An Analytical Model of the Service Provisioning Time within the Harmony Network Service Plane

A. Willner
University of Bonn, Germany
Inst. of Computer Science 4
willner@cs.uni-bonn.de

J. Ferrer Riera, J. Garcia-Espin, S. Figuerola
i2CAT Foundation, Spain
Network Technologies Cluster
{jordi.ferrer,jage,sergi.figuerola}@i2cat.net

M. De Leenheer, C. Develder
Ghent University - IBBT, Belgium
Dept. of Information Technology
{marc.deleenheer,chris.develder}@intec.ugent.be

***Abstract*—Grid computing aims at offering standardized access to heterogeneous and distributed resources for scientific communities. However, in order to support emerging next generation Grid applications with specific Quality of Service requirements, the interconnecting networks have also been considered as first-class allocable Grid resources and have been also taken into account for the co-scheduling process. In the last few years, a number of network resource provisioning systems were developed, however, without providing specific analysis on the scalability of potential architectural design alternatives. Our approach is to formulate a fundamental analytical model to evaluate the expected service provisioning time using different architectures as a function of the involved transport domains. To validate our results, we have used measurements obtained from the European IST-FP6 Phosphorus project testbed. The main contribution is to provide an instrument to obtain reference values to support architectural design decisions even in an early stage of the development phase.**

***Index Terms*—Network Service Plane, Grid Computing, Performance Evaluation, Bandwidth on Demand**

I. INTRODUCTION

Since its emergence more than a decade ago in the context of the I-WAY [1] project, the concepts behind Grid computing have become increasingly popular and have been widely adopted in the field of high performance computing. This is compounded by the popularity of cloud computing and virtualisation of both network and IT resources Within this scope, dynamic resource co-allocation and end-to-end dynamic provisioned services became one of the main research areas. The fact of considering the network as a first-class Grid resource and integrate it in the multidomain scheduling process, requires some developments of specific Quality of Service (QoS) features that the underlying infrastructure has to support, and an agreement-based resource management system.

Currently, several so called Network Service Planes (NSPs) architectures are used to provide this capability to a higher level. The challenge is to analyze the scalability of such an NSP considering both job workloads and different available network topologies. As stated in [2], performance measurement analysis has become an important tool to decide the best network topology and must, therefore, be carefully chosen. Defining and executing a performance evaluation of Grid-enabled architectures is not a straightforward task, since the requirements considered in this manuscript are manifold: correctness in the selection of the metrics, realistic workload models, and accurate workload generators [3].

So far, a number of national and international projects have focused on this particular research area of network provisioning with Grid service awareness. The outcome of one of these projects, the IST-FP6 Phosphorus project, was the Harmony NSP [4]. It serves as the basis for the following discussion as its design allows to handle different architectures.

In this paper, we assume a multidomain, multitechnology, and multivendor scenario, where each administrative domain runs under a local Network Resource Manager (NRM). The NSP is populated by different entities. Depending on the relationships and interaction patterns among them and the role of each single entity, the NSP can operate under centralized, hierarchical, daisy-chained, hybrid, or meshed architectures, which are also referred to as deployment models in this article.

Based on these assumptions, we have developed an analytical model, which describes the dependencies and communication workflow between the involved systems. This model is focused on the end-to-end service provisioning time and can be used to predict the expected delays in the service provisioning process.

In order to validate our model we have compared the results with measurements gained from an emulated Harmony testbed. The configuration parameters for both the testbed and the model were acquired from the actual IST-FP6 Phosphorus testbed.

The key contribution of this paper is to provide a methodology to analyze the scalability of a chosen NSP architecture as a function of the involved transport domains given different deployment models. Overall, this paper helps to construct a foundation for further network research and developments in the field of resource co-allocation and generic network service interfaces in heterogeneous environments.

The remainder of the paper is structured as follows. We give a brief overview of related work in the context of dynamic, on-demand bandwidth allocation systems in Sec. II. In Sec. III, we state the problem addressed and the terminology used along the paper. We also present the possible architectures and the communication protocol used within the service plane. In the subsequent Sec. IV the analytical model is presented. The results obtained from performance analysis of the model are given, compared, and discussed in Sec. V. Finally, we close giving some conclusions and considerations and describe future work in Sec. VI.

II. RELATED WORK

Research activities on novel network service plane architectures and protocols are anticipated to address a wide range of innovations. But all previous studies known to the authors in this particular field have not addressed the scalability aspects of different design options.

The IST-FP6 Phosphorus project for example, where the Harmony service plane is contextualized, addresses some of the key technical challenges to enable on-demand, end-to-end network services across multiple, independent, high-performance transport domains. There are several projects aiming at similar challenges: EU Géant project with the *AutoBAHN* system, the Interdomain Control Protocol based on OSCARS, the G-Lambda project, the Dynamic Resource Allocation in GMPLS Optical Networks (*DRAGON*), or the Grid-enabled GMPLS [5] also within the IST-FP6 Phosphorus project. Furthermore, the concept of NSP has increased its presence in the standardization bodies. The Open Grid Forum Network Service Interface (*OGF-NSI*) [6] working group aims at creating a common interface that provides transparent provisioning of the network resources. The GLIF Generic Network Interface (*GNI*) task force aims at similar challenges.

III. PROBLEM STATEMENT AND TERMINOLOGY

A. Motivation

Consider a network consisting of interconnected, heterogeneous, and independent administrative domains, which are used as transit domains. Within this scope, a domain is considered as a high performance network controlled by its corresponding control plane. The control plane may or may not be homogeneous among the whole set of independent domains, where the latter assumption is more realistic. In this scenario, interconnection with third party domains has to be statically proposed, agreed and set up by all involved parties, leading to high delay in interdomain path provisioning.

The problem of how to find, reserve, allocate and instantiate an end-to-end path between two or more resources on demand (considering both intra- and interdomain resources) with certain QoS requirements, while supporting heterogeneous network technologies is treated in detail in [4]. The solution we have worked on, the so-called Harmony system, consists of extending the architecture by means of adding one extra layer, the NSP, over the Network Resource Provisioning System (NRPS) of each transport domain. The NSP consists of at least one Interdomain Broker (IDB) and one Harmony NRPS Adapter (HNA) for each subjacent domain.

In general, the topology of a network seriously affects its reliability, throughput, or even traffic patterns [7]. Likewise, the topology of the NSP directly influences the reliability, performance, and scalability of the service plane itself. Thus, an inadequate choice of the NSP deployment model may lead to the non desirable situation where lower layers are underperforming. Consequently, the problem we consider here in detail is the evaluation and study of the performance of the different architectures supported by an NSP. For each one of these architectures we provide an evaluation and study of service provisioning time when setting up a path request through all involved HNAs and IDBs populating the NSP.

B. Terminology

For the further discussion we will define the architecture, terminology and notations used in this paper. In this context, it is essential to indicate that we strictly distinguish between the *data plane*, which transports the data packets, the *control plane*, which configures the underlying data plane, and the *service plane*, which is used to aggregate different control planes on a higher level.

1) Data Plane: We define the data plane as a directed graph $G_d = (V_d, L_d)$, whereby V_d is a set of vertices called *endpoints* and L_d a set of ordered pairs of vertices, called *links*, that can be provisioned on demand.

2) Control Plane: Endpoints within the same *administrative realm* are configured by NRPSs like ARGON [8] or UCLP [9]. These endpoints are assumed to be fully meshed among each other using *intradomain links*, which for the purpose of this study is a valid assumption due to the higher link density within domains rather than among them.

3) Service Plane: In order to interconnect endpoints between different administrative domains, we define a service plane as follows: directed graph $G_s = (V_s, L_s)$, whereby V_s is a set of vertices called *IDB* or *HNA* and L_s a set of ordered pairs of vertices, called *service link*.

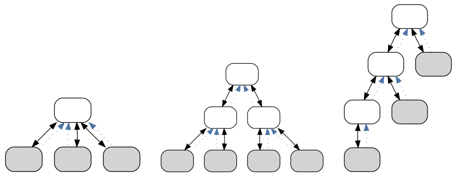
Furthermore, a path in G_d is denoted as $p_d = (u_{d_0}, v_{d_0})$, (u_{d_1}, v_{d_1}), $\ldots$, $(u_{d_{d-1}}, v_{d_{d-1}})$, with $d \in \mathbb{N}_0$, $len(p_d) = d$; $u_{d_0}, u_{d_1}, \ldots, u_{d_n}, v_{d_0}, v_{d_1}, \ldots, v_{d_n} \in V_d$; and each *domain* D with id_d as an unique identifier.

C. Communication Protocol

In order to allow different domains with incompatible NRPSs to communicate to each other, every NRPS has its associated HNA. Thereby, each domain offers the same Harmony Service Interface (HSI) to make reservations and exchange topology information.

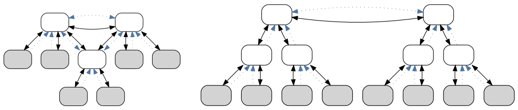
1) Signaling: The signaling protocol within the service plane is responsible for creating, cancelling and querying advance network resource reservations, as well as asking for the availability of a network resource. The actual reservation process follows a non blocking, two-phase commit protocol scheme. A reservation itself contains one or more services and each service contains one or more unicast or multicast connections. Here, a connection is a requested path with certain QoS requirements between one source Transport Network Address (TNA) and one or more target TNAs.

2) Routing: The topology information in the NSP is restricted to a basic set of three elements due to confidentiality reasons: the endpoints, including the border ones, the interdomain links, and the domain itself. Each network point receives a unique TNA identifier within the domain it is attached to. Every domain exports its border endpoints connected to interdomain links and the interdomain links themselves. Hence, a transport network controlled by a single HSI capable system is seen



(a) centralized (b) hierarchical (c) daisy chain

Fig. 1. Basic architectures.



(a) meshed (b) hybrid

Fig. 2. Advanced architectures.

as a cloud with a set of border endpoints. If two or more domains are controlled by a single IDB, this new superdomain will be seen as one cloud with a subset of the original border endpoints. Border endpoints connected to the other domains controlled by the same IDB are kept as intradomain endpoints and consequently they are not pushed upwards.

D. Architecture

As a consequence of the communication protocol design, different possible architectures of the NSP can be implemented. Within the following figures *gray boxes* emphasis HNAs, *white boxes* IDBs, *solid arrows* represent reservation requests, and *dotted arrows* topology exchange messages:

1) Centralized Architecture: As depicted in Fig. 1a, entities within the service plane can be structured in a centralized fashion. This is the most elementary architecture available and consists of one single IDB.

2) Hierarchical Architecture: When layering and connecting different centralized architectures on different levels (cf. Fig. 1b), more than one IDB is involved. Consider two neighbor countries, each having two transport domains controlled by one IDB. One upper IDB controls the interdomain scenario maintaining privacy requirements of each independent transport network.

3) Daisy-Chained Architecture: A service plane populated by entities composing a daisy chain organization can be seen as a specific case of the hierarchical architecture. Each IDB is responsible for only one other child IDB, as depicted in Fig. 1c. Since other projects [10] have chosen this approach we consider it as another comparison.

4) Meshed Architecture: The meshed architecture, as depicted in Fig. 2a, implies that no hierarchy is involved between the IDBs. Each IDB can communicate with any other IDBs on the same level in a peering relationship fashion.

5) Hybrid Architecture: In a further evolution, the NSP is composed of IDBs operating in a mixed mode, as shown in Fig. 2b. When the NSP is deployed in an hybrid approach, hierarchical and peering relationships between entities are allowed. Therefore, a single IDB can act as a peer in a federation of several service plane entities and have children entities at the same time.

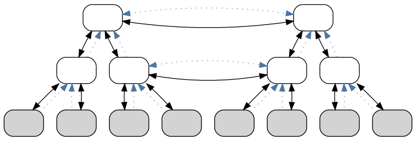


Fig. 3. An example of an unfeasible architecture.

6) Other Architectures: However, due to topology exchange protocol implementation, not all the service plane architectures are allowed to be deployed. Fig. 3 depicts one example of an unfeasible architecture of the service plane, due to the second peering level in the hierarchy. To bypass this limitation in the above mentioned example, a peering relationship between all IDBs must be established (cf. III-D4). Hence, as general rule, the meshed core of the service plane is allowed to have as many hierarchy levels as required, but these levels are not allowed to peer between them.

IV. ANALYTICAL MODEL

First, we consider the *centralized* service plane in order to construct the analytical model. The time an HSI capable service plane entity $v_s \in V_s$ needs to process a reservation request p_d internally is denoted as $t_{v_s}^{p_d}$, and then delegates the reservation request to its child entities, being these child entities HNAs. The communication between them and the parent is parallelized. In detail, the internal time $t_{v_s}^{p_d}$ a service plane entity $v_s \in V_s$ takes for processing the request as a function of the domains involved in the path $d > 0$, whereby $len(p_d) = d$, and the domains $n \geq 0$ under control of the IDB that are not involved in the path request p_d can be spanned into:

$$t^{centr}(p_d, v_s) = t_C^{centr} + t_{var}^{centr}(n, d) \qquad (1)$$

whereby t_C^{centr} represents the fixed time spent for receiving, marshalling, and preprocessing the request, and t_{var}^{centr} is the variable time required (i) in order to compute the interdomain path t_{pc}, including the availability request of the resources involved in the current obtained path; (ii) to forward the request towards the corresponding $v_s \in V_s$ entity, t_{fw}; and (iii) to collect the responses of the child entities, store the results in the database, create the reservation and send it back to the initial requester, t_{db}. These functions are denoted by the following equation ($\gamma \mathrel{\widehat{=}}$ children):

$$t_{var}^{centr}(n, d) = t_{pc}(n, d) + t_{fw}(d) + t_{db}(d)$$

$$t_{fw}(d) = max_{i=1..d \in \gamma(v_s)}(t_{prov_i})$$

$$t_{db}(d) = \sum_{i=1}^{d} t_{db_i}$$

$$t_{pc}(n, d) = \begin{cases} t_{pc}(n) & \text{if } d = 1, \\ t_{pc}(n) + max_{i=1..d \in \gamma(v_s))}(t_{avail_i}) & \text{if } d > 1. \end{cases}$$

whereby t_{db_i}, t_{prov_i}, and t_{avail_i} represent the time spent for each domain $i = 1..d \in \gamma(v_s)$ in updating its state in the database, provisioning the service itself at the NRPS, and determining for the availability of the resources included in the connection request p_d respectively. Considering the *hierarchical* operating mode, the child entities are in this case other IDBs. As defined above, the communication between the parent and the immediate low level of the hierarchy is parallelized. Hence, we can define the total service provisioning time recursively as a function of the hierarchy level h in the service plane as follows (with L_h the set of service nodes on level h):

$$t^{hier}(p_d, v_s) = \begin{cases} t^{centr}(p_d, v_s) & \text{if } |h| = 1, \\ t^{centr}(p_d, v_s) + max_{v_{s_i} \in L_h}(t^{centr}(p_{d_i}, v_{s_i})) & \text{if } |h| > 1 \end{cases} \quad (2)$$

where the case $|h| = 1 \wedge v_s \in L_1$ corresponds to the case that the service plane V_s is only populated by one single v_s in the top and unique hierarchy level. Being as a result, the centralized service plane operating mode.

In case of using the *meshed* model for the service plane, there is a network of service plane entities on top level $V_{s,top}$. Furthermore, considering the hybrid architectures, each service plane entity on top level $v_s \in V_{s,top}$ can be the root of a hierarchy entity structure. Considering a path p_d request for a meshed service plane operating in a parallel mode, we define the total service provisioning time as follows, depending whether or not the resources requested are all under the umbrella of the same service plane entity v_s where the request is received, since we consider the meshed service plane having several points of entry, that is $\forall u_n \in p_d : u_n \in v_s$ for the first case and $\exists u_n \in p_d : u_n \notin v_s$ for the second one, which are also denoted $p_d \in v_s$ and $p_d \notin v_s$.

$$t^{mesh}(p_d, v_s) = \begin{cases} t^{hier}(p_d, v_s) & \text{if } p_d \in v_s \\ t^{hier}(p_d, v_s) + max_{v_i \in V_{s,top}}(t^{hier}(p_{d_i}, v_i)) & \text{if } p_d \notin v_s \end{cases} \quad (3)$$

with $t^{hier}(p_d, v_s)$ to be evaluated with Eq. 2 in both pure meshed or feasible hybrid topologies and considering in the second case that the subpath $p_{d_i} \in v_{s_i}$.

V. EVALUATION

A. Workload

The workload plays an important role in experimental systems performance evaluation. The research community has already presented a broad range of attempts, using different techniques such as fixed interarrival times and Poisson arrivals, or even using traces from conventional parallel supercomputers [11]. Considering the service plane as a whole, it can be seen as a transactional system. Therefore, the performance evaluation is focused on the number of transactions processed under a determined time. The load model used in this analysis considers fixed interarrival times between requests and considers a worst case-fashioned approach where each connection requests p_d involves all the transport networks under the umbrella of the service plane.

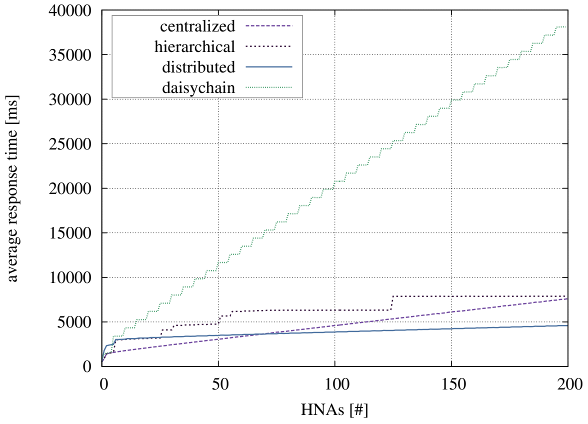


Fig. 4. Average response time while creating a reservation within different architectures as a function of the number of involved HNAs. Except for the centralized architecture a maximum of 5 HNAs per IDB are allowed.

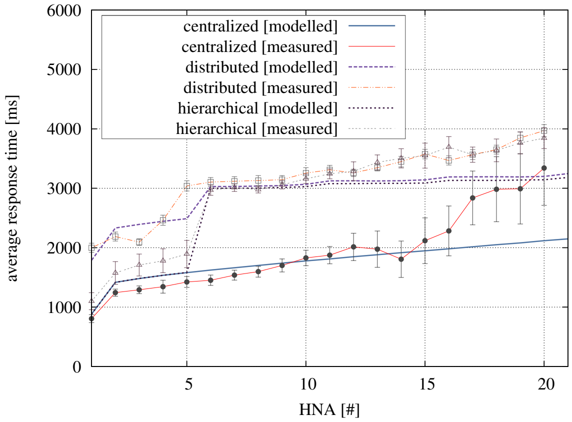


Fig. 5. Centralized and meshed emulation measurements validating the expected analytical model service provisioning time

B. Analysis

Based on the analytical model presented in Sec. IV, the hypothetical response time behavior of each architecture as a function of the involved HNAs is depicted in Fig. 4. While enhancing the number of involved entities (abscissa) the expected average response time (ordinate) for each architecture (lines) was calculated. Using measurements obtained from the Phosphorus testbed [4], a normal distribution was used to model the response time of the different components and the calculations were repeated 1000 times. The confidence interval in the worst case was $\pm 26.7ms$, less than 1% of the total service provisioning time. The used values in ms are as follows: $avg(t_C^{centr}) = 1.8$, $stdev(t_C^{centr}) = 1.35$, $avg(t_{db_i}) = 18.6$, $stdev(t_{db_i}) = 7.507$, $avg(t_{pc}(n)) = 11.37 \cdot n$, $stdev(t_{pc}) = 2.04$, $avg(t_{prov_i}) = 499.4$, $avg(t_{prov_i}) = 57.2$, $avg(t_{avail_i}) = 99.8$, and $stdev(t_{avail_i}) = 19.8$. Furthermore, for this analysis, a maximum of 5 HNAs per IDB for the hierarchical and meshed architecture is assumed.

It can be seen that the average response time of the centralized architecture should be the lowest up to about

70 HNAs. After this the response time within the meshed architecture should be the lowest due to its smaller gradient. Moreover, the hierarchical architecture shows leaps based on the fact that the underlying levels of hierarchy increase with the number of involved HNAs. At about 200 HNAs the response time within the hierarchical and centralized architecture is almost the same. In closing, inherent to its functional principle, the analytical model provides with accurate service provisioning times that allow us to determine the different performance behavior of the operating modes.

C. Validation

The analytical model has been validated by comparison with results obtained from Harmony NSP emulations, by running the prototype of Harmony, which has been deployed over a virtual infrastructure slice provisioned by the European IST-FP7 Federica [12] project.

Fig. 5 depicts the service provisioning times as a function of the involved transport domains in the path request p_d. Again we have increased the number of involved entities (abscissa), calculated the expected, respectively measured the average, response time (ordinate) for each setup (lines). The emulation results present slightly higher or lower values than the expected from the analytical model, although the bounds of the standard deviation obtained remain close enough to the analytical model. Moreover, the measurements confirm the leap envisaged by the model in the hierarchical and distributed model at 5 HNAs. Fig. 5 also confirms the increasing trends predicted by the analytical model. These results allow us to determine that the model designed can be used in order to analytically predict service provisioning times for the different NSP architectures.

Although the comparison of modeled and measured data depicted slight differences, the quality of the analytical model remains and it allows us to achieve the initially expected goals of the modeling process.

VI. Conclusion and Future Work

We have examined the service provisioning time of a multidomain Network Service Plane (NSP) when deploying different architectures or models. We have studied the corresponding delays caused by the centralized, hierarchical, daisy chained, and meshed constructs using an analytical model. In practice, although being a well-known problem in the National Research and Education Network (NREN) communities, the choice of the correct deployment is not yet solved. The presented Harmony NSP allows seamless interdomain cooperation, while the selected deployment model must be carefully studied, in order to obtain good performance in the network reservation scheduling and instantiation, which may seriously affect Grid workflows otherwise. In our work, we have compared the analytical model results with some preliminary emulations performed with Harmony, which show a good theoretical approach to real performance and under specific workloads.

Based on the performance results we have obtained, the centralized approach provides a faster service provisioning time when controlling a small- or medium-sized data plane. However, due to the IDB nature, where the path computing and forwarding process depend on the number of controlled transport domains, the scalability of the centralized approach deteriorates in favor of the meshed model, which maintains lower response times when controlling larger data planes. On the other hand, the service provisioning time of the hierarchical model is critically affected by the delay introduced due to the communication process between hierarchy levels. Nevertheless, the hierarchical model is the most suitable for assuring privacy and security in real environments due to its nature and the strict control of the relationships between entities it can perform.

Future work for the short-term includes a realistic workload building process to further evaluate the performance and scalability of the service plane. Besides a deeper study on hybrid deployments we have planned to go two steps forward with the performance evaluation. On the one hand, we have implemented an NSP simulator based on the Python programming language, which allows creating independent entities simulating a queueing system corresponding to an IDB. This will allow us to further study algorithms for building service plane architectures with relevant knowledge about their performance. On the other hand, in order to emphasize and support the given analysis we want to provide performance measurements of the different deployment models of the NSP using the prototype of Harmony.

VII. Acknowledgments

Research for this paper was partially financed by the EU in the IST-034115 project Phosphorus. We thank IST-213107 FP7 Federica for their contribution and support. The side performance evaluation has been partially executed over the Federica infrastructure. Develder and M. De Leenheer are postdoctoral fellows of the Research Foundation - Flanders (FWO).

References

[1] I. Foster *et al.*, "Software Infrastructure for the I-WAY High-Performance Distributed Computing Experiment," in *HPDC*, vol. 96, 1996, pp. 562–571.

[2] J. Degila *et al.*, "A survey of topologies and performance measures for large-scale networks," *IEEE Comm. Surveys and Tutorials*, vol. 6, pp. 18–31, 2009.

[3] A. Iosup *et al.*, *On Grid Performance Evaluation Using Synthetic Workloads*. Springer Berling, Heidelberg, 2007, pp. 232–255.

[4] A. Willner *et al.*, "Harmony - advance reservations in heterogeneous multi-domain environments," in *NETWORKING '09: Proc. of the 8th Int. IFIP-TC 6 Networking Conf.* Springer-Verlag, 2009, pp. 871–882.

[5] G. Zervas *et al.*, "Phosphorus grid-enabled GMPLS controla plane (G2MPLS): architectures, services, and interfaces," *Comm. Mag., IEEE*, vol. 46, pp. 128–137, 2008.

[6] G. Roberts, T. Kudoh, and I. Monga, "OGF Network Service Interface Working Group (NSI-WG)," 2010.

[7] N. Kamiyama *et al.*, "Network topology design using analytic hierarchy process," in *Comm.: Proc. of the IEEE International Conf. on Comm. ICC*, 2008, pp. 2048–2054.

[8] C. Barz *et al.*, "ARGON: Reservation in Grid-enabled Networks," in *Proc. of the 1. DFN-Forum on Communication Technologies*, May 2008.

[9] E. Grasa *et al.*, "UCLPv2: a network virtualization framework built on web services [web services in telecommunications, part II]," *Comm. Mag., IEEE*, vol. 46, no. 3, pp. 126–134, 2008.

[10] A. Lake, J. Vollbrecht, A. Brown, J. Zurawski, D. Robertson, M. Thompson, C. Guok, E. Chaniotakis, and T. Lehman, "Inter-domain Controller (IDC) Protocol Specification," May 2008.

[11] H. Li, "Workload characterization, modeling and prediction in grid computing," Ph.D. dissertation, Universiteit Leiden, Jan. 2007.

[12] P. Szegedi *et al.*, "With evolution for revolution: managing federica for future internet research," *Comm. Mag., IEEE*, vol. 47, pp. 34–39, 2009.