EPiC Series in Computing

Volume 78, 2021, Pages 59–68

Proceedings of the European University Information Systems Conference 2021

ICT-based continuous innovation and research in German university admissions

Guido Bacharach[1], Winfried Raab[2], Hans Pongratz[1,3], Peter Pepper[1,4]

[1] Stiftung für Hochschulzulassung (SfH), Germany

[2] Leibniz Supercomputing Centre, Garching, Germany

[3] Technical University of Munich (TUM), Germany

[4] Technische Universität Berlin, Germany

guido.bacharach@hochschulstart.de, raab@lrz.de, pongratz@tum.de, peter.pepper@tu-berlin.de

Abstract

The German Foundation for University Admission (Stiftung für Hochschulzulassung – SfH) operates the Germany-wide procedure for awarding university places. With over 2 million applications per run, the system causes a high workload in the supporting technical infrastructure, especially at the end of the allocation period. In order to handle such a load, the SfH cooperates with one of the leading scientific computing centers, the Leibniz Supercomputing Centre of the Bavarian Academy of Sciences and Humanities (Leibniz-Rechenzentrum der Bayerischen Akademie der Wissenschaften – LRZ). The goal of this cooperation is to continuously improve the IT infrastructure via joint research and development. It is a showcase project in the LRZ for coordinated workflows across institutional boundaries. The research results are intended to help not only the SfH procedure, but also other specialized software stacks, to perform optimally through innovative solutions. This article presents the results of the first two years of this research cooperation and the planning for 2021.

1 Introduction

Since 2010, the LRZ and the SfH have been successfully cooperating in the operation of the SfH's specialized procedure. The LRZ is a national and European high-performance computing center and was founded in Munich in 1962. With the development of computer and information technology, the LRZ has grown to become one of the largest computing centers in Europe for science, research and education.

As a leading national institution, the LRZ strives to continuously develop and improve its range of services according to current needs and to optimally orient them to the needs of science. In doing so, it conducts research and development work in the field of informatics and its applications in order to

S. Bolis, J.-F. Desnos, L. Merakos and R. Vogl (eds.), EUNIS 2021 (EPiC Series in Computing, vol. 78), pp. 59–68

develop innovative IT services, open up new IT application fields and ensure stable, professional, highly available, secure and efficient operation of IT services and IT resources based on current IT technologies, among other things (LRZ website, (2021)).

The SfH, based in Dortmund, is a foundation under public law. The main task of the SfH is to carry out and coordinate the application and admission process for courses of study with restricted admission. To this end, the Dialog-Oriented Service Procedure (Dialogorientiertes Serviceverfahren – DoSV) was developed as a web-based system that coordinates applications and admissions in locally restricted degree programs nationwide, thus preventing multiple admissions and increasing transparency for applicants (Stiftung für Hochschulzulassung (SfH) website, (2021)).

The Foundation has been operating the DoSV since the application period for the 2012/13 winter semester. In this system, an online-based data comparison enables the effective distribution of offers for study places. The main aim of the system is to enable applicants to be admitted quickly, without the need for lengthy backlogs, and to avoid vacant places and multiple admissions (DoSV Application portal website, (2021)).

Since the application period for the summer semester 2020, the reformed Central Allocation Procedure (Zentralverfahren – ZV) for degree programs with nationwide admission restrictions has also been an integral part of the DoSV. As a result, admission offers for medicine, dentistry, veterinary medicine, and pharmacy are now also matched with all other admission offers of the study programs participating in the process. The adapted DoSV consists of an application phase, the coordination phase, and the final coordinated move-up phase.

According to SfH figures, 151 German higher education institutions with 1,818 study programs participated in the integrated DoSV procedure in the winter semester 2020/21. This procedure was attended by 309,416 applicants with 2,189,480 applications (Stiftung für Hochschulzulassung (SfH) website, (2021)).

In spring 2019, LRZ and SfH decided to expand their future cooperation on the basis of a “Cooperation Agreement for the Further Development of the Dialog-Oriented Service Procedure for the Allocation of Study Places”.

2 Challenges

The DoSV software is a custom development whose scope and complexity in terms of operation, maintenance and development cannot be compared to standard software. This is mainly due to very strict and partly also very complex legal rules that are imposed on the system by sixteen state laws together with various court rulings, including rulings by the Supreme Court (Bundesverfassungsgericht). The IT systems of the DoSV including their test and training environments as well as further web-based support systems have to be operated in a data center that combines high standards of reliability and security with high performance.

The figures described above concerning applicants, applications and study programs show that the DoSV is a system with a large volume of data. The access profile is very inhomogeneous: phases with low activity alternate with phases of extremely high demand. Extreme usage peaks occur just before the application deadline. This poses high challenges for providing services that are sufficiently performant without causing excessive costs.

Due to the importance of the subject, the allocation of study places and the parallel further development of the DoSV require an absolutely reliable and smooth operation of the DoSV. Problems of a similar nature also affect universities and their systems, so the solutions developed here are relevant to and can be used by all universities nationwide.

3 Goals of the cooperation

The SfH would like to master these challenges together with the LRZ. The DoSV, as a complex and unique software that requires a highly complex, mass-capable, fail-safe, reliable and innovative IT base infrastructure, serves both as a research object and as a prototype for possibly comparable IT infrastructures and applications of other users of the LRZ.

Through this collaboration, the cooperation partners intend to gain medium-term impetus in the necessary expansion of capacities – capacity planning – and competencies in information security – including certification – and in research into user behavior. The SfH applications are characterized by very large user numbers and very unusual usage behavior and load profile.

The insights that can be gained from the operational use of the application cannot be obtained using synthetic load or usage tests. Therefore, it is important for the cooperation partners to be able to use actual operational behavior as a source of knowledge. The knowledge gained from this and from operation can be used for optimization and transferred to other areas.

In addition to the classic operating processes and concepts for availability and resilience, this also applies to incident and incident management, change management, and other service management processes.

The LRZ wants to transfer the knowledge gained in this research cooperation to other services and user scenarios. Conversely, the SfH would like to use the knowledge gained to optimize its own systems and processes and make the corresponding information available to the universities and their manufacturers of student-information-systems.

This cooperation should indirectly provide both sides with important insights for their respective fields of activity.

4 System Requirements

4.1 VMware-as-a-Service

One of the central points for the reliable operation of the DoSV environment is the choice of suitable infrastructures with redundant hardware and mature software. The LRZ always strives to provide both the proven techniques for stable operation and the opportunity for research and further development of the existing infrastructure.

Stable DoSV operation is achieved by symmetrically splitting dedicated hardware in two different fire protection zones. All components are mirrored and thus fail-safe against hardware defects. An independent power supply with diesel generators ensures that there is no loss of performance even in the event of a power failure. All DoSV systems are located in multi-secured shelters and are thus shielded from unauthorized physical access.

The network is also configured via redundant load balancers and switches and, with various VLAN levels, enables extensive separation between the publicly accessible web services and the internal zones for DoSV data processing and management.

By means of virtualization through VMware vSphere, the system load is distributed over 736 cores and 15,616 GB RAM to date and, with the hardware expansion from 06/2021, to 2,784 cores and 40,192 GB RAM. The disk space is provided by a synchronously mirrored metro cluster with a current volume of 2x 166/110 TB (gross/net) via NFS. More than 700 virtual instances with the operating system SUSE Linux Enterprise Server for the central DoSV applications and partly Debian OS for management services are used. The deployment of the VMware platform described can also be referred to as VMware-as-a-Service or, more generally, Infrastructure-as-a-Service (IaaS).

4.2 Cloud-as-a-Service

Are there alternatives to VMware as IaaS? One of the LRZ's central services is High Performance Computing (HPC), which provides a wide range of application-specific computing services for its users, from Linux clusters to supercomputers. For several years, a compute cloud (LRZ Compute Cloud, (2021)) has also been among the available options. The hardware was last renewed in 2019 and, in addition to sufficient CEPH storage, now also has a pool consisting of dual NVidia Tesla V100/16GB GPU systems. Thus, the possibility to test state-of-the-art algorithms is given.

In HPC, however, the focus is on the quantity and less on the stability of the resources. Due to the cost-benefit analysis, redundancy is lacking in many places, so there have to be several interruptions per year, e.g. for maintenance work. The OpenStack cloud installed here is well developed, but does not offer the characteristics in terms of resilience and load balancing that VMware vSphere does. Thus, the compute cloud could be used to develop the DoSV. For productive use, however, too much programming work would be required to replicate the properties of VMware.

5 Results to date

In the two years that this cooperation agreement has now been in place, valuable research results have been achieved, particularly in optimizing the load and performance behavior of complex specialized procedures such as DoSV. With new monitoring, a model was created for further load and performance tests using the DoSV as an example, but also independently of the DoSV as a scientific result. Based on this work, the load and performance behavior of the DoSV could be further analyzed by SfH and optimization measures could be initiated. The results of this research are described below.

5.1 Improving data collection and data analysis as a basic tool for evaluating existing operating concepts and developing new ones

The mandate to the system administration was not only to expand the existing monitoring system quantitatively to all systems listed in the 2019 implementation agreement, but also to add new measurements qualitatively and to store them as measurement series in compliance with data protection requirements. Specifically, an additional or new monitoring system called Checkmk (Checkmk website, (2021)) was activated in 2019 and data collection was successively extended to all server, storage and network instances. Access to the data stock and data retrieval by the monitoring system is guaranteed for selected employees of the LRZ and the SfH.

5.1.1. New Monitoring System Checkmk

It is well known among system administrators that the real challenge of successful server operation lies not in providing a functional system, but in maintaining and preserving it. But how can smooth operation be guaranteed, especially for larger infrastructures with hundreds of servers and thousands of services? This can only be achieved through redundant service design and the use of powerful, reliable and flexible system monitoring. The latter makes it possible to detect problems at an early stage and to react (pro)actively to them. In this way, prolonged service outages, which often mean not only financial losses but also damage to the image of the company, can be specifically avoided.

Checkmk is a monitoring solution developed by the Munich-based company Tribe29 (until 2019 Mathias Kettner GmbH). Originally, Checkmk is based on the very well-known monitoring tool Nagios, but it has been improved in numerous aspects and, above all, its operation has been simplified. In contrast to Nagios, Checkmk provides all the necessary elements in a single package - including around 1,700 proprietary plug-ins for monitoring a wide range of system services, as well as a performance-

optimized monitoring core, the so-called Checkmk microcore. The underlying functional principle is based on the client-server model: The servers (clients) to be monitored send their system data with the help of the Checkmk agent to a central server on which the Checkmk server service is installed, which evaluates the received data and displays it in a visualized form according to freely definable criteria. Which services are to be monitored can be defined very fine-grained centrally via the web interface of the monitoring server.

Checkmk has already been used at the LRZ for some time to monitor internal systems and has proven to be flexible and extremely reliable. Starting in October 2019, as part of the cooperation between the SfH and LRZ, the system monitoring of the servers used for the university startup project, began to migrate from the previous monitoring tool Up.time (Idera Uptime Infrastructure Monitor, formerly Up.time) to Checkmk. The need for a change had already become increasingly clear over the past few years. For example, Up.time could no longer be updated due to the manufacturer's changed licensing policy, and the version currently in use is based on outdated cryptographic procedures and requires outdated Java environments. Updating the Up.time environment would only have been feasible by completely reinstalling and reconfiguring the monitoring system and, above all, by investing heavily in up-to-date licenses. However, with per-host licensing, these would have been significantly more expensive and inflexible than with the Checkmk counterpart.

In addition, the internal use of Checkmk at the LRZ has shown that it is not only functionally equal to Up.time, but superior in some areas - for example, in the options for graphical evaluation of system data. Another advantage is that all configuration steps can be carried out not only via a web interface, but also via the command line.

The experience gained and the now broad know-how in using the monitoring solution make the decision to use Checkmk in the SfH project a logical and consistent step. In order to avoid any interruption of system monitoring during the migration, the Checkmk server component was installed in parallel with Up.time on its central monitoring server. The various SfH servers were then successively included in the new monitoring system. As of January 2020, 426 systems included. Several challenges arose in the process, e.g. a concept had to be developed to integrate the servers into the new monitoring system. With the help of host and service groups as well as the possibility of labeling, a structure could finally be created that would enable the management of the systems and the evaluation of their performance data to be even more efficient in the future.

Furthermore, the connection of the client systems to the monitoring server also had to be realized technically. The client-server connection is encrypted via a secure shell and is additionally secured by SSH key pairs. The Checkmk server initiates an SSH connection of a non-privileged user to the system to be monitored at regular intervals in order to query the status of the system services. Via a so-called command restriction, this user is only allowed to execute the Checkmk agent on the target system before the connection is terminated again. The Checkmk agent itself is a simple shell script that takes care of collecting the corresponding system data.

With the switch to Checkmk and its implementation according to current security standards in the server landscape of the university startup project, the LRZ will be even more flexible in the future and – especially with regard to the possibility of graphically evaluating the data – more powerful than before. The focus was not only on the inventory of new or still missing systems, but also on optimizing the monitoring of existing systems. The extensibility of Checkmk through plugins and specially developed service checks made it possible to adapt the monitoring even better to the specifics of DoSV. For example, it is now possible to monitor the exact, actual memory consumption of Java applications on app servers. In addition, Checkmk now continuously checks the response times of all DoSV websites and sounds the alarm in good time when SSL certificates expire. The central NFS shares used for DoSV services are now integrated into the monitoring, as is the utilization of the NFS storage system. To enable continuous improvement of the DoSV software with the help of the data and insights gained, daily performance reports of the production environment as well as the error analysis environments 1 and 2 are also generated and made available to the SfH.

5.1.2. Data analyses for improvements

Based on the development work described above, the load and performance behavior of the DoSV could be intensively analyzed by SfH and optimization measures could be initiated. The optimization of the load and performance behavior exemplarily for the DoSV was achieved through two activities in 2020:

- With the new monitoring, a model for further load and performance tests was created using the example, but also independently of the DoSV as a scientific result. In the process, the LRZ provided the associated data via the monitoring system. The evaluation of the data as well as the compilation of gained research results and findings is done by the SfH.
- An important aspect is the continuous optimization of the DoSV database infrastructure, which in some cases can produce significant improvements or time savings in the processing of various DoSV processes. In 2020, the SfH payed special attention to the database design within the scope of its possibilities and, with the support of the LRZ, implement and document access times gained with regard to the delivery of monitoring data and the adjustments responsible for this.
- Possible optimizations in the interaction between load balancer and multicore applications for large data volumes were investigated.

The results of this research work are described in detail in an analysis report of the SfH development. The summary of this report up to this point was that the present research work is intended as a “starting point” for the maintenance of the overall system. Pure hardware scaling or workflows for problem solving should be left out for the time being. As a result of all measures, not only this first steps make the system faster and accelerate error analysis, but also the use of the hardware should also be in an improved proportion to the task to be performed.

Load tests and additional measures to ensure operation are planned. Furthermore, regular analyses of the necessary LOG entries with an evaluation of the current classification and, if necessary, reclassification of e.g. LOG level ERROR or WARN. In addition, monitoring tools will be used during operation for the web server, app server and database in order to be able to react early to any problems that may arise. The further steps will be continued in cooperation with the LRZ in the joint research project in the coming semesters.

6 Research work planned in 2021

6.1 Load and Performance Behavior

For the optimization of the load and performance behavior for the DoSV, runtime data was already extracted and evaluated in 2020. From the evaluations, initial conclusions could be drawn about the causes of the system's performance behavior. The supposedly promising causes were implemented in the software.

The next step will be to test the implementations in 2021 and to further evaluate and analyze the deviations. The goal is to optimize performance to the extent that fast and legally compliant behavior of the DoSV is ensured.

6.2 Container-as-a-Service, a new Platform

An additional research focus in 2021 will be the virtualization of DoSV services. The demand for container-based compute resource utilization has increased dramatically in recent years as containers make it easier to build, package, and deploy an application or service and all of its dependencies throughout its lifecycle and across different compute environments. In parallel with the first LRZ trials with Docker and Kubernetes, more and more LRZ customers are starting to virtualize their applications. This leads to the decision in 2020 to investigate multiple professional platforms and to start building a new LRZ service Container-as-a-Service (CaaS) from 2021. The goal is to transition CaaS to a stable user operation starting in 2022, thus adding another service to the LRZ offering. The path to the current trial phase was mainly shaped by Alexander Götz (LRZ) over the last two years. His research and analysis work form the basis for establishing the new LRZ service, which can be referred to as "LRZKube - Kubernetes-as-a-Service/Shared Container Infrastructure@LRZ". This is the provision and operation of (shared) Kubernetes clusters for container infrastructures.

6.2.1. Target Group for CaaS and Implementation Process

The target group for the service, which is already in use in several prototypes on a test basis, initially includes research projects, such as the GeRDI https://www.gerdi-project.eu/ project (Generic Research Data Infrastructure), as well as isolated internal LRZ services in the area of research. In mid to late 2021, initial experience could be gained in the form of further development of the DoSV as virtualized applications. In the long term, the offering of managed Kubernetes clusters is to be extended to all users, especially those who have already gained experience with application virtualization on the LRZ compute cloud or on LRZ VMware environments. With a managed container infrastructure, users no longer have to deal with the sometimes very complicated setup, maintenance and monitoring; they can focus solely on running their applications. Deploying individual Kubernetes platforms or resources in a shared cluster also allows SfH to easily implement container orchestration and reap all the associated benefits. The further development of DoSV is thus becoming more and more independent of the given hardware conditions. Since the SfH is currently considering the transformation of its monolith into a microservice- and container-oriented architecture, the adoption of the CaaS activities of LRZ will give both sides an ideal opportunity to gain experiences about this process. The implementation process to the CaaS production environment provides for three steps:

1. internal process / service; 2. friendly user phase incl. DoSV as virtual apps; 3. official service

6.2.2. Expected Benefit

Kubernetes is already being used in many parts of the IT industry and is also becoming increasingly popular in science in the fields of AI and data science. On the one hand, Kubernetes enables the simple operation of a large number of containers as well as the fast and seamless development of software architectures that are composed of dozens or sometimes several hundred individual services, the so-called microservices. In addition, more and more software systems are being delivered as a collection of containers, greatly simplifying the installation and operation of such systems. With the establishment of CaaS, the LRZ expects to simplify the operation of microservice architectures as well as LRZ services, such as GitLab. CaaS also serves as an architecture for running containers on shared infrastructure: there is already experience with running virtual applications, which in turn will be needed by future software applications. Furthermore, CaaS provides an optimal infrastructure for the development of software and thus forms the basis for the operation of Software-as-a-Service (SaaS) and Function-as-a-Service (FaaS/Serverless Computing). In addition, Kubernetes forms the access to the Cloud Native System of CNCF https://landscape.cncf.io/.

6.2.3. Requirements

The requirements of the new service include guaranteeing a standard SLA for the management/control servers of the customer systems. In addition, special SLAs for customers, such as for SfH, are conceivable. These may include the provision of a dedicated management server. Another requirement is that even if a management server fails, the applications in customer systems remain unaffected or continue to operate without disruption. The new infrastructure is to be monitored using the proven monitoring tools Checkmk, Splunk and possibly Prometheus. Monitoring the management or control server as the central instance of the system plays an important role here. Another interesting option would be to perform installations in a secured environment (without a network connection to the outside). This is known as an air-gapped environment, which is suitable for high-security systems, for example for processing highly sensitive data.

6.2.4. Roadmap

Currently there are two demonstration systems at the LRZ, one platform was realized using Rancher (RANCHER website, (2021)) and is currently available as https://kube.gerdi.org. Another platform was established using Kubermatic (KUBERMATIC website, (2021)) and is currently available as https://kubermatic.gerdi.org. A third platform using OpenShift (RedHat OpenShift website, (2021)) was investigated, but discarded for closer examination. After extensive testing of the above demonstration systems, the choice falls on the use of the Kubermatic framework to deploy individual Kubernetes environments. The available VMware and OpenStack platforms are used to set up the test installations. If necessary, additional physical rack servers can also be added. The following Figure 1 shows the planned roadmap for establishing CaaS in the LRZ. The test systems have been installed. The activation of a Kubermatic staging system is to take place in the coming weeks with the support of the manufacturer. Following this, the “Friendly User Phase” will begin with the involvement of SfH software development. During this phase, the possibility of also converting DoSV operations to containers in the long term will be investigated. After an extensive trial phase, the CaaS service is to be made available to all LRZ customers as professional services from 2022.

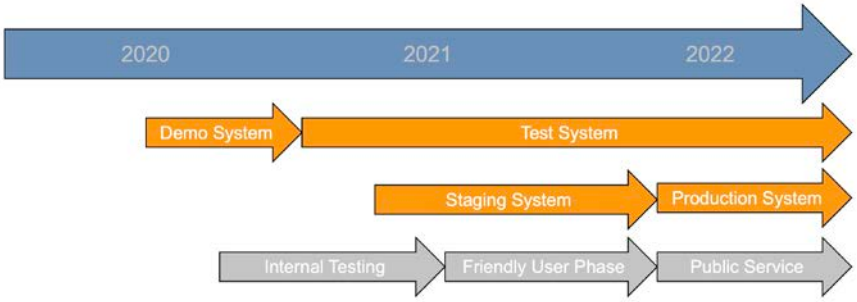

Figure 1: Timeline to the production launch of CaaS

6.2.5. The Virtualization of the DoSV

The challenges of virtualizing the DoSV lie primarily in ensuring security, stability and performance. In order to make an assessment and detect problems here, the first step requires the development and use of component analysis tools. A second step relates to container security with isolation by the host operating system. To obtain more precise information, the monitoring tool used may need to be further developed and optimized. The third step then involves testing mechanisms for managing and securely separating different user environments, i.e. the Kubernetes platforms that can be deployed in each case using Kubermatic.

6.2.6. Demand and Financing

The interest of LRZ staff in CaaS is relatively high now; more and more LRZ customers are asking about offering this service. A relatively large demand is expected from the AI/ML community. Another LRZ scientific cooperation partner has already inquired about a CaaS solution.

The financing model depends on the license costs of the distribution to be used. In addition, there are the investment costs for the infrastructure and the necessary personnel costs. These costs are allocated to billing units. Billing takes place over the period of use, as is also common with other providers, per hour and cluster – cf. AWS/GCE. With Rancher, billing can be per cluster node used and per hour; with Kubermatic, billing is per vCPU and per hour. While a billing system is included with Kubermatic, it would have to be developed additionally with Rancher.

6.2.7. Requirements for Startup

The system requirements for a startup with Rancher or Kubermatic consist of 3 to 6 VMware instances, each with 4 vCPUs, 16GB RAM and 1TB SSD storage, and one VMware instance with 2 vCPUs, 8GB RAM and 100GB HDD for a Gateway/Bastion Host + S3 Gateway. In addition, there are 3 to 6 more OpenStack Cloud instances with SSD storage. For the ETCD backups, which should be made every 3h, a 1TB NAS drive would be needed as well as additional IBM Spectrum Protect (ISP) nodes for archiving. For network connectivity, a dedicated VLAN is required for the cluster as well as a virtual firewall. In addition, there is a staffing level of ≥ 2 FTE for basic operations, for further development and for 2nd level support as well as ≥ 2 students for routine tasks and for 1st level support.

7 Conclusion

As already indicated in the introduction, the focus of the cooperation between the SfH and the LRZ is on the continuous further development and optimization of the IT infrastructure. It must be considered that DoSV operations are always the focus of a broad public. If there are system errors, there are quickly negative headlines in the press. The addition of new technologies must be well planned and tested before they are deployed. While improvements are made with Checkmk in terms of monitoring proven infrastructures, testing with a Container-as-a-Service platform (CaaS) opens the way to the future. CaaS creates a very attractive platform for DoSV to develop software and, at the same time, the production environment that can meet the ever-increasing demand for quality and quantity.

8 References

Checkmk website (2021). *Checkmk Everything monitored.* Retrieved May, 2021 from: https://checkmk.com/

DoSV Application portal website (2021*). Bewerbungsportal für das Dialogorientierte Serviceverfahren (DoSV).* Retrieved May, 2021 from: http://hochschulstart.de/portal

KUBERMATIC website (2021). *Kubermatic Kubernetes Platform.* Retrieved May, 2021 from: https://www.kubermatic.com/products/kubermatic/

LRZ website (2021). *You can count on us!* An information flyer about Leibniz Supercomputing Centre. Retrieved May, 2021 from: https://www.lrz.de/wir/lrz-flyer/lrz-flyer.pdf

LRZ Compute Cloud (2021). *The LRZ Cloud portal.* Retrieved May, 2021 from: https://doku.lrz.de/display/PUBLIC/Compute+Cloud

RANCHER website (2021). *Deliver Kubernetes-as-a-Service.* Retrieved May, 2021 from: https://rancher.com/products

RedHat OpenShift website (2021). *Manufacturing at the edge with Red Hat OpenShift.* Retrieved May, 2021 from: https://www.openshift.com/

Stiftung für Hochschulzulassung (SfH) website (2021*). Das Informations- und Bewerbungsportal.* Retrieved May, 2021 from: http://hochschulstart.de/wir-ueber-uns and http://hochschulstart.de/startseite/statistik

9 Author Biographies

Guido Bacharach, Head of Strategy and Digitization Unit at the Stiftung für Hochschulzulassung (SfH) in Dortmund since 2014. After his study he had managing positions especially in the sales area and in public services. The focus of his work is on strategic digitization, process improvement and project management. He is member of the Deutsche Gesellschaft für Projektmanagement (GPM e.V.).

Dr. Hans Pongratz is Senior Vice President for IT-Systems & Services and the Chief Information Officer (CIO) of the Technical University of Munich (TUM), Germany. He studied informatics and economics at TUM and received a doctorate degree for his thesis "IT architecture for the digital Higher Education Institution". After his studies he worked as consultant and journalist, before he returned to TUM. In 2011 he got SVP (CIO) at TUM and is responsible for the consequent implementation of TUM’s IT-Strategy "Digital University". He is member of numerous boards, committees, reviewer groups, and e.g. co-founder of the digital credentials consortium (DCC). Since February 2021 he is additionally the chief architect of the Stiftung für Hochschulzulassung (SfH).

Winfried Raab is working as systems specialist for the Linux operating system at the Leibniz Supercomputing Center (LRZ) since the beginning of 2000. After initial work on setting up and operating an infrastructure for Linux workstation PCs, the focus of his work is on operating a high number of physical and especially virtual server instances. The Linux servers form the basis both for numerous LRZ services and for the LRZ's customers, which include the Stiftung für Hochschulzulassung (SfH). His team is embedded in the High-Performance Computing department and, with the exception of the Linux clusters and supercomputers, represents all Linux installations of the LRZ.

Prof. Peter Pepper studied Mathematics at the TU Munich, where he also obtained his PhD and his "Habilitation", both in Computer Science. In 1981 he was a Research Fellow at Stanford University. From 1985 to 2016 Prof. Pepper held a chair in "Programming Languages and Compiler Construction" at the TU Berlin. Over several years Prof. Pepper also had an affiliation as Research Coordinator with the Fraunhofer FIRST institute in Berlin. Moreover, he was heading a group at the DCAITI, a joint research institute of TU Berlin and Daimler AG. Since 2018 he is CIO (Technischer Geschäftsführer) of the SfH. Prof. Pepper was a member of various scientific organizations, notably of the IFIP Working Group 2.1, and of the EASST, where he acted as a member of the Board during the founding years. He also was a member of the Modelica Association, which is responsible for the design of the standardized Modelica modeling language.