools to parameterize adaptive software application in an SOA environment helps to extend enterprise model utilization over its original domain of documentation to integration purposes with other software systems [WJE06]. Our future work will focus the technical implementation of necessary software parts of the framework. Until now, we realized the automated generation of WSDL files, but the web-services itself have to be created automatically as well. Beside the technical aspects, we will clarify what information should be added to the model and what data should remain in the identity management system.

Literaturverzeichnis

[BP06] Buecker, A.; Perttila, J.: Deployment Guide Series: IBM Tivoli Identity Manager. IBM Redbooks, http://www.redbooks.ibm.com/redbooks/pdfs/sg246477.pdf, retrieved 2006-04-19.

[Er05] Erl, T.: Service-oriented architecture: concepts, technology, and design, Prentice Hall PTR.

[FK92] Ferraiolo, D.F.; Kuhn, D.R.: Role Based Access Control. 15th National Computer Security Conference, 1993.

[Gr04] Greiffenberg, S.: Method Engineering in Business and Government, Dr. Kovac, Hamburg.

[HIS06] HIS: Hochschul-Informations-System GmbH, http://www.his.de

[JWD07] Juhrisch, M.; Weller, J.; Dietz, G.: Towards a Model-driven Approach to Control Identity Management Systems. In: Proceedings of the 11th Pacific Asia Conference on Information Systems (PACIS 2007), 2007, Auckland, New Zealand.

[JW08] Juhrisch, M.; Weller, J.: Connecting Business and IT – a Model-Driven Web Service Based Approach. In: Proceedings of the 12th Pacific Asia Conference on Information Systems (PACIS 2008), 2008, Suzhou, China.

[Sc01] Scheer, A.W.: ARIS – Modellierungsmethoden, Metamodelle, Anwendungen, 4. Auflage, Springer, Berlin.

[SD05] Stojanovic, Z.; Dahanayake, A.: Service-Oriented Software System Engineering: Challenges and Practices, IDEA Group Inc.

[TC06] Tissot, F.; Crump, W.: An Integrated Enterprise Modelling Environment. In (Bernus, P.; Mertins, K.; Schmidt, G., Hrsg.): Handbook on architectures of information systems; S. 439-567

[Wa02] Walsh, A.E.: UDDI, SOAP, and WSDL: The Web Services Specification Reference Book, Prentice Hall Professional Technical Reference, New Jersey, 2002.

[WJE06] Weller, J.; Juhrisch, M.; Esswein, W.: Towards using visual process models to control enterprise systems functionalitie, Int. J. Networking and Virtual Organisations (3:4) 2006; S. 412-424