

Performance Improvement Barriers for SAP Enterprise Applications: An Analysis of Expert Interviews

Adrian Streitz

Technische Universität München
Garching, Germany
adrian.streitz@in.tum.de

Harald Kienegger

Technische Universität München
Garching, Germany
harald.kienegger@in.tum.de

Maximilian Barnert

Technische Universität München
Garching, Germany
maximilian.barnert@in.tum.de

Helmut Krcmar

Technische Universität München
Garching, Germany
krcmar@in.tum.de

ABSTRACT

Performance evaluation with regard to response time of software applications is a crucial task. In particular, this is essential for Enterprise Applications with high demand for time-sensitive transactional operations. Although the effects of neglecting performance considerations within the software development life cycle are known, the development process has not significantly changed over the last years. However, companies' interest in software performance is increasing. This paper identifies the barriers regarding performance improvement of software applications during early stages of the development process by focusing on SAP Enterprise Applications. In order to capture recent situations within the industry, we conducted expert interviews with both experienced SAP software developers and product managers working in different industry sectors in Germany. Our key findings show a range of different reasons for poor or missing performance improvement considerations.

KEYWORDS

Software Performance Evaluation, SAP ERP, Response Time, Performance Improvement Barriers, Software Development Life Cycle, SAP Enterprise Applications

ACM Reference Format:

Adrian Streitz, Maximilian Barnert, Harald Kienegger, and Helmut Krcmar. 2018. Performance Improvement Barriers for SAP Enterprise Applications: An Analysis of Expert Interviews. In *ICPE '18: ACM/SPEC International Conference on Performance Engineering, April 9–13, 2018, Berlin, Germany*. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3184407.3184434>

1 INTRODUCTION

Well-defined business processes are essential for every company and form a fundamental structure to aim for a certain business goal. Moreover, the success of companies in a competitive environment associates that all its business processes run effectively

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ICPE '18, April 9–13, 2018, Berlin, Germany

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-5095-2/18/04...\$15.00

<https://doi.org/10.1145/3184407.3184434>

[12]. In the industry, most business processes are supported by Enterprise Resource Planning (ERP) systems. Therefore, companies are dependent on the IT infrastructure where these processes are used. Market leader for ERP systems is SAP SE with 23% market share, as stated in a report by Panorama Consulting Solutions [15]. In Germany, this share is almost twice as big - 46% [1]. Therefore, companies have to rely on stable and efficient SAP ERP solutions. Interruptions and bad performance, which cover not only response time, but also throughput and resource utilization [3], lead to financial damage due to unused working hours, forgone turnover or delays within production lines.

According to Brunnert et al. [4], there are two possibilities to ensure performance goals with regard to predefined metrics. Software performance engineering [18] describes how to achieve performance goals during system development. Application performance management [13] is the counterpart for providing stable performance during the operation phase of a software product. Both organizational units follow different paradigms for their work. While development teams aim for realizing new functionalities in a short period of time, operation teams are interested to keep the entire IT landscape in a stable state. However, they are both crucial in order to achieve the performance goals together.

From an operational point of view, performance metrics are defined to ensure service-level agreements by optimizing system parameters on infrastructure, operating system or software level and dynamic resource allocation. Instead, development teams focus on optimizing algorithms, data queries respecting database indices and quality of source code. Following the idea of Shen et al. [14], it is reasonable to put more effort in the development phase, since software changes later in the development life cycle increase overall product costs significantly. Performance tests of Enterprise Applications are one possible solution, but at the same time difficult to perform during the development phase of a software product [18].

Although there are existing tools in SAP ERP systems to test and analyze performance metrics of SAP Enterprise Applications, e.g. Single Transaction Analysis, SQL Performance Trace or ABAP Runtime Trace, which support SAP developers during implementation, 43% of SAP end-users are not satisfied with the performance of their daily in-use SAP Enterprise Applications [6]. In order to close this gap, we want to identify barriers which hinder performance improvement during the development phase of SAP Enterprise Applications.

The major goal of this paper is to show the current situation of performance evaluation procedures in companies relying on SAP ERP software and developing Enterprise Applications within this ecosystem. This includes the identification of current barriers in the applied performance evaluation process and a following discussion of approaches. The contributions of this paper are: (1) to present barriers considering performance improvements for SAP Enterprise Applications and (2) to discuss and come up with ideas to address the identified barriers in order to improve the process of performance evaluation for SAP Enterprise Applications.

The paper is structured as follows: in Section 2 we present the environment and surrounding components of an SAP Enterprise Application to highlight the main differences to other Enterprise Applications. Section 3 inspects related work in this area. Section 4 describes the applied approach, gives insights about the interview partners and summarizes the open-structured interview design. In Section 5 we outline our findings extracted from the expert interviews conducted. We follow up with a discussion in Section 6 and address ideas for future work. Section 7 summarizes and concludes the paper.

2 SAP ECOSYSTEM

In this section, we present the SAP ERP ecosystem with its programming language ABAP for which we consider all following SAP Enterprise Applications. Before ABAP is described as programming language, the concept and the background are mentioned. As the client-server-based development of business applications requires a different approach than the development of locally executable software, an overview of the SAP architecture, the modular concept and the development environment is important for the developer.

ABAP as programming language is thereby intended for dialogue-oriented database applications. The business and technological environment must provide a basis for many users working simultaneously on a shared database. These circumstances require an appropriate data security and system architecture, which is provided by SAP ERP. The SAP Business Suite complements it with some further software applications.

For legal reasons, certain data must be made available for several years or even decades. That is why SAP ensured its independence by introducing their own programming language. As one of the key values, today's ABAP interpreter are still able to process source code from early day programs. This is also the basis for an open and expandable system [11].

On the other hand, these opportunities require a modular design of the system. From a technical perspective, this results in a distribution on several servers and services. From a business perspective, different functional modules are set up and complemented by customer or industry specific solutions, e.g. banking or healthcare. Common SAP modules of ERP systems are Production Planning (PP), Material Management (MM), Sales and Distribution (SD), Human Resource Management (HRM) and many more. But even with those modules, all users work with the same data pool to avoid redundancies and isolated applications [11].

3 RELATED WORK

Compared to Enterprise Applications from other global IT players, SAP Applications are known for their long execution times. Simple tasks as handling billings from business trips or requesting planned holidays using SAP Applications end up with bad performance. This performance results even though the SAP system and the utilized applications are deployed on premise. In contrast, Amazon Online Store, as famous and known example, performs much faster, although the Enterprise Application is not deployed locally. Instead, it has to deal with additional latency issues caused by the individual internet connection of each customer.

With the intention to improve performance considerations on a theoretical basis, Tüma defined in 2014 the term Performance Awareness [16] as an act of performance observation with a follow-up procedure to reduce performance anomalies. According to Tüma [16], without the respect of Performance Awareness in early stages of the software development life cycle, a long-term balance of application performance against development effort and maintenance costs cannot be achieved.

For a component-based system, as the SAP ERP system is intended to be designed, one major factor besides the deployment platform and the dispute of resource allocation is the fact of how efficient the component has been implemented [10]. Since receiving insights on performance of Enterprise Applications gets increasingly difficult due to factors like continuous development [4], it is important to reduce the complexity of a feedback for the developer to a minimum.

As a cooperation project between Compuware Corporation and Pierre Audoin Consultants (PAC), both companies conducted a trend study on the topic of SAP Performance Management in the year 2010 [6]. The trend study was performed with 588 companies in different industry sectors employing at least 500 people and using SAP in their production environment in Belgium, France, Germany, Italy, the Netherlands, Spain, United Kingdom and the United States. The survey covered three main topics: (1) current rating of SAP Enterprise Application performance, (2) financial risks considering SAP performance problems, and (3) applied solutions for SAP Performance Management.

Although all companies run critical business processes on their SAP system landscape, almost 43% complained about bad performance of their SAP systems and 40% consider not being prepared for real-time performance identifications. 96% answered that there is a financial risk, if SAP performance problems arise and 50% of them plan to invest in software solutions to monitor performance values of SAP Enterprise Applications in order to provide the necessary service quality.

Following this purpose, Compuware developed a software product based on the information from the trend study to provide a wide-ranging performance management tool. However, it does not reach out to the developers who are in charge for creating an efficient application.

Our analysis of conducted expert interviews identifies barriers for performance improvements of SAP Enterprise Applications and offers at the same time a research roadmap presenting an approach to address the problem of complexity by reducing the provided performance information for the developer to a minimum.

4 RESEARCH DESIGN

The status quo of current literature for performance consideration in software development processes of SAP Enterprise Applications is rather scarce. Due to lack of theoretical knowledge in the field of performance evaluation for SAP Enterprise Applications, we decided to conduct interviews with experts in the field of Application Development for SAP. According to Bogner et al. [2], expert interviews are a suitable qualitative assessment method since they concern expert's perception. Moreover, it enables open questions for personal opinions and insights in current SAP development processes. In our case, we made use of this research design to discover current situations of the software development processes for the SAP ecosystem and draw parallels to existing statements.

The performed research design follows the rules of an open-structured interview. This means the interviewer has a predefined interview guide with multiple questions. However, neither the exact formulation, nor the order of questions is determined. With this strategy, we allow even additional questions fitting in the current discussion context revealing new interesting issues on the research topic. After a fixed prolog, in which the interview partner and we as interview initiator introduced ourselves, we started asking about the personal experience and the current occupation. Afterwards, no matter in which direction the interview has moved, all experts contributed to the following main questions:

- Is performance considered in current SAP development projects?
- Who is in charge for fixing performance problems of SAP Enterprise Applications?
- Does performance reflect a critical success factor for SAP Enterprise Applications?
- How do you measure performance of SAP Enterprise Applications?
- When do you discover performance issues of SAP Enterprise Applications?
- What are main barriers evaluating performance in early stages of the SAP development process?

4.1 Interview Partners

In a first step of the interview process, we had to look for a suitable target group. According to Glaeser and Laudel [8], the selection of the interview partners is an important task, since a good set of interview partners is crucial for the quality of gathered information.

Since we require knowledge about SAP Enterprise Applications, the resulting performance values and the software development process, we focused on a target group that is currently involved in the field of either IT Development or IT Operations.

To get in touch with people who were willing to share their experiences, we contacted first SAP mentors who are key partners in certain technical fields, like SAP Development, by email. Besides, we made contact with the DSAG¹ who initially created a new initiative for the topic DevOps². Together, we collected six interview partners, principally SAP developers with more than fifteen years of ABAP development experience. Some of them act currently with different

¹German community of SAP users located in Walldorf; members are from all kind of business sectors

²A software engineering concept combining software development (Dev) and software operation (Ops).

Table 1: List of Interview Partners

Expert	Business Sector	Job Position	SAP ABAP Experience
A	SAP Consulting	CEO	8
B	SAP Consulting	IT Development	10
C	Automotive	IT Operations	9
D	Insurance	IT Development	7
E	Software vendor	Division Manager	4
F	Software vendor	IT Development	8

roles, e.g. as CEO, division or product manager. All available experts are listed in Table 1.

As part of the interview, all experts had to assess themselves regarding their SAP ABAP development experience on a scale from 0 to 10, where 0 means having no experience and 10 being a full ABAP expert. This information is integrated in Table 1 in the fourth column. Because most of the interview partners are spread over Germany, all interviews, except of one, have been conducted via telephone. The interviews lasted 35 minutes on average. In order to capture the interviews entirely, they were all recorded and transcribed afterwards.

5 FINDINGS

In general, all interview partners confirmed the main result statements from the trend study described in Section 3 for SAP Enterprise Applications. Although, the trend study has been conducted in the year 2010, which reflects a big time period for IT evolution with technical changes of SAP components of the ERP environment, the raised statements are still valid today. *"Until now, performance has always played a tangential role. However, it is set to be more focused in the future"*³. Regarding the statements of the conducted interviews, some aspects of the trend study have been described in more detail by the experts and combined with the current situation within the development divisions.

All interview participants agreed consistently that performance evaluation regarding SAP Enterprise Applications is still an important topic that should not be forgotten to be considered. However, some of them admitted that present development projects often do not consider performance evaluation at all. *"Performance or load tests are not performed at all (...). This is because from a strategical point of view, there is currently no demand from the management for it"*. According to the experts, today's software development projects focus primarily on functionality, clarified in the requirements engineering phase, or software usability due to time and budget constraints. *"First of all, the application has to run and the compliance has to fit. If there is some time left, we can look after the application performance"*. Nevertheless, almost all interview partners mentioned a trend towards higher integration of performance evaluations, especially for SAP Enterprise Applications, in the software development process for the future. The basis for that statement is the increasingly complex software and system landscape of the entire IT environment. *"In a certain part, the current trend is to put*

³All interviews were conducted and transcribed in German language. For this paper statements have been translated by the author.

everything on high performing machines (...). Of course we can put a bad performing application on high performing hardware, but for a permanent solution this is very expensive (...) I am certainly convinced that machines are not able to solve problems of bad designed software applications".

Regarding the question who eventually should be responsible to fix performance problems, the opinions differ from expert to expert. Some interview partners issued the statement that performance consideration is mainly a task of an SAP Basis consultant ⁴. Others however, think that SAP developers should be skilled in the topic of application performance. Concerning our proposition of an extra occupation for a specialist in performance engineering, all interview partners were clearly against it. Performance expertise should not be focused in a new job role, but rather be integrated within existing ones, e.g. software developers, administrators or project managers. *"If a software developer has no feeling and knowledge about performance criteria, he is not a real developer but rather a programmer".* Nevertheless, it is preferable to create a new taskforce which consists of software developers and system administrators. Both groups should complement each other. *"If a software developer does not succeed by himself, it is a good idea to get together with people from SAP Basis"* and exchange knowledge in order to manage the issue together. This central idea follows the aspect of bridging the gap between the organizational units of the development and operations division.

Regarding the importance of performance for SAP Enterprise Applications, the experts had the opinion that this non-functional requirement reflects a critical success factor for each single software application. In combination with direct visibility for customers, e.g. through a web application using a SAP Enterprise Portal in the background, or a SAP Enterprise Application which is used by employees of the IT support division communicating with customers, the sense of importance for performance evaluation is much higher. *"Colleagues from the support division often need information in a certain time window. (...) If a customer is on the phone, we need quick access in order to provide further information. There is no time for delays".*

By directing the focus towards performance measurement of SAP Applications in today's companies, we had to determine that performance is principally measured against a subjective feeling. *"There are no SLAs, KPIs or other objective values, which we use for evaluating our SAP Applications at the moment"* was the answer of one of our interview partners. Application users who get in touch with a new software are either happy because of the given functionality which reduces time effort of a certain task, or they complain about missing functionalities and long response times of the new application. Depending on the user's subjective acceptance, performance anomalies are either discovered or not.

Regarding the initial discovery moment of bad performing SAP Enterprise Applications, all interview partners agreed that the identification of performance problems happens too late in today's software development life cycle. *"Performance consideration during the development phase of a software product has played a tangential role in the past (...). Performance anomalies are only discovered because they occurred during functional testing." "It is possible that some*

software developers look after performance of their own programs just for fun. But in general, they do not. So it is postponed to a later point in time". Another *"big problem is that programmers are often provided from external consulting companies (...) and performance tests are performed after a certain milestone of the software project has been reached. This leads to the circumstance that the programmer of the software code is not available anymore when the problem arises".*

When we asked our experts about current barriers leading to a late consideration of performance in the life cycle of a software product, we noticed that one of the mentioned barriers was consistently the same. Almost all experts pointed out the fact that nowadays software developers have a non-existing knowledge of software performance. *"There are software developers en masse who have not been confronted with performance issues of software applications before".*

Table 2 lists further results of identified barriers from our conducted interviews ranked by number of occurrences. Each barrier will be discussed in more detail in Section 6 and enhanced by possible solutions when given by the experts.

6 DISCUSSION

In this section we want to discuss the identified barriers and possible solutions to encounter them in an appropriate way. First of all, we want to present the current situation of performance consideration and performance evaluation for SAP Enterprise Applications in a nutshell. Based on the results from the conducted expert interviews, we describe the problems and barriers in detail. The individual points that have been covered will be enriched by further comments of the experts and the ideas of how they could be resolved. Finally, we want to present a promising approach called Performance Awareness [16] with an ongoing research roadmap.

Our findings of the conducted interviews show a current lack of performance consideration of SAP Enterprise Applications. If performance evaluation is considered at all, it is often just a subjective feeling by asking end-users about their overall satisfaction with the application. This does not reflect the sense of a real performance evaluation against certain KPIs or SLAs. As a good example, those objectives should be provided with a given technical specification created in the requirements engineering part of a software life cycle process.

6.1 Problems and Barriers

The most specified barrier against performance improvements of SAP Enterprise Applications is that *"SAP software developers do not have sufficient knowledge about software performance or performance tuning".* *"We have currently a lack of performance qualification for software developers. Every software developer follows an individual programming style based and extended on personal experience, technical opportunities, trainings, trial-and-error proceedings, and much more".* However, they are not taught *"in methods to look after software performance, e.g. how do I have to construct a performance optimized program operation or SQL statement".* *"Available training sessions, performance seminars, or other qualification courses are very rare".* At this point, we could offer more opportunities in order to sensitize software developers by paying more attention to performance aspects of software applications.

⁴Special job position term for a system administrator in the context of SAP

Table 2: Performance Improvement Barriers for SAP Enterprise Applications

Rank	Barrier	Expert Mentions
1	Missing knowledge about software performance	A, B, C, D, E
2	Different characteristics of SAP development, evaluation and production system	A, D, E, F
3	Bad communication between IT Operations, Development and Specialist Department	C, D, E, F
4	Missing dedicated experts for software performance	C, D, E
5	Missing information about existing SAP tools to analyze software performance	A, B, C
6	Unavailable production workload on SAP development and evaluation system	C, E
7	Handling of existing performance tools complex and complicated	D

Another problem considering performance issues, as early as possible in the software development life cycle, is that applications are initially developed on another system with different characteristics than the production one. SAP propagates as best practice to establish a system landscape divided in at least three parts, providing a development, quality assurance and production system. In an optimal way, the development system represents a sandbox environment where all necessary SAP components but no data according to the target system are installed. *"It makes a huge difference conducting performance and load tests on real data sets or generated example data, which does not reflect the real world scenario afterwards"*. Therefore, it is crucial to generate or consider real data sets by not affecting the production system by itself.

The next aspect claims a bad communication between colleagues from IT Operations, Software Development and Specialist Departments, e.g. Sales, Logistics or Human Resource Management. *"Many users from the Specialist Departments adjusted to the fact that certain SAP Applications perform badly"*. So to speak, they get used to this situation step by step and *"do not report upcoming performance problems immediately when they notice them"*. Consequently, the actual *"problem stays undiscovered until the application gets used on a massive scale"*. A reasonable solution to encounter the existing communication gap is to integrate people from Specialist Departments deeper into software development projects. Hence, feedback from an end-user's perspective reaches effectively the responsible person from the software development department.

In contrast to the first aspect where software developers should be trained with topics of performance issues, e.g. being aware of memory consumptions of certain operations or iterating over a ton of data which is not necessary to end up with the needed result, there is also the idea of a *"dedicated IT expert regarding performance engineering of software applications"*. Today's *"software developers are not feasible to meet all demands with the same quality"* because of the increasing complexity of current and future software projects. This does not mean that software developers should not be aware of performance at all. In fact, they should *"have knowledge and insights of developing well performing applications"*, but at the same time be open-minded to work together with people from IT Operations or in this case with performance engineering experts.

Despite, SAP already offers existing tools to gather information about different performance and software metrics. However, the information about existing tools is not available to all SAP developers. *"There are for example SAP tools to analyze the execution plan of an SQL statement regarding its JOIN conditions (...). Using this tool, you*

will get information about created indices and the fact whether they are used in your certain statement or not. If not, you may continue your investigation about the root cause".

Even though the development system reflects the same characteristics as the production one, the production workload is still missing. The workload on the production system is essential since it affects resource allocation and response time of other applications executed at the same time. *"In addition, big companies perform a lot of batch processes. This means automatically that at a certain time during the day the number of SAP dialog work processes are reduced and SAP batch processes are added instead (...), vice versa for the daylight time"*. This behavior has to be taken into account if we are interested in reliable performance evaluation results.

As the last point, one of the interview partner referred to the fact that even when people, in particular software developers, are aware of some existing SAP performance tools, they often do not use them because of a complicated handling. The mixture of not being a performance expert on the one hand and the existing time pressure to finish the software project by a given deadline on the other hand, directs the decision of each single software developer *"to fulfill all function requirements first"*. Moreover, *"the compliance of the software application has to fit"*. Only thereafter, and with respect to the remaining time for the application development part, software developers look after the application performance.

6.2 Performance Awareness

One response to the lack of performance consideration of software applications is the concept of Performance Awareness [16]. Tüma presented the term at the ICPE 2014 as one of the keynote speakers. Performance Awareness defines the ability to observe performance, detect problems and react to them. A major part of this concept concentrates on the support of software developers providing insights on performance of application source code that is currently developed.

A number of approaches already encounter the issue of supporting software developers to increase Performance Awareness during development. The existing approaches may be classified in either measurement-based or model-driven performance engineering ones.

For the first category, Bureš et al. [5] propose an approach to formulate performance goals during the design phase, even before the software application is started to be implemented. Afterwards, their approach collects performance measurements during application runtime and presents the information to the developers. Weiss

et al. [17] aim at a similar approach, where lightweight and tailored benchmarks are automatically performed in order to track the impact of source code changes. Since software is often modularized, the approach of Horký et al. [9] takes advantage of this paradigm and proposes to enhance software libraries by adding performance information to them.

The work of Danciu et al. [7] forces a model-driven approach instead. Here, a performance model is automatically derived from source code and enhanced with annotations from the developers. Based on a performance model, the approach is able to provide response time estimations for Java EE applications.

Regarding SAP Enterprise Applications, it is conceivable to develop a model-driven performance engineering approach following the example of Danciu et al. [7]. The model-driven approach is the most promising one considering both already existing performance values from a deployed SAP system and different hardware specifications of other systems of the entire landscape at the same time, e.g. of a quality assurance system. Following this strategy, it is possible to estimate response times from performance simulations paying attention to the SAP system environment where the developed SAP Enterprise Application is intended to be finally deployed.

7 CONCLUSION

Software performance improvement is relevant in order to address long-running and bad performing applications. According to a study of Compuware and PAC [6], 43% of the interviewed companies quoted being dissatisfied with the performance of their SAP systems. Moreover, it is well known that in modern development processes performance tests are conducted in late development phases after the main functionality is implemented. This phenomenon has not changed over the years, although we are aware of increasingly high costs for optimizations of late software changes.

This paper outlines barriers for performance improvement in early stages of the software development life cycle of SAP Enterprise Applications. With open-structured expert interviews we identified seven barriers that hinder performance improvement during the development phase of a software product. The key findings show that most barriers are based on missing performance skills of software developers, different technical characteristics of development and production system, and bad communication between different departments of a company.

While applying the concept of Performance Awareness [16], software developers get insights on the performance of applications they are currently developing. Following this strategy and encouraging both software developers and IT administrators to work tighter with people from the Specialist Departments together can help create more efficient SAP Enterprise Applications on lower costs. The work of integrating Performance Awareness into the SAP Development Environment is a promising task and needs to be considered for future research.

REFERENCES

- [1] Martin Bayer. 2016. Kampf der ERP-Titanen. (Feb. 2016). Retrieved October 17th, 2017 from <https://www.computerwoche.de/a/kampf-der-erp-titanen>, 3223108
- [2] Alexander Bogner, Beate Littig, and Wolfgang Menz. 2009. *Interviewing Experts*. Palgrave Macmillan, London, UK.
- [3] Andreas Brunnert, André van Hoorn, Felix Willnecker, Alexandru Danciu, Wilhelm Hasselbring, Christoph Heger, Nikolas Roman Herbst, Pooyan Jamshidi, Reiner Jung, Jóakim von Kistowski, Anne Koziol, Johannes Kroß, Simon Spinner, Christian Vögele, Jürgen Walter, and Alexander Wert. 2015. *Performance-oriented DevOps: A Research Agenda*. Technical Report. <http://arxiv.org/abs/1508.04752>
- [4] Andreas Brunnert, Christian Vögele, Alexandru Danciu, Matthias Pfaff, Manuel Mayer, and Helmut Krcmar. 2014. Performance Management Work. *Business & Information Systems Engineering* 6, 3 (Jun 2014), 177–179. <https://doi.org/10.1007/s12599-014-0323-7>
- [5] Tomáš Bureš, Vojtěch Horký, Michal Kit, Lukáš Marek, and Petr Tůma. 2014. *Towards Performance-Aware Engineering of Autonomic Component Ensembles*. Springer Berlin Heidelberg, Berlin, Heidelberg, 131–146. https://doi.org/10.1007/978-3-662-45234-9_10
- [6] Compuware Corporation and Pierre Audoin Consultants. 2010. *SAP Performance Management - A Trend Study by Compuware and PAC*. Technical Report.
- [7] Alexandru Danciu, Alexander Chrusciel, Andreas Brunnert, and Helmut Krcmar. 2015. *Performance Awareness in Java EE Development Environments*. Lecture Notes in Computer Science, Vol. 9272. Springer International Publishing, Chapter 10, 146–160. https://doi.org/10.1007/978-3-319-23267-6_10
- [8] Jochen Gläser and Grit Laudel. 2010. *Experteninterviews und qualitative Inhaltsanalyse*. Springer-Verlag, Wiesbaden, Germany.
- [9] Vojtěch Horký, Peter Libiř, Lukáš Marek, Antonin Steinhauser, and Petr Tůma. 2015. Utilizing Performance Unit Tests To Increase Performance Awareness. In *Proceedings of the 6th ACM/SPEC International Conference on Performance Engineering (ICPE '15)*. ACM, New York, NY, USA, 289–300. <https://doi.org/10.1145/2668930.2688051>
- [10] Heiko Koziol. 2010. Performance evaluation of component-based software systems: A survey. *Performance Evaluation* 67, 8 (2010), 634–658. <https://doi.org/10.1016/j.peva.2009.07.007>
- [11] Karl-Heinz Kühnhauser and Thorsten Franz. 2015. *Einstieg in ABAP* (4., aktualisierte und erw. Aufl. ed.). Rheinwerk, Bonn, Germany. 575 pages.
- [12] Kevin P. McCormack and William C. Johnson. 2001. *Business Process Orientation: Gaining the E-Business Competitive Advantage*. CRC Press.
- [13] Daniel A. Menascé. 2002. Load Testing, Benchmarking, and Application Performance Management for the Web. In *Int. CMG Conference*. 271–282.
- [14] V. Y. Shen, Yu Tze-jie, S. M. Thebaut, and L. R. Paulsen. 1985. Identifying Error-Prone Software: An Empirical Study. *IEEE Transactions on Software Engineering* SE-11, 4 (April 1985), 317–324. <https://doi.org/10.1109/TSE.1985.232222>
- [15] Panorama Consulting Solutions. 2017. *2017 Top 10 ERP Systems Rankings Report*. Technical Report.
- [16] Petr Tůma. 2014. Performance Awareness: Keynote Abstract. In *Proceedings of the 5th ACM/SPEC International Conference on Performance Engineering (ICPE '14)*. ACM, New York, NY, USA, 135–136. <https://doi.org/10.1145/2568088.2576097>
- [17] Christian Weiss, Dennis Westermann, Christoph Heger, and Martin Moser. 2013. Systematic Performance Evaluation Based on Tailored Benchmark Applications. In *Proceedings of the 4th ACM/SPEC International Conference on Performance Engineering (ICPE '13)*. ACM, New York, NY, USA, 411–420. <https://doi.org/10.1145/2479871.2479934>
- [18] M. Woodside, G. Franks, and D. C. Petriu. 2007. The Future of Software Performance Engineering. In *Future of Software Engineering, 2007. FOSE '07*. 171–187. <https://doi.org/10.1109/FOSE.2007.32>