PERFORMANCE OVERHEAD OF ERP SYSTEMS IN PARAVIRTUALIZED ENVIRONMENTS

André Bögelsack
Department of Informatics, Technische Universitaet Muenchen
Boltzmannstraße 2, 85748 Garching, Germany

Keywords: Performance, ERP system, Virtualization, Zachmann test.

Abstract: Virtualization is a big trend in current IT world and is used intensively in today's computing centres. But little is known about what happens to the performance of computer systems when running in virtual environments. This work focuses on the performance aspect especially in the field of Enterprise Resource Planning systems (ERP). Therefore, this work utilizes a quantitative approach by using laboratory experiments to measure the performance differences between a virtualized and non-virtualized ERP system. First, on basis of a literature review a performance measurement framework will be developed to provide a comprehensive guideline how to measure the performance of an ERP system in a virtualized environment. Second, the performance measurement focuses on the overhead in CPU, memory and I/O intensive situations. Third, the focus lays on a root cause analysis. Gained results will be analyzed and interpreted to give recommendations for further development of both ERP system and virtualization solution. The outcome may be useful for further computing centre design when introducing new ERP systems and service delivery concepts like Software as a Service (SaaS) in a virtualized or non-virtualized environment.

1 INTRODUCTION

Virtualization's history starts in 1964 with the research project called CP-40 from IBM (Donovan and Madnick 1975). Goal of this project was the development of the so called virtual machine/virtual memory time-sharing operating system. Research output from CP-40 was utilized in the IBM System/360-67 (Donovan and Madnick 1975). This mainframe was equipped with an operating system which was able to share the resource for several users and run several instances of an operating system in so called virtual machines. Later in 1974 Popek and Goldberg (Popek and Goldberg, 1974) defined the formal requirements for virtualization, which are valid until today and form the main guidelines for all developers of virtualization solutions. During the 1970's till the 1990's virtualization was well established in the world of mainframes, but not in the world of conventional x86 personal computers. This is because of the x86 architecture was not built to support virtualization and does not meet the requirements for virtualization (Robin and Irvine, 2000). A first virtualization solution for x86 world was launched by VMware in 1999 (Devine et al., 1998). This solution overcomes the gap between the x86 architecture and the virtualization idea by bringing in a technique called trapping. Trapping is very time and CPU consuming because every executed instruction in the virtual machines has to be scanned during runtime (LeVasseur et al., 2006). Utilizing trapping in a virtualization solution assigns a virtualization solution to the type of "full-virtualization".

In 2001 a project called Denali was launched, which followed the idea of lightweight virtual machines (Whitaker et al., 2002). They developed a kind of virtualization solution for running specially adopted, slim operating systems in their virtual machines. Their main goal was the adoption of an isolation kernel for performance, scale and simplicity. This is often cited as the beginning of "paravirtualization" which may be described as second type of virtualization (Nakajima and Mallick, 2007). In 2003 a development project called Xen from the university of Cambridge used another method to run virtual machine on x86 hardware (Barham et al., 2003). They moved from the trapping idea to the idea of modifying guest operating systems to be aware of an underlying virtual machine monitor. This leads to a performance improvement compared to full-

Bögelsack A. (2009).
PERFORMANCE OVERHEAD OF ERP SYSTEMS IN PARAVIRTUALIZED ENVIRONMENTS.
In *Proceedings of the 11th International Conference on Enterprise Information Systems*, pages 201-207
DOI: 10.5220/0002194102010207

virtualization but has the disadvantage that every guest OS has to be modified before running on top of Xen. The main motivation for Xen was to improve performance of virtualization.

When dealing with the performance of virtualization solutions there are three main aspects to pay attention to: CPU, I/O and memory (Huang et al., 2006). Every guest OS operation accessing CPU, I/O or memory produces an overhead because of the work of the hypervisor. The hypervisor's task is to intercept these accesses. Depending on the type of virtualization this may produce more (full-virtualization) or less overhead (paravirtualization). A lot of research work has been done for the Xen hypervisor dealing with overhead measurements for file accesses, network accesses or CPU accesses (see Related Work). But to our knowledge there are no performance overhead measurements dealing with the performance of ERP systems when running in virtual machines. In (Casazza et al., 2006) the need for a new kind of virtualization benchmark is discussed because existing benchmarks may not be adapted to virtual environments. The main argument is that isolated benchmarks may not be suitable to cover all performance topics in virtual environments. This work follows the idea and argues that ERP systems show different performance behaviour than isolated benchmarks show. As ERP systems build the backbone of today's most important core business processes, nobody knows what happens when running an ERP system in a virtualized environment. There are no scientific publications available. So, this research work focuses on the performance overhead for ERP systems running in paravirtualized environments.

2 RESEARCH GOALS

To evaluate the impact of paravirtualization on the performance of ERP system, this work follows the central hypothesis: paravirtualization influences ERP system's performance negatively. Such hypothesis may be derived from former work in the area of performance overhead research which attested e.g. a performance loss up to 50% for I/O processing in a virtualized environment (Cherkasova and Gardner, 2005). Adopting these results to the area of ERP systems gives a starting point for this work as ERP systems are I/O intensive and may experience performance degradation too.

This very general hypothesis is elaborated into three main research questions, which are oriented after Barham (Barham et al., 2003), Huang (Huang et al., 2006), Svobodova (Svobodova, 1976) and Jain (Jain, 1991).

1. How can the ERP system's performance be measured in a paravirtualized environment and which performance measures are important?

The purpose of dealing with the question of how to measure performance in this setting is to find out how the ERP system can be stressed so that its operations produce a massive workload on the hypervisor and how the operations of the ERP system as well as the hypervisor can be monitored. The work focuses on paravirtualized environments as such environments are known for a better performance than full-virtualized environments (Nakajima and Mallick, 2007). By analyzing available literature, appropriate benchmark tools will be discovered which are suitable for the ERP system and are able to produce massive workload on the ERP system. The range of benchmark tools will be limited by the type of workload the tools provide and to which ERP system the benchmark tools can be applied to. This research question does also cover the discovery of appropriate performance measures whereas performance measures can be understood as "a proper set of parameters upon which the evaluation will be based" (Svobodova, 1976). The set of performance measures is limited by the benchmark tool as well as the ERP system itself. The choice which performance measures are necessary or not is oriented after the three overhead aspects CPU, memory and I/O (Huang et al., 2006). After determining the needed performance measures the work focuses on the available performance monitors. Regardless if these monitors are hardware or software monitors they must be able to return the necessary numerical performance data. The number of performance monitors will be limited by the used hypervisor, ERP system, benchmark tool, performance measures and operating system. Finally, within the scope of this research question we research possible testing environments (breadboard construction) which include test hardware, test software, possible workloads and performance measures.

By utilizing a literature review for answering the first research question the outcome will include possible benchmarks for the ERP system, performance measures and an appropriate testing environment.

2. Does a hypervisor significantly influence CPU, memory and I/O operations of an ERP system working in a paravirtualized environment in high workload situations?

Stressing the ERP system with distinct CPU, memory and I/O intensive workloads we assume that the hypervisor will be stressed too because of the previously mentioned interceptions. Although there are some publications available dealing with CPU and memory overhead we believe due to the architecture of the ERP system (with a rather large number of concurrently active processes) we will get different results. The work focuses on high workload situations as these situations influence the subjective user experience when working with an ERP system e.g. due higher response times. To quantify the assumed performance degradation we will run several test runs inside the paravirtualized environment and compare them to results gathered in a native environment. Modifying the configuration of the hypervisor (e.g. scheduler) and ERP system allows us to determine which configuration suits best the interaction of the ERP system and the hypervisor. The hypervisor's impact on performance can be derived already from the architecture and previous work (Barham et al., 2003) but the answer should be deduced in an empirical way to be able to quantify the impact.

Answering the second research question enables us to highlight the performance impact of paravirtualization on massive CPU, memory and I/O loads when running an ERP system.

3. What causes the performance overhead of the ERP system in a paravirtualized environment?

The last research question focuses on the root cause analysis of performance degradation. On basis of extensive monitoring inside the hypervisor, the ERP system and the operating system during the test runs we aim on statements what causes the overhead (e.g. swapping problems, scheduler problems, lock problem inside ERP system etc). Analysis's findings may also be of interest for other hypervisors. Cherkasova (Cherkasova et al., 2007) tested several configurations for the internal scheduler in Xen and pointed out performance differences due to different scheduler configurations. By utilizing monitors to gather data and by changing the configuration during the test runs we can analyze system behavior in detail.

Researching the aforementioned topics the work as a whole highlights the performance impact of paravirtualization on an ERP system when running massive CPU, memory and I/O operations. In addition it allows for recommendation for choosing the best ERP system configuration and gives insights into the reason for performance degradation.

3 LIMITATIONS

This work is limited in its scope by two factors. First, the work is limited to a specific virtualization solution. Second, this work chooses a specific ERP system.

The current market for virtualization is very large: several different virtualization solutions form a heterogeneous market (Jehle et al., 2008). On top of the heterogeneity there are several virtualization techniques and kinds of virtualization available (Nakajima and Mallick, 2007). Because of the better performance of paravirtualization this work chooses paravirtualization as a field of focus for the performance tests. Currently the main research focus lays on Xen, which is a typical paravirtualization solution. As Xen is open-source, several vendors began to develop their own implementation of this hypervisor. In this work one of these implementations from Sun Microsystems™ is used.

Another limitation is the ERP system. The current market in the field of ERP system is dominated by SAP. Most of the Fortune 500 companies use SAP software to support their core business processes. As SAP is the market leader and this work is positioned in the SAP University Competence Centre, the used ERP system for evaluation will be SAP ERP 2005. A slightly modified edition of this ERP system is available as the so called SAP Linux Certification Suite (SLCS) (Kühnemund, 2007). It is used internally by SAP to test and benchmark new hardware platforms for the first time. As the server hardware, which will be used for the performance evaluation in this work, is provided by Sun Microsystem™, a specially ported version for this hardware platform will be used.

4 RELATED WORK

The current research focus lies on paravirtualization. There are several works available dealing with the performance impact of paravirtualization in different ways and in different fields of applications ((Huang et al., 2006), (Cherkasova and Gardner, 2005) , (Mennon et al., 2005), (Zhang and Dong, 2008), (Youseff et al., 2006)). For the performance of ERP system there are many works available dealing with the user's performance in ERP system but only a few works focus on the performance of the ERP system itself ((Kemper et al., 1999), (Wilhelm, 2003), (Zeller and Kemper, 2002)). Therefore the current research status is divided into two subsets: a first subset describes related work in the area of performance measurement in paravirtualized environments. A second subset explains the current work in the area of ERP system's performance.

4.1 Performance Measurements in Paravirtualized Environments

The first work to be pointed out is the work of Barham et al. (Barham et al., 2003), who developed Xen, a hypervisor operating with a new concept for virtualization. To testify their newly developed hypervisor they ran several test runs dealing with some typical fields of applications: SPEC INT2000 benchmark, building a Linux kernel, Open Source Database Benchmark suite, file system benchmark and a SPEC Web99 benchmark. In addition Barham ran several smaller operating system benchmarks for examining the performance overhead in particular subsystems. Several other test runs focus on the performance isolation capability of Xen. The result of the work is a small performance loss in every test run compared to a native environment. As virtualization represents an additional software layer this overhead is intuitively understandable.

In 2005 Cherkasova et al. (Cherkasova and Gardner, 2005) measured the CPU overhead for I/O processing in Xen. To do so they utilized a SPECweb'96 and SPECweb'99 benchmark to stress the CPU with massive I/O and network traffic. They discovered that heavy I/O and network traffic can produce a huge CPU overhead in the control domain of Xen. In 2007 Cherkasova made suggestions how to improve the performance by applying CPU schedulers (Cherkasova et al., 2007).

Huang performed a NAS parallel benchmark to get a first impression of the overhead which is produced by using virtualization (Huang et al., 2006). The results showed performance degradation of up to 12% - 17%. Actually they looked for a new bypass I/O which allows virtual machines to access I/O devices directly.

Mennon focused on the performance overhead for network traffic in 2005 (Mennon et al., 2005). Therefore, a tool named Httperf was used and they point out a performance overhead up to 25% depending on the network traffic actions.

Zhang and Dong investigated the performance overhead when using the new hardware-supported virtualization technique from Intel, called Intel VT (Zhang and Dong, 2008). The paper states that a performance overhead between 5% - 20% can be experienced. To gain these results they utilized several benchmarks like Kernel build, SPEC Int or SPEC JBB.

Youseff et al. compared several HPC (high performance computing) kernels to the Xen kernel (Youseff et al., 2006). The authors have done a performance evaluation by using several different benchmarks for e.g. disk performance, communication or memory performance. Their publication shows a performance loss up to 30% (high workload) depending on the field of application.

In the area of HPC a lot of work was done regarding HPC application performance in virtualized environments ((Ranadive et al., 2008), (Farber, 2006), (Liu et al., 2006)). They do not deal with the overhead issue but the overall performance.

As an essence one can derive that there is a lot of work available regarding the measurement of performance and performance overhead when using Xen. Each work uses well established benchmarks. But there is no work available which deals with the performance overhead when running an ERP system in a virtualized environment.

4.2 Performance Measurements of ERP systems

Compared to the available publications for the performance of virtualized environment, current work for the measurement of an ERP system's performance are rare. This work chooses SAP ERP 2005 as the market leader to be an instance of such ERP software systems. For SAP ERP 2005 there are three significant works.

Kemper discussed the possibility of tuning the performance for SAP R/3 systems (Kemper et al., 1999). They focused on the main memory management and the database. For the database they mentioned several available benchmarks, like TPC-D. In another work Zeller and Kemper (Zeller and Kemper, 2002) showed how to increase the performance for the database in case of special database operations. They especially paid attention to the main architecture of SAP R/3 by using a special developed testing tool called SSQJ but mainly focus on the database and not on the entire ERP system.

In his PhD dissertation Kai Wilhelm discusses storage systems serving a SAP R/3 system (Wilhelm, 2003). The focus lies on characterization of the storage load during benchmarks, the performance measurement as well as the recommendation for the 'right' storage system in a future SAP system landscape.

In summary, former research work focussed on the database and the underlying storage system whereas an improvement of the overall performance of the SAP R/3 system was achieved by very special improvements in one of the components of the SAP R/3 system. Keeping this in mind and connecting this with the research which was done in the field of virtualization there is a need for an evaluation of the performance of SAP R/3 systems in virtualized environments.

5 RESEARCH STRUCTURE

The research structure in this work is influenced by quantitative research design and induction (Bortz and Döring, 2005). The research strategy in this work will be in accordance to the work of (Svobodova, 1976) and (Jain, 1991). In Svobodova's book about computer performance measurement she introduced three main steps for performance measurement:

1. Define performance measures.

2. Determine the quantitative value of performance measures and analyze system performance with respect to system structure and system workload.

3. Assign qualitative values to different levels of performance measures and assess systems performance.

Jain independently developed a more detailed view on the performance measurement process and proposed ten steps. Jain's ten steps together with Svobodova's three steps form a comprehensive performance measurement process. Mapping these steps onto the described research questions draws the research process for this work which is oriented after Jain's ten steps and goes into more detail with Svobodova's steps where it is necessary and supportive.

1. The *goal* of this performance measurement work is to find out if ERP system's performance differs in virtualized environments compared to non-virtualized environments. This work focus on ERP system's performance as little is known in this field of application. Paravirtualization is chosen from the types of virtualization as the most research work is done in the filed of paravirtualization. The *system* will consist of an exemplary ERP system from SAP, which is running on hardware from Sun Microsystem.

2. *System services* in an ERP system are manifold. This work focus on the ability of the ERP system to allocate memory and work in the main memory of the underlying hardware/operating system. This ability is needed by the system to process user requests. The faster system works in the memory the faster it can respond to user requests. Another *system service* is the ability to store data persistently in a database. Therefore the system utilizes the underlying hardware/operating system, too. Paravirtualization may influence the ability and the performance of the ERP system. Hence, we need an appropriate workload which is able to stress the ERP system with such memory and I/O operations. Stressing the ERP system with high memory load will automatically stress the CPU, too. A possible *outcome* may be a number of the highest value for memory and I/O operations of the ERP system.

3. The *performance metrics* are the criteria used to compare the performance and they are derived from the system, the workload and the architecture which was chosen to be measured. In this research work one can think of simple metric for the memory operations as well as a simple metric for the I/O operations, e.g. throughput. At this stage some thoughts of Svobodova can be added as she mentioned the so called performance measures. The term "performance measures" can be understood as a set of parameters upon which the evaluation will be based. Keeping this in mind and remembering the first research question in this work, the first step in the research process is to determine the appropriate performance parameters. This includes a literature review to find out which performance parameters are suitable and how they can be measured. Of course performance parameters depend on the system architecture. The literature review is oriented after the system architecture and covers the main components as well as their performance parameters.

4. Fourth step is about listing all *system and workload parameters*, which affect the performance. System parameters do not change during the measurements and workload parameters change during the measurement. This means parameters like the basic information about hardware, configuration of operating system and configuration of the ERP system itself will be kept constant. A workload

parameter is the usage of a virtualization solution. After a first test run the list of parameters may be extended.

5. During the test runs a lot of *factors* are to be studied. Factors are parameters that change during a test run. Such factors may be slight changes to the system's settings but also bigger changes like the usage of a virtualization solution (or not). This work will start with a short list of parameters, which then will be extended if necessary. Parameter's list depend on the determined performance metrics (see step 3) as well as system outcomes (see step 2).

6. Selecting the *evaluation method* involves the measurement of the system. Besides simulation and analytical modelling, measuring is one of the possible evaluation methods. This work uses measurement because the measurement subject is a system which already exists.

7. Determination of the correct system *workload* is an outcome of research question 1. The most important step is to "characterize the workload by distributions of demands made on individual system resources" as well as to "define a unit of work and express the workload as a number of such units" (Svobodova, 1976) For the measurement this work utilizes as synthetic workload, which can be described as an "artificial reproducible workload". Task of the workload is to produce heavy memory and CPU load as well as heavy I/O load. Several workloads are available for SAP systems (SAP Benchmarks.,01/20/2009). Selection of a suitable workload is made by answering research question 1.

8. *Designing* the experiment means to establish the laboratory environment as well as the breadboard construction. After establishing the environment several test runs will be run. That will lead to the determination of the quantitative value and allow for analyzing the systems' performance. This step is the core one in the work as it involves several measurements of the ERP system in non-virtualized environment and paravirtualized environment. For estimating the performance this work uses a combination of measurements and analysis. For the analysis of the system's performance several test runs must be run to gain the necessary data. Barham (Barham et al., 2003) used seven test runs for each test. This work will follow this idea but will add two more test runs as a warm-up phase for the ERP system. This is important because the ERP system has to fill its internal buffers to be fully operational. Filling the buffers in a first test run may falsify the test results. For recording the performance parameters it is necessary to utilize performance monitors. Performance monitor are "tools that facilitate analytic measurements" (Svobodova, 1976) and "observe the activities on the system" (Jain, 1991). Generally there are two types of a monitor: hardware and software monitors (Leung,1988).

9. *Analysing and interpreting* the data is described as translating the quantitative values into qualitative values. This means to find some scale of 'goodness' for the relative measure of non-virtualized ERP system and paravirtualized ERP system. The results of this step are qualitative statements about the performance difference of ERP systems running in a –non-virtualized environment compared to paravirtualized environment.

10. The last step is about presenting the results which means to prove or reject the hypothesis.

The following table gives an overview about the mapping of the research question to this iterative process:

Table 1: Mapping Jain's steps onto research questions.

Research Question	Step (after (Jain, 1991))
Motivation	Goals and system boundaries
1	System services and outcomes
	Performance metrics
	System and workload parameters
	Performance factors
	Evaluation techniques
	Workload selection
	Experiments design
2 & 3	Analyze and interpret data
	Present results

To outline the research strategy in few words: this work is hypothesis-driven (as the work assumes a performance overhead caused by paravirtualization), it uses laboratory experiments for performance measurement to determine quantitative values. The work utilizes a synthetic workload and uses quantitative analysis for interpreting the gained results.

6 PRACTICAL IMPACT

The most significant outcome will be the performance results. As there are no scientific publications about the measurement of paravirtualized ERP systems, it is an addition to research in this field to have an evaluation of how these complex software systems behave in virtualized environments. This may have an impact on the future computer centre design when new ERP systems are introduced to the computing centre (with or without virtualization).

Another outcome of this work will include recommendations for the further development of hypervisors and ERP systems. As the work will analyse the virtualized ERP system's performance it can state out possible bottlenecks and may suggest changes to the design of hypervisor/ERP system as well as the interaction between both components.

7 TIMELINE

Table 2: Timeline.

Duration / Milestone	**Activity**
01/2009 –03/2009	Literature review for RQ 1
03/2009 –12/2009	Performance measurements and test runs for RQ 2
12/2009 –06/2010	Root cause analysis for RQ 3
07/31/2010	Dissertation version 1.0

REFERENCES

Donovan, J. J., Madnick, S. E., Hierarchical approach to computer system integrity. In: *IBM Systems Journal,* Volume 14, No. 2, Pages 188-202 (1975)

Popek, G., J., Goldberg, R., P., Formal Requirements for Virtualizable third Generation Architectures. In: *Communications of the ACM*, Volume 17, Number 7, Pages 412-421 (1974)

Robin, J., S., Irvine, C., E., Analysis of the Intel Pentium's Ability to Support a Secure Virtual Machine Monitor. In: SSYM'00: Proceedings of the 9th conference on USENIX Security Symposium (2000)

VMWARE Devine, Scott W., Bugnion, Edouard, Rosenblum, Mendel, Virtualization system including a virtual machine monitor for a computer with a segmented architecture (1998)

LeVasseur, J., Uhlig, V., Chapman, M., Chubb, P., Leslie, B., Heiser, G., Pre-Virtualization: soft layering for virtual machines. Technical Report 2006-15, TH Karlsruhe (2006)

Whitaker, A., Shaw, M., Gribble, S. D., Scale and performance in the Denali isolation kernel. In: *Proceedings of the 5th symposium on Operating system design and implementation*, Pages 195-209 (2002)

Nakajima, J., Mallick, A. K., Hybrid-Virtualization – Enhanced Virtualization for Linux. In: Linux Symposium, Volume 2, Pages 87-96 (2007)

Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I., Warfield, A., Xen and the Art of Virtualization. In: *SOSP '03 - Proceeding of the nineteenth ACM symposium on Operating systems principles*, Pages 164-177, (2003)

Huang, W., Jiuxing, L., Abali, B., Dhabaleswar, K. P., A Case for High Performance Computing with Virtual Machines. In: *ICS '06: Proceedings of the 20th annual international conference on Supercomputing*, Pages 125-134 (2006)

Casazza, J. P., Greenfiled, M., Shi, K., Redefining Server Performance Characterization for Virtualization Benchmarking. In: *Intel Technology Journal*, Volume 10, Issue 3, Pages 243-251 (2006)

Cherkasova, L., Gardner, R., Measuring CPU Overhead for I/O Processing in the Xen Virtual Machine Monitor. In: *Proceeding of the annual conference on USENIX Annual Technical Conference*, Pages: 387-390 (2005)

Svobodova, L., Computer Performance Measurement and Evaluation Methods: Analysis and Applications. American Elsevier Publishing Company, Inc., New York (1976)

Jain, R., The art of computer systems performance analysis: techniques for experimental design, measurement, simulation, and modelling. John Wiley & sons, Inc., Littleton, Massachusetts (1991)

Cherkasova, L. Gupta, D., Vahdat, A., Comparison of the Three CPU Schedulers in Xen. In: *ACM SIGMETRICS Performance Evaluation Review*, Volume 35, Issue 2, Pages 42-51 (2007)

Jehle, H., Wittges, H., Bögelsack, A., Krcmar, H.: Virtualisierungsarchitekturen für den Betrieb von Very Large Business Applications. In: *Proceedings of Multikonferenz Wirtschaftsinformatik*, Pages 1901-1912 (2008)

Kühnemund, H., Documentation for SLCS v2.3. SAP AG, Walldorf (2007)

Mennon, A., Santos, J. R., turner, Y., Janakiraman, G. J., Zwaenepoel, W, Diagnosing performance overheads in the Xen virtual machine environment. In: *Proceedings of the 1st ACM/USENIX international conference on Virtual execution environments*, Pages 13-23 (2005)

Zhang, X., Dong, Y., Optimizing Xen VMM Based on Intel® Virtualization Technology. In: *Proceedings of the 2008 International Conference on Internet Computing in Science and Engineering*, Pages 267-274 (2008)

Youseff L., Wolski R., Gorda B., Krintz C., Evaluating the Performance Impact of Xen on MPI and Process Execution For HPC Systems. In: *Proceedings of*

Second International Workshop on Virtualization Technology in Distributed Computing (2006)

Kemper, A., Kossmann, D., Zeller, B., Performance Tuning for SAP R/3. In: *Data Engineering*, Volume 22, Number 2, Pages 32-39 (1999)

Wilhelm, K., Messung und Modellierung von SAP R/3- und Storage-Systemen für die Kapazitätsplanung (2003)

Zeller, B., Kemper, A., Exploiting Advanced Database Optimization Features for Large-Scale SAP R/3 Installations. In: *Proceedings of the 28th VLDB Conference* (2002)

Ranadive, A., Kesavan, M., Gavrilovska, A., Schwan, K., Performance Implications of Virtualizing Multicore Cluster Machines. In: *Proceedings of the 2nd Workshop on System-level Virtualization for High Performance Computing* (2008)

Farber, R., Keeping "Performance" in HPC: A look at the impact of virtualization and many-core processors. In: *Scientific Computing* (2006)

Liu, J., Huang, W., Abali, B., Panda, D., K., High Performance VMM-Bypass I/O in Virtual Machines. In: *ATC* (2006)

Bortz, J. and N. Döring (2005): Forschungsmethoden und Evaluation für Human- und Sozialwissenschaftler. 3, Springer, Heidelberg 2005.

SAP Benchmarks. http://www.sap.com/benchmarks, accessed on 01/20/2009

Leung, C. H. C., Quantitative Analysis of Computer Systems. John Wiley & Sons, Inc., London (1988)