Supporting Virtual Organization Lifecycle Management by Dynamic Federated User Provisioning

Wolfgang Hommel
Munich Network Management Team
Leibniz Computing Center
Boltzmannstr. 1
D-85748 Garching
Germany
wolfgang.hommel@mnm-team.org

Michael Schiffers
Munich Network Management Team
Ludwig Maximilian University Munich
Oettingenstr. 67
D-80538 Munich
Germany
michael.schiffers@mnm-team.org

Abstract

For highly dynamic Grid scenarios, scalable solutions for resource, service and access management are essential. However, in today's real world Grid projects, organizations still struggle with system and account management tools that have been designed for intra-organizational use and fail to handle cross-organizational requirements as imposed by Dynamic Virtual Organizations (DVOs). A critical issue in managing the lifecycle of DVOs is the adequate handling of user information when creating and maintaining accounts. Based on a comprehensive Grid scenario, we derive criteria for DVO life cycle management, analyze the shortcomings of existing approaches and present a policy-based approach, which allows for the integration of DVO-management schemes with existing Identity & Access Management (I&AM) systems.

Keywords

Virtual Organizations, Lifecycle Management, Service Provisioning, Federated Identity Management

1 Introduction

When creating Dynamic Virtual Organizations (**DVOs**), such as those considered in Grid Computing [11], multiple autonomous organizations (sometimes called *real* organizations (**ROs**)) will be involved in providing the DVO's services by contributing some of their local resources. Managing DVOs thus necessitates methods and tools to cope with three different layers of abstraction: a resource layer, a service layer, and an organizational layer. While resource management is a classical area of IT management, and IT-service management has come into the fore recently, managing the complete lifecycle of DVOs has not been researched in depth yet.

Resources, services, and DVOs, considered as Managed Objects (**MO**), share a similarly structured lifecycle, consisting of the phases *planning*, *building*, *operating and*

*Parts of this work have been funded by the German D-Grid Initiative under contract 01 AK 800 B.

changing, and *withdrawing* [9, 4]. Traditionally, services used to have a longer lifetime than resources; for example, users of a file storage service did not notice that certain resources, such as a broken hard drive, were replaced or upgraded. However, in highly dynamic VOs, resource lifetimes can easily exceed a service's or even a DVO's lifetime, and resources are not necessarily dedicated to a single service or a single organization anymore. This leads to the problem of coping with interwoven lifecycles of resources, services, and organizations from a holistic perspective.

A critical part in managing DVOs consists of dynamic membership management which refers to both the provider and user communities. Membership management includes the selection of appropriate members (Service Providers (**SPs**) or users) based on their availability, capability, capacity, and on some economic criteria. Any changes in selection criteria will result in updates to this constellation. For each modification of the DVO, such as joining or leaving members or adding or deleting resources, user accounts must be created or adapted and trust relations must be maintained. These activities are generally subsumed under the label *user provisioning*. The cycle of setup, configuration, maintenance and deletion of user accounts is also known as Identity & Access Management (**I&AM**).

In today's real world Grid projects, user provisioning is not handled adequately by the Grid middleware yet. It is often still a manual task, which does not scale well for highly dynamic situations and causes a lot of administrative overhead. We investigate how DVO-triggered user provisioning can be seamlessly integrated into the participating organizations' local account and access management infrastructure by adapting techniques from the area of Federated Identity Management (**FIM**) to DVO scenarios.

The paper is structured as follows: In section 2, we present with EmerGrid a fairly complex Grid scenario to derive DVO lifecycle management criteria from it. Section 3 deals with existing methods and tools for user provisioning and their deficiencies. We present a policy-based approach to user provisioning for the management of DVOs in section 4. Section 5 demonstrates the applicability of the solution in the context of DVO-management, section 6, on the other hand, shortly summarizes an application to the scenarion from section 2. The core components of our approach are being implemented as a prototype, which is briefly discussed in section 7. Section 8 finally summarizes the paper.

2 The EmerGrid Scenario

In order to investigate the challenges of DVO life cycle managemet in more detail, we will use a scenario of an (as yet fictive) Emergency Grid (EmerGrid)*.

EmerGrid (see figure 1) describes an as yet fictive crisis management scenario addressing the correct and effective management of diverse crisis and disaster situations. It has been inspired by the FireGrid scenario [3], the Next Generation Grid Disaster scenario [19], and by the health care scenario described in [15] where Grid technologies have been proposed for improving controlled interventions in emergency cases by synchronizing on-the-spot data (typically read from sensors) with data base contents and high per-

*There is a UK funded *Emergency Grid* project which, however, has a completely different focus.

formance computing facilities for, e.g., simulating the behaviour of tunnel constructions under sudden exposure to explosives with their (sometimes unknown) chemical profiles.

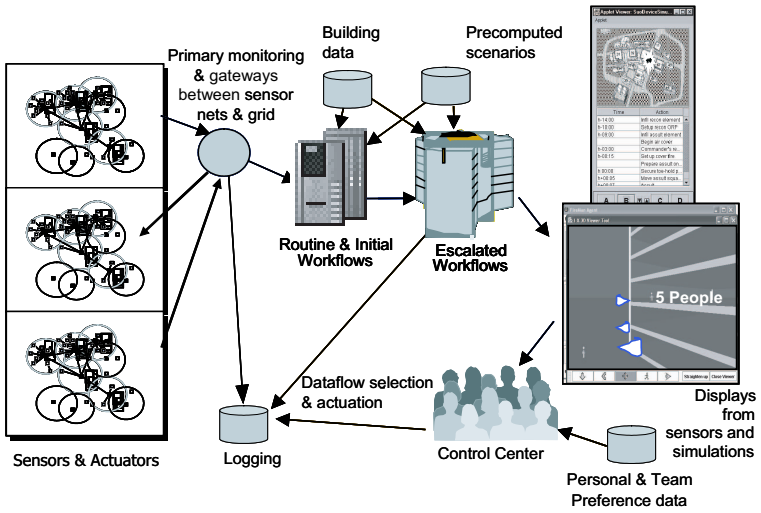

Figure 1: The EmerGrid Scenario (adapted from [3])

In case of emergent crisis events (or training and simulation manœuvres [3]) the crisis response teams will spontaneously form a DVO within EmerGrid (EmerGrid–VO), either instigated by a distinguished principal (a role we call *VO Provider* (**VOP**)) or automatically. Unfortunately, however, it is a priori not known which SPs, resource owners, users, and administrators *need* to contribute to an EmerGrid-VO and which ones *may* support it. In principle, there may exist several SPs which are able to offer the required or similar services, others only under constraints, and again others completely unconditioned since they are legally obliged to participate in EmerGrid. Discovering these providers, evaluating their reputation, selecting an optimal package from available, often similar, service offers, and negotiating adequate SLAs, belong to the crucial issues for EmerGrid just as an adequate user provisioning does. All (dynamically joining) members of EmerGrid-VOs need to be authenticated correctly and authorized – based on their 'home environment' information – to use and/or manage EmerGrid services and resources according to given policies. Since user provisioning in EmerGrid has to be performed in an ad-hoc manner, the requirements for dependability, interoperability, and extensibility are high.

EmerGrid includes a sophisticated component for monitoring QoS parameters and provider performance. Should, thus, the quality of services or the reputation of SPs change over time, EmerGrid-VOs will be re-configured on-the-fly replacing poorly performing services, dropping contract breaking SPs, and accommodating to new requirements induced by situational changes within the crisis event.

3 State-of-the-Art

Lifecycle management of IT-resources and -services in general is not a new research area. Although the respective management concepts are well understood for resources in single domains, mostly small-scale and static environments, they instigate a lot of non-trivial open issues in dynamic, federated environments.

I&AM systems, implemented within enterprise boundaries, typically provide a central identity repository based on a relational database management system or a directory service such as an LDAP server, which holds all relevant user information like credentials, billing information and high-level service access control lists. Often, users can register and update their personal information through web-based portals, while their access rights are being maintained by service administrators or the enterprise's central customer management.

Unless the offered services are, for example, LDAP-enabled and make direct use of this central user data repository, they must be fed with the relevant information about the users that are entitled to use the service. For this purpose, I&AM systems implement so-called *connectors*, through which the account data can be imported in the provisioned services. While most service connectors in use today support the proprietary data format of the provisioned services, the Service Provisioning Markup Language (**SPML**) [17], a recent standardization effort by OASIS, starts to be adopted by most major I&AM vendors. SPML-enabled services implement a Provisioning Service Target (**PST**), which can be managed by one or multiple Provisioning Service Points (**PSP**). SPML defines XML-based messages for creating, modifying, deleting, and searching user accounts, which consist of arbitrary attributes, on the PST. Typically, the syntax and semantics of the exchanged user attributes must match the I&AM system's information model, which varies with the enterprise's preferences, offered services, and deployed software, and thus limits the cross-organizational interoperability of this approach. Furthermore, SPML fails to provide a comprehensive security model, resulting in full access to all user accounts for any authenticated PSP, which comprises a severe information leak in any scenario, in which a service or resource is offered to more than one DVO.

Federated environments pose must stricter requirements on the interoperability, security, and functionality of identity management. The support for different information models, cross-organizational single sign-on, different types of user credentials, and redundancy-avoiding data consistency requires new protocols for FIM. The Security Assertion Markup Language (**SAML**) [5] defines XML-based messages to exchange identity information, which can be categorized into authentication, authorization, or general attribute assertions that are requested by a SP from an Identity Provider (**IDP**). Consequently, each SP trusts the IDPs in the federation to authenticate its users, can delegate authorization tasks to it, and does not have to acquire and maintain the user's personal information itself. SAML, which also is an OASIS ratified standard, is the basis of various sophisticated FIM approaches, such as the Liberty Alliance specifications [23], and Shibboleth [6]. The SAML assertion format is also supported by former competitive approaches, such as IBM's and Microsoft's Web Services Federation Language (WS-Federation). However, the FIM protocols available today have been designed for web sites

and web services in the e-commerce world. While enabling the provisioning of single services, FIM effectively ignores or even bypasses established I&AM systems, resulting in a lack of control about who is allowed to use the services and the scheduling of limited resources, which are offered to multiple DVOs.

With the adoption of Web Services in Grid middleware implementations [2], this deficiency has made its way into DVOs resulting in the participating organizations to potentially loose control over the resources they contribute [22].

A key objective in federated environments is therefore a thorough management of resources, services, organizations, and their orchestration while favouring local control. Several solutions to user provisioning in Grids have been proposed.

The Community Authorization Service (**CAS**) [20] has been developed for the Globus toolkit [10] to solve the access management problem. A major difficulty with CAS relates to its inability to issue Attribute Certificates (**AC**). Instead, CAS issues completely new certificates with the CAS server's Distinguished Name as the subject and the authorization information included in an extension. As a consequence, when a service receives this certificate, it cannot effectively decide who the owner is without inspecting the extension. This means that existing services, in Globus-based Grids, would need to be modified to use a CAS certificate. Another difficulty arises from the fact that a CAS does not record groups or roles, but only permissions, which, however, breaks one of the fundamental rules of Grids: the local administrator needs to have total control about what happens in his domain (again 'locality over globality').

Compared to CAS, Akenti [7] is an AC-based authorization system. However, Akenti is targeted on authorizing accesses to web resources, particularly to web sites. This means that it is completely unfeasible to use it for other needs, for example in a VO. Akenti also does not link identities with groups or roles, but – like CAS – with permissions. Although this is done at the resource side (thus not removing the control from the resource itself, like CAS does), not having any intermediaries will sooner or later lead to fragmentation and inconsistencies between permissions.

The Privilege and Role Management Infrastructure Standards Validation (**Permis**) [8] system implements a Role Based Access Control (**RBAC**) mechanism where ACs are kept in a single AC repository. Certificates are requested from the repository *after* a successful authentication process making it difficult – if not impossible – to obtain and present at the same time information on more than one VO, a critical requirement for several Grid projects [18]. Permis is essentially a policy engine. Permis is best kept on the local sites where the resources it controls are located. Consequently, having multiple VOs in the federation, it will be hard to maintain consistency among the various repositories.

The Virtual Organization Membership Service (**VOMS**) [1] was developed by the European Data Grid project to allow for attribute-based authorization to the Globus Toolkit job management services. It uses X.509 attribute certificates in a push mode to assert attributes in a modified version of the Globus Toolkit. Due to its construction, however, VOMS is not easily interoperable with Web Services-based technologies emerging in the Grid community. VOMS also does not support a pseudonymous mode, nor does it have any other provisions for privacy support. A major weakness of VOMS is its centralistic approach yielding a single point of failure.

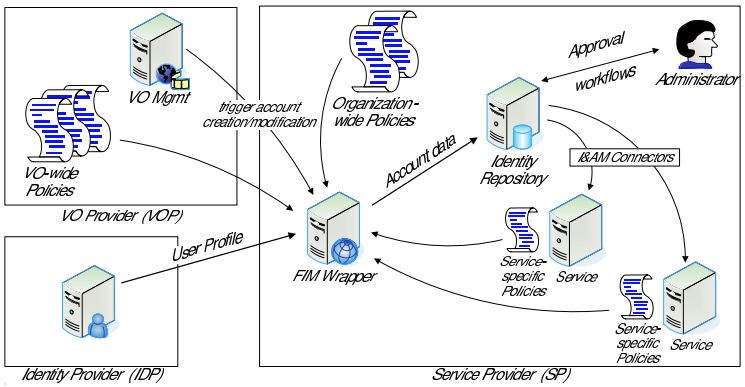

Figure 2: Policy-driven account management based on user profiles

Summarizing the state-of-the-art we conclude major deficiencies in managing a DVO's lifecycle. As appropriate user provisioning represents one of the basic functionalities of adequately managing DVOs we will now emphasize on this topic.

4 Dynamic Identity Management and Federated User Provisioning

Supplying the required user account information for the provisioning of a distributed service is a challenging task in both intra- and inter-domain scenarios. One major drawback of the existing Federated Identity Management (FIM) approaches, which we summarized in the previous section, is that they require that services are FIM-enabled, i.e., that each service must be capable of sending FIM requests and handling the assertion responses itself; this especially includes the task of distributing the user data to the services and their underlying resources. While this communication model is certainly suitable for simple Web Services and web-based applications, it cannot be applied to legacy services or low-level resources in general. In our scenario, we need a more light-weight approach that is easier to integrate into existing service and resource infrastructures.

To facilitate FIM-based user provisioning for DVO management, we propose a gateway component between DVO management systems (run by a role we call VO Provider (**VOP**), see section 2) and the local I&AM systems. At its core, it consists of a policy-driven *FIM wrapper* (see figure 2), which receives account creation, modification, and deletion requests from the VOP through the standard SAML protocol (see section 3) and triggers the local I&AM system to handle them accordingly.

Our FIM wrapper is a Web Service, which is running at each SP from which the VOP requests the creation of an account for one of those services or resources, which the SP has agreed to contribute to the VO (e.g., a simulation service or a sensor network in EmerGrid). Intra-organizational resource and service dependencies are handled by the local I&AM system, leveraging the existing management infrastructure. The VOP only specifies the requested type of operation (create, modify, delete), the affected service or resource (e.g., 10 GB of file storage with a throughput of at least 20 MB/sec), and a user identifier (e.g., a SAML handle).

Once the SP has received the VOP's request, the required user information must be fetched from the user's identity provider. We use policies to specify which user attributes are required for which services, and distinguish between DVO-wide, SP-wide and service-specific policies. These policies may contain arbitrary conditions, such as the rejection of blacklisted users, or the acceptance of previously unknown users only if their full contact information, such as telephone number and email address, is provided. This allows a tighter integration into the local I&AM systems, whose information model typically specify several mandatory and optional user attributes. By using DVO-wide policies, it can be ensured that global objectives will be achieved.

The FIM wrapper retrieves the required user attributes by issuing a SAML request to the user's IDP, resulting in a SAML *attribute assertion*. After verifying the data's policy compliance, the wrapper stores these attributes in the SP's local I&AM repository, by either creating or modifying the user's profile. Then, the I&AM system triggers the account creation or modification in all locally affected services and resources, implicitely handling the dependencies and triggering optional approval workflows.

Special care must be taken about account termination. Usually, the VOP triggers the deletion of a user's account when access to a service is no longer needed. Typically, the account will not be deleted on the service immediately, but only locked and still kept for accounting, billing, statistics and auditing purposes. As an exeption, the SP can also terminate the account, e.g. upon the detection of fraudulent usage. In the latter case, the VOP will be notified and has to handle the situation appropriately, e.g. by setting up an account for a different user, or by choosing a different SP for the same user. While this could be automated through VOP-sided policies, manual interaction might be better suited for such problems.

5 Supporting VO Lifecycles

In the previous section, we described the SP-sided architecture of our solution. Before going into details about the implementation and its application in our scenario, we will now describe the VOP-sided handling of the DVO lifecycle.

By considering DVOs as managed objects, just like services and resources, we leverage established IT management best practices. This approach is reasonable, as DVOs represent an emerging organizational paradigm, which is characterized by its high dynamics, and because DVOs are neither autonomous nor do they 'own' their members or resources.

Consequently, a DVO is a dynamically creatable, manageable, and destroyable object, which is subject to a set of VOP-issued management operations such as `createDVO`, `stopDVO`, and `addMemberOrg`. The main purpose of these operations is to invite ROs to delegate services and/or resources to the DVO and to provide the respective DVO-wide policies as depicted in figure 2.

VOP has been prototyped within the framework of the Open Grid Services Architecture (OGSA) standard [13]. The following lifecycle phases are supported:

Planning In the case of an emergency or a training situation, a DVO profile must be created; this task is supported by generic templates, which can also be derived from previous missions. This profile lists initial requirements for the setup of a suitable

DVO, e.g. the types of required sensor data and processing power, and an optional list of users and their assignment to these services.

Building The DVO is created using the `createDVO` operation, but does not have any members yet. Based on the assembled DVO profile, the VOP starts SLA negotiations with ROs regarding their participation and contribution to the DVO. Selected ROs are added to the DVO using the `addMemberOrg` operation. Once 'enough' members have been discovered to fulfill the set of initial requirements, the DVO is started using `startDVO`. Note that the selected member organizations need not delegate all their members and resources and that the notion of 'enough' requires an appropriate metrics. `startDVO` also triggers one or more `createRole` commands for establishing the organizational context of the DVO.

Operating At the beginning of this phase, the user accounts, which have already been specified in the planning phase, are created on the involved services at the respective DVO members. The FIM wrapper component at each involved SP is used for this service, and the VOP keeps track of created accounts based on the `addMemberUser` operation, which is passed a tuple *(user id, SP, service, service options)*. Optionally, several `assignTask` comands may be issued for assigning users to services.

After this initialization, the operating phase consists of processes such as monitoring and accounting, until a change in the DVO configuration is required, which leads to the *changing* phase.

Changing Changes to the DVO become necessary either due to progress in tasks or due to error situations. Normally, different services and user accounts will be required, e.g. when crisis handling moves from the as-is analysis phase to the solution finding or problem solving phase. A different SP must also be chosen in case a resource becomes unavailable, e.g. due to an unrecoverable hardware failure at the SP. In general, changes are realized by

- determining the new requirements, similar to the specification of initial requirements in the *planning* phase,
- negotiating with ROs about their contribution, as in the *building* phase,
- triggering the deletion of obsolete user accounts, involving the `delMemberUser` operation, and removing ROs no longer required using `delMemberOrg`. Note that withdrawing an RO from a DVO results in deleting all respective user accounts.
- using `addMemberOrg` to integrate the new SPs, and creating user accounts on their services, involving the `addMemberUser` operation described above.

After the change has been completed, the DVO returns to the *operating* phase.

Withdrawing Once the DVO has fulfilled its purpose, or the mission is aborted, the DVO will be stopped using `stopDVO`. In this state, no more changes can be made, but the managed object still exists and can be used for auditing, debriefing, accounting and billing purposes, which are facilitated by a precise logfile of all operations and changes that have been made throughout the lifecycle. Finally, `destroyDVO` will be used once the managed object is no longer needed.

Additional functions, such as `pauseDVO` and `resumeDVO` are provided to support training situations in EmerGrid. Furthermore, `rebuildDVO` combines the *building* and *changing* phases in order to check for a better, e.g. more economic, composition of SPs for the

current service requirements. This, however, requires a transparent migration of running jobs and data from one SP to another; specifying the interface to the Grid middleware, which handles this task, will be part of further research.

6 Application to the Scenario

In EmerGrid, VOPs have been instantiated based on contractual agreements between the potentially participating real organizations. All involved technical components are able to handle multiple, concurring requests. For the sake of clarity, however, the following description is restricted to a single VOP.

In real or training crisis events, a request for the creation of a DVO is sent to a VOP. The required services and resources can be selected either based on an a priori defined crisis profile, or can be specified individually in case no suitable profiles have been defined yet.

As the first part of the DVO creation process, the VOP has to select suitable providers for each required service and resource, by applying an arbitrary calculation, which contains availability, capability, capacity and economic criteria.

If a sufficient – according to an appropriate metrics – set of providers was discovered, the DVO will be created. While 'booting a DVO' (i.e., initially reserving and allocating the required services and resources) the set of account information required for accessing the resources and services must be provisioned before they can be used.

A typical VOP task is now the assignment of users to services using a sequence of `assignTask` commands. In the EmerGrid scenario, this activity is supported by a web-based management frontend in which the crisis commander-in-chief can assign members from predefined crisis response teams according to their expertise. Note that the specialists are directly associated with services and that the services, due to their criticality, are directly coupled to the EmerGrid workflow system and its induced timeline. Note also that the EmerGrid time schedule is flexible enough to allow the VOP to negotiate the acquisition of complementary services in due time.

For each user, the VOP then triggers the account creation by contacting the FIM gateway component of the service provider, which in turn retrieves the user account information from the respective identity provider and feeds it into the provider's I&AM system, which finally creates the accounts on the service and its interdepending resources.

Similarily, changes in service requirements are handled, for example when the crisis management workflow requires a shift from sensor data acquisition over high performance computing to visualization services.

After crisis handling has completed, all remaining accounts and service reservations are deleted, triggering the subsequent accounting and billing processes. Then, the DVO itself will be stopped, and a report about the crisis handling workflow is generated, which serves for later mission debriefing, training scenario generation and overall process optimization. Finally, the DVO will be destroyed and garbage-collected.

7 Implementation

We are implementing the FIM wrapper component on top of the Shibboleth middleware [6], which features so-called *Attribute Acceptance Policies* (**AAPs**) on the service

provider side. We have replaced Shibboleth's built-in AAP engine, which supports only simplistic AAPs in a proprietary format, with a hook to a XACML Policy Decision Point (**PDP**), for which we use Sun's open source reference implementation [21].

We use the XACML policy language elements as follows to control which user attributes are retrieved from its identity provider and whether they are suitable for the SP's local I&AM system:

- The services, which the policy applies to, are the `Resources` in XACML terms.
- The URIs of the requested user attributes are specified as XACML `Subjects` in the usual URI format.
- The different operation types can be encoded as XACML `Actions`. For the specified VOP functionality, we are only using the `read` action.
- An XACML `Condition` element can be used for the formulation of acceptance conditions; XACML offers a variety of statements and functions, which can be used to create arbitrary complex conditions, e.g. based on the user attributes' values or environmental data such as the current date and time, or the requested resources' current load.

The internal workflows for the operations specified in section 4 are realized as follows:

Account creation The following steps are performed whenever the VOP requests the setup of a new account for one of the services the SP contributes to the DVO:

1. The requestor needs to be authenticated and her authorization for creating accounts needs to be verified.
2. The user profile is then fetched from the IDP. While the correctness of this information is trusted, the presence of all required attributes must be verified.
3. The information from the user profile is then used to preliminarily check the user's authorization to use the service. Any mismatches against the defined policies will lead to an immediate rejection of the request.
4. If the user is not known to the organization, a new account is created in the local I&AM system; otherwise, the account is updated with the current user profile data.
5. The requested services or resources are finally scheduled for assignment to the new account; thus, the I&AM's internal workflow management can be used, for example to immediately unlock the resource for the user, or to trigger optionally required approval processes.

Account modifications Changes in accounts are handled by the same workflow described previously, except that the users must already be known locally and different approval workflows might apply.

Account termination Account termination requests are passed on to the I&AM system after authentication. Typically, accounts will not be deleted immediately at the SP, but only suspended or locked, as they are still required for accounting, billing and auditing statistics.

We have chosen Shibboleth as a base for our implementation, as it handles the underlying SAML requests and responses; we plan to implement a more lightweight solution, e.g. based on OpenSAML, as a student project later.

8 Conclusion and outlook

We have derived DVO management criteria from EmerGrid, a fictive, yet fairly complex crisis handling Grid scenario. Besides the handling of the DVO's lifecycle itself, we focussed on the DVO-wide user provisioning process, i.e. the creation, modification and deletion of user accounts on the required services and resources. We demonstrated how policy-based management can be used to automate this process to a large extent, while still preserving the service provider's control over its own resources. Finally, we presented a short overview of our XACML-based prototype.

Our further research regarding Dynamic Virtual Organizations and Federated Identity Management expands in two directions: firstly, we will focus on the definition of interfaces for privacy management between Identity Providers and Service Providers, augmented by FIM components proposed in this and earlier work [16]; secondly, we will finalize the set of DVO-management operations with the objectives to extend the *Open Grid Services Architecture* (**OGSA**) by suitable generic VO management functionalities [12].

ACKNOWLEDGEMENT

The authors wish to thank the members of the Munich Network Management (MNM) Team for helpful discussions and valuable comments on previous versions of this paper. The MNM Team, directed by Prof. Dr. Heinz-Gerd Hegering, is a group of researchers of the University of Munich, the Munich University of Technology, the University of the Federal Armed Forces Munich, and the Leibniz Supercomputing Center of the Bavarian Academy of Sciences. The team's web–server is located at http://www.mnm-team.org/

References

[1] Roberto Alfieri, Roberto Cecchini, Vincenzo Ciaschini, Luca dell'Agnello, Ákos Frohner, Alberto Gianoli, Károly Lörentey, and Fabio Spataro. VOMS, an Authorization System for Virtual Organizations. In F. Fernández Rivera, Marian Bubak, A. Gómez Tato, and Ramon Doallo, editors, *European Across Grids Conference*, volume 2970 of *Lecture Notes in Computer Science*, pages 33–40. Springer, 2003.

[2] Mark Baker, Amy Apon, Clayton Ferner, and Jeff Brown. Emerging Grid Standards. *IEEE Computer Journal*, pages 43–50, April 2005.

[3] D. Berry, A. Usmani, J. Torero, A. Tate, S. McLaughlin, S. Potter, A. Trew, R. Baxter, M. Bull, and M. Atkinson. Firegrid: Integrated emergency response and fire safety engineering for the future built environment. In Simon J. Cox and David W. Walker, editors, *Proceedings of the UK e-Science All Hands Meeting (AHM 2005)*, pages 1034–1041, Nottingham, UK, September 2005.

[4] Luis M. Camarinha-Matos, Hamideh Afsarmanesh, and Martin Ollus, editors. *Virtual Organizations – Systems and Practices*. Springer Science and Business Media, ISBN 0-387-23755-0, 2005.

[5] S. Cantor, J. Kemp, and E. Maler. Security Assertion Markup Language v2.0. `http://www.oasis-open.org/committees/download.php/7737`, 2005.

[6] Scott Cantor, Steven Carmody, Marlena Erdos, Keith Hazelton, Walther Hoehn, and Bob Morgan. Shibboleth Architecture, Working Draft 09. `http://shibboleth.internet2.edu/docs/`, 2005.

[7] D. Chadwick and O. Otenko. A Comparison of the Akenti and PERMIS Authorization In-

frastructures in Ensuring Security in IT Infrastructures. In Mahmoud T El-Hadidi, editor, *Proceedings of the ITI First International Conference on Information and Communications Technology (ICICT 2003) Cairo University*, pages 5–26, 2003.

[8] David Chadwick and Alexander Otenko. The PERMIS X.509 Role Based Privilege Management Infrastructure. In *7th ACM SACMAT*. ACM Press, 2002.

[9] Gabrijela Dreo Rodošek. *A Framework for IT Service Management*. Habilitation, Ludwig-Maximilians-University, Munich, Germany, 2002.

[10] Ian Foster. Globus Toolkit Version 4: Software for Service-Oriented Systems. In *Proceedings of the IFIP International Conference on Network and Parallel Computing*, volume 3779 of *LNCS*, pages 2–13. Springer Verlag, 2005.

[11] Ian Foster, Carl Kesselmann, and Steven Tuecke. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International Journal of High Performance Computing Applications*, 15(3), 2001.

[12] M. Garschhammer and M. Schiffers. Integrated IT-Management in Large-Scale, Dynamic, and Multi-Organizational Environments. In *Proceedings of the 12th Workshop of the HP OpenView University Association (HPOVUA'05)*, Porto, Portugal, Jul 2005.

[13] Global Grid Forum (GGF). The Open Grid Services Architecture, Version 1.0, January 2005.

[14] H.-G. Hegering, S. Abeck, and B. Neumair. *Integrated Management of Networked Systems – Concepts, Architectures and their Operational Application*. Morgan Kaufmann Publishers, ISBN 1-55860-571-1, 1999. 651 p.

[15] H.G. Hegering, A. Küpper, C. Linnhoff-Popien, and H. Reiser. Management challenges of context-aware services in ubiquitous environments. Proceedings of the 14th IFIP/IEEE Workshop on Distributed Systems: Operations and Management (DSCOM 2003), Heidelberg, 2003.

[16] Wolfgang Hommel. Using XACML for Privacy Control in SAML-based Identity Federations. In *Proceedings of the 9th Conference on Communications and Multimedia Security (CMS 2005)*, Salzburg, Austria, September 2005.

[17] Darran Rolls (Hrsg.). Service Provisioning Markup Language (SPML) Version 1.0. OASIS Committee Specification, 2003.

[18] Matthias Kasemann. Computing Coordination Aspects for HEP in Germany. Presentation at the International ICFA Workshop on HEP Networking, Grid and Digital Divide Issues for Global e-Science, Daegu, Korea, May 2005.

[19] NGG2 Expert Group. Next generation grid 2: Requirements and options for european grids research 2005-2010 and beyond. Final Report, Jul 2004.

[20] L. Pearlman, V. Welch, I. Foster, C. Kesselman, and S. Tuecke. A Community Authorization Service for Group Collaboration. IEEE Workshop on Policies for Distributed Systems and Networks, 2002.

[21] Seth Proctor. Sun's XACML Implementation. `http://sunxacml.sf.net/`, 2004.

[22] David Vecchio, Marty Humphrey, Jim Basney, and Nataraj Nagaratnam. CredEx: User-Centric Credential Management for Grid and Web Services. In *Proceedings of IEEE International Conference on Web Services*. IEEE Press, July 2005.

[23] Thomas Wason, Scott Cantor, Jeff Hodges, John Kemp, and Peter Thompson (Eds.). Liberty Alliance ID-FF Architecture Overview. `http://www.projectliberty.org/resources/specifications.php`, 2004.