

Performance Impact of Type-I Virtualization on a NewSQL Relational Database Management System

J. Bryan Osborne

Computing & Mathematics
Oral Roberts University, Tulsa, OK, USA

Abstract—For more than 40 years, the relational database management system (RDBMS) and the atomicity, consistency, isolation, durability (ACID) transaction guarantees provided through its use have been the standard for data storage. The advent of Big Data created a need for new storage approaches that led to NoSQL technologies, which rely on basic availability, soft-state, eventual consistency (BASE) transactions. Over the last decade, NewSQL RDBMS technology has emerged, providing the benefits of RDBMS ACID transaction guarantees and the performance and scalability of NoSQL databases. The reliance on virtualization in IT has continued to grow, but an investigation of current academic literature identified a void regarding the performance impact of virtualization of NewSQL databases. To help address the lack of research in this area, a quantitative experimental study was designed and carried out to answer the central research question, "What is the performance impact of Type-I virtualization on a NewSQL RDBMS?" VMware ESXi virtualization software, NuoDB RDBMS, and OLTP-Bench software were used to execute a mixed-load benchmark. Performance metrics were collected comparing bare metal and virtualized environments, and the data analyzed statistically to evaluate five hypotheses related to CPU utilization, memory utilization, disk and network input-output (I/O) rates, and database transactions per second. Findings indicated a negative performance impact on CPU and memory utilization, as well as network I/O rates. Performance improvements were noted in disk I/O rates and database transactions-per-second.

Keywords—Database benchmarking; NewSQL; relational database; virtualization

I. INTRODUCTION

Drastically changing paradigms of data management and storage, Big Data continues to be a disruptive technology in the world of computing [1]. To provide acceptable performance, databases required new architectures and a built-from-scratch approach to overcome performance limitations inherent to traditional relational database management systems [RDBMSs; 2]. The newest of these databases are referred to as NewSQL and have been gaining traction with their ability to support massively large datasets, provide atomicity, consistency, isolation, durability (ACID) transaction guarantees, support for structured query language (SQL) queries, and do so with the performance provided by NoSQL solutions [3].

The use of virtualization in the modern era provides information technology (IT) management the ability to more efficiently manage resources through improved utilization rates and allows for greater flexibility of existing equipment and increased scalability of physical servers [4]. Pogarcic, Krnjak and Ozanic [5] demonstrated that virtualization could provide

decreased capital and operation costs for businesses in areas of actual and procurement costs for server hardware, utility, and administration costs. Virtualization technology provides the foundation for cloud computing, which continues to shape the world of modern IT [6]. Cloud computing providers regularly utilize virtualization's flexibility and scalability to maximize revenue and meet the increasing demands for their services [7]. Cloud computing relies heavily on virtualization to provide server and database services to customers [8].

NewSQL databases have demonstrated superiority in performance testing against NoSQL databases involving Internet of Things (IoT) applications [9] and as a solution for Big Data OLTP applications [10]. Performance comparisons of NewSQL offerings appear in the literature but are limited to comparisons of NewSQL databases against each other in bare metal [9, 11], virtualized [12], and cloud [13] environments. There is a lack of published research on the performance implications of the virtualization of NewSQL systems. The continued increase in the use of virtualization technology in data centers and the increase in the implementation of NewSQL database systems in cloud computing indicated a need for this research effort.

The strong presence of virtualization as a technology in IT services including the cloud [4], the continued growth in the use of NewSQL databases [3], and the simultaneous use of both, presented a need to understand the impact of one on the other. A review of extant literature indicated a void with respect to the examination and quantification of the impact of virtualization on NewSQL RDBMS. Therefore, the research effort posed a central question of "What is the performance impact of Type-I virtualization on a NewSQL RDBMS?" NewSQL databases are relational by nature and therefore the measures related to RDBMS are relevant, as is the impact of virtualization on relevant system metrics. The use of throughput of RDBMS software, measured by transactions-per-second (TPS) as suggested by Bitton, et al. [14], remains a prevalent and accepted metric for performance along with the use of benchmarking software to perform testing [15]. Therefore, performance measures of system CPU, memory, disk and network I/O, and NewSQL database throughput were utilized as the dependent variables in this research. The impact on these dependent variables due to changes in the independent variable, stated as the condition of the system being bare metal or Type-I virtualized, led to the central question being operationalized into five null (and corresponding alternative) hypotheses, one for each of the five performance measures to be tested. The research quantified the level of impact virtualization caused to a system running a NewSQL database

via an experimental design constructed to measure and analyze variables relevant to system performance in both bare metal and virtualized environments. The investigation contributes to the body of knowledge by filling a void in the academic literature regarding the performance impact of virtualization of NewSQL databases. Further, it provides information relevant to potentially needed modifications to a system to more efficiently host a NewSQL RDBMS.

The rest of the paper is organized as follows. Section II briefly discusses virtualization, NewSQL database management systems, and research relevant to the project. Section III describes the experimental methodology, hardware and software configuration, benchmarking software used in the test environment and statistical methods utilized. Section IV contains the results of the statistical analysis of the data collected and a discussion of the results of the analysis. Finally, Section V presents conclusions reached.

II. BACKGROUND AND RELATED WORK

A. Virtualization

Type I hypervisors interface directly with the computer's underlying hardware on which they run and map the physical resources of a computer to the VM being hosted, as shown in Fig. 1. This architectural approach has led to these types of hypervisors being referred to as native [16] or bare metal [17] hypervisors as they execute directly on the host computer and serve as a link between the virtual machines and the host computer.

The goal of virtualization is to use a layer of software to abstract the specific hardware of a computing machine from operating systems executing on the machine [18]. Thus, virtualization creates both an abstraction and encapsulation of the various components of the underlying hardware [19], and it is the role of the hypervisor to provide this abstraction as well as provide the virtual machine's integrity and isolation from other VMs on the same hardware [20]. Three major virtualization approaches, full virtualization, paravirtualization, and hardware-assisted virtualization, have been used to overcome the shortcomings of the x86 architecture [20].

The impact of virtualization caused by the insertion of a layer of processing between the guest OS and hardware has been demonstrated by extant research. In a review of 112 publications produced prior to 2016, Kao [21] found that the majority of evaluations of virtualized environments utilized benchmarks to measure performance. The specific benchmarks used by studies reviewed as part of this literature review mirrored those found by Kao [21] and reflected the emphasis found in the literature on using measurements of CPU, memory, disk, and network utilization as a basis for performance analysis. While it can be argued that differences exist between benchmarks and real-world workloads [22], benchmarking has a long history and provides for the gathering of metrics using a well-defined, universally accepted set of tests that can be compared across applications, operating systems, and hardware platforms [15]. Based on the review of all articles, the research focused on these metrics to assist in the evaluation of the impact of virtualization on NewSQL databases.

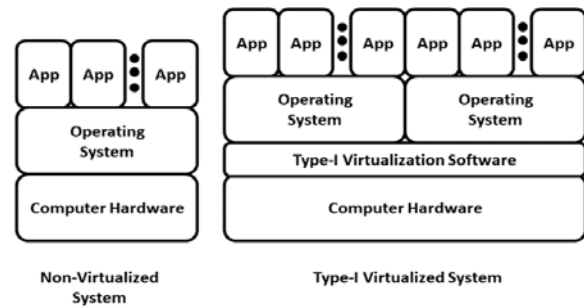


Fig. 1. Comparison of Non-virtualized and Type-1 Virtualized Systems.

B. NewSQL

The term "NewSQL" is generally attributed to a report issued by the 451 Group analysts Matthew Aslett, who is referring to what was at the time new vendor offerings of databases that supported SQL, ACID transactions, and the high performance and scalability of NoSQL databases [23]. A short time later, in a blog post on Communications of the ACM website, Michael Stonebraker argued that instead of continuing the "gold standard in enterprise computing" of multiple relational databases connected by extract-load-transform (ETL) processes, NewSQL database systems, with the characteristics listed above, address consistency issues and preserve SQL language capabilities [24]. In a blog posting, Stonebraker outlined five specific characteristics of NewSQL as having ACID transactional ability, concurrency control that was non-locking, high performance, possessed a distributed, horizontally scalable, shared-nothing architecture, and SQL as application language. Although these characteristics were not presented in the format of an academic paper, the qualifications listed are present in academic works discussing NewSQL systems [2, 10, 12, 25-27].

NewSQL systems can be categorized into three types, novel systems built from scratch utilizing a new architecture, middleware approaches, and Database-as-a-Service (DBaaS) offerings [2]. The research in this paper focuses on an example of the first category, which includes the characteristics outlined by Stonebraker [24] for NewSQL systems and is created through the writing of completely new code, thereby being free from architectural choices of existing systems [28]. The development of the system as a new application provides the ability to manage best the disk and memory storage, replication approaches, query optimization, and node communication, allowing better performance than systems built through the layering of existing technologies [2].

Research by Hrubaru and Fotache [29] evaluated the performance of RDBMS and NewSQL using the TPC-H benchmark, capturing and analyzing loading and query times, which are essentially throughput measures, as well as the amount of memory used by the DBMSs. Not surprisingly, the in-memory NewSQL instance used more memory than the two RDBMS but performed better in most cases in terms of loading and query times. Similarly, Oliveira and Bernardino [11] compared two NewSQL RDBMSs using TPC-H, comparing loading and query execution times, but an evaluation of memory utilization was not performed. In both of these

articles, as with Fatima and Wasnik [9], the experimental setup only used a single server to perform tests. Two studies involving performance measures of virtualized NewSQL environments were identified. Both focused on comparing RDBMS performances when running in virtual machines, but no evaluation was made comparing bare metal and virtual environments. Kaur and Sachdeva [12] compared throughput performance measures of four popular NewSQL databases but did not specify the exact tests performed, only providing latency times captured for read, write, and update operations and total execution time. Tests were performed using a single instance RDBMS running in a single Type II hypervisor. While relative performance measures provide some helpful information, it is limited due to the use of single-instance NewSQL servers and a Type II hypervisor, neither of which would be common in a larger scale production system. Borisenko and Badalyan [30] utilized a Type I hypervisor with individual cluster nodes running in separate VMs to evaluate two NewSQL RDBMSs. One of the NewSQL RDBMSs evaluated, Apache Ignite [31], would be considered by the definitions provided by Pavlo and Aslett [2] as a middleware approach. The other, VoltDB [32], is a new architecture, thereby providing a performance comparison between the two approaches. The workload utilized was described as "TPC-H like," and metrics gathered and analyzed consisted only of query execution times [30]. Although both investigations utilized virtualized environments, Borisenko and Badalyan [30] and Kaur and Sachdeva [12] treated the performance testing as if performed on bare metal and did not provide any insights into the performance impact of virtualization.

Given the relevance of cloud computing today and the importance of virtualization to cloud computing, the implementation of NewSQL databases in the cloud is very probable [25]. Previous research has identified comparative performance studies of NewSQL databases, but none that reflected the impact of Type-I virtualization on NewSQL databases. Further, the studies examined reflected the use of single-node systems, not reflecting the true distributed nature of NewSQL databases [2, 27, 28]. This paper fills the gap in the literature by providing a quantification of performance impacts to distributed NewSQL systems due to Type-I virtualization.

III. METHODOLOGY

The quantitative experimental research embodied in this study provides insight into the impact of Type-I virtualization on a NewSQL database, as well as a foundation for further exploration of potential mechanisms to best leverage virtualization in this new and important realm of computing. The research quantified the level of impact virtualization causes to a system running a NewSQL database via an experimental design constructed to measure and analyze variables relevant to system performance in both bare metal and virtualized environments. The central research question, "What is the performance impact of Type-I virtualization on NewSQL relational databases?" operationalized into five null hypotheses, each stating there was no difference in recorded values of the respective dependent variables during database benchmarking for a system hosting a NewSQL database

instance when comparing bare metal and Type-I virtualized environments.

The configuration of the system, bare metal or virtualized, reflects the independent variable in the experimental design. Metrics of CPU and memory utilization, disk and network I/O measurements, and the number of transactions-per-second executed by the databases represent dependent variables. These values were captured and recorded during benchmark tests executed on bare metal and virtualized systems built on identical servers. The values captured were then analyzed to quantify differences in performance between bare metal and virtualized systems.

The computer hardware used consisted of a Hewlett Packard Enterprise (HPE) Apollo r2600 chassis with 3 HPE dual-processor nodes. Each processor node was comprised of two Intel Xeon Broadwell E5-2698v4 2.4GHz CPUs, each with 14 cores and 50MB cache, 128GB of DDR4-2400 memory, and a single 480MB SATA solid-state drive (SSD). Internode communications are performed through 1 GbE Ethernet ports connected via an Omnipath 100Gb port. The computing nodes resided on an isolated network, ensuring only intended access to the machines.

The software configuration of each node varied based upon its functionality in the test. Each node ran the same operating system, CentOS 7.7.1908, an open source version of the Red Hat Enterprise Linux operating system [33]. Two of the three hardware nodes were designated as database cluster nodes using the recommended CentOS infrastructure server template, NuoDB NewSQL RDBMS, and performance metric collection software. The third node, configured to execute benchmark and load generation software, was configured as a developer's workstation to allow for the compiling of software with the GNU compiler.

In a recent study by Almassabi, Bawazeer [27], 13 of the top NewSQL databases were identified. From this group, NuoDB CE 4.0.4.2 [34] was selected for the current research effort as it meets the requirements defined by Stonebraker [24]. NuoDB is a distributed, peer-to-peer, ACID-compliant, elastic, and highly scalable relational database management system that falls into the "new architecture" class, elastic and highly scalable [35], and offers a fully functional, community edition version available at no cost.

NuoDB is designed to operate in bare metal, virtualized, and cloud environments [36]. The database management system has a two-tier, distributed architecture separating transactional and storage tiers. The transactional tier is an in-memory tier responsible for atomicity, consistency, and isolation aspects of the ACID transactional model and is designed to ensure fast access to data by applications. The storage management tier is responsible for the durability aspect, ensuring data is safely stored when committed, and providing data in case of a cache miss. Although all nodes are peers in a cluster, NuoDB nodes execute as either a Transaction Engine (TE) or a Storage Manager (SM). SMs maintain complete, consistent, independent copies of the entire database. TEs cache database tables in memory, accept database requests and execute SQL queries. For purposes of

this research, one of the two nodes functioned as an SM and the other database node as a TE. Installation of the NuoDB software was performed per the manufacturer's instructions [37]. Also installed on each of the database nodes was the performance metric gathering tool, "nmon" [38], a software tool written in the C programming language, demonstrated to be effective in capturing metrics in performance testing research [39, 40]. The nmon application captured data on database nodes for CPU utilization measured as a percent of total available, total system memory utilization measured in MB allocated, disk I/O measured in KB/s, and network I/O measured in KB/s, each of which represents a dependent variable in the experiment.

The third node ran the load-generating, benchmarking software, OLTP-Bench [41]. OLTP-Bench is an open-source, extensible, flexible benchmarking testbed capable of executing a number of existing benchmarks on both on-premise and cloud databases [42]. The Java source code for the OLTP-Bench software is open-source and available on GitHub and was downloaded and compiled for the CentOS platform on the benchmark node. The software was utilized to exercise the NewSQL database and system and provided the transactions-per-second measurement, a dependent variable, for all tests. Due to the different nature of online transaction processing (OLTP) and online analytical processing (OLAP), separate databases with different designs have typically been implemented [43]. A growing need for real-time analytics has generated a change in this paradigm, and new databases, like NewSQL, have been developed to provide support for these needs [44]. Subsequently, benchmarking tests to evaluate the performance of databases designed to handle mixed workloads are needed [45].

One such benchmark, currently supported by OLTP-Bench, is CH-benCHmark [46]. CH-benCHmark is a hybrid/mixed-workload benchmark designed to execute TPC-C (OLTP) and TPC-H-equivalent (OLAP) queries concurrently against a common set of database tables. The entities and relationships of the TPC-C model are implemented without modification, and only a slight modification to the TPC-H schema is made to ensure the integration into the TPC-C schema is non-intrusive. Previous work by Oliveira and Bernardino [11] and Hrubaru and Fotache [29] revealed issues running all 22 of the TPC-H queries against new architecture NewSQL databases due to lack of support for the SQL HAVING clause, view creation capabilities, and excessively long-running query conditions. Initial tests performed as part of this research revealed similar incompatibility issues as well as hang-ups in benchmark execution with the CH-benCHmark TPC-H comparable queries. The subset consisting of seven of the CH-benCHmark TPC-H comparable queries, which ran without issue, was placed in the OLTP-Bench configuration file to be used for testing.

Based on the assumption that a One-Way ANOVA would be utilized for statistical analysis, a sample size based upon a power analysis using the G*Power application [47] indicated a need for 21 bare metal and 21 virtualized runs of the benchmark test against the NewSQL databases. The first set of tests in the experiment in this research effort was the execution of the CH-benCHmark using the OLTP-Bench testbed against

the NuoDB database in a bare metal environment. The creation and loading of the test databases were performed in separate, sequential steps from the execution of the benchmark test to delineate the response of the system under test to these operations, as part of an effort to ensure "cold" runs of the benchmark workload [48]. The stopping and restarting of the database after loading causes a flushing of the database server buffer pool, eliminating data caching between runs and enhancing consistency in values of performance metrics. Preliminary benchmark runs revealed that system performance metrics stayed in a consistent range in benchmark runs lasting as long as one hour. It was also determined that the entire set of TPC-H like queries was completed in approximately five minutes, even in the presence of concurrent OLTP transactions. Therefore, benchmark runs of fifteen minutes were used for data collection runs to allow three sets of the TPC-H queries.

Once all experiments had been run under bare metal conditions, the "treatment" (virtualization software) was installed on the same hardware used for the first set of experiments. VMware 6.5 [49] served as the virtualization software utilized in the experiment. VMware is a major player in the virtualization space, garnering an 80.7 % share of the 2017 virtualization market [50]. A single virtual machine closely matching the specifications of the physical machine on which it resides was created on the nodes of the database cluster. Each database node used a thin-provisioned, 400GB disk configured using a SCSI-controller in dependent mode and located in a VMware datastore. The VMware Paravirtual SCSI controller was used following Dakic [51], Goldsand and Brown [52], and VMware [53]. The VMware VM Network was configured with the physical NIC connected to a vSwitch, which was connected to the virtual machine.

Each virtual machine was installed with the identical software configuration, i.e., operating system, NewSQL database, and performance metric gathering software, as was installed on the machine in its bare metal state. The same type and number of benchmark runs were performed, and performance data was collected. Once data collection was completed under virtualized conditions, the two sets of data required for the independent variable, bare metal versus virtualized environment, were made available for analysis.

IV. RESULTS AND DISCUSSION

Care was taken to ensure the appropriate statistical approach was taken in the comparison of data collected in bare metal and virtualized environments. Using SPSS, an evaluation to determine the normality of the individual datasets was completed via a Shapiro Wilk test, followed by either an ANOVA or Kruskal-Wallis H (K-W) test depending on whether a parametric or non-parametric test was needed to determine statistical significance. A summary of the inferential statistical tests and changes due to virtualization is provided in Table I, and a detailed description of the statistical results follows.

The results of the ANOVA analyzing CPU utilization for the SM node indicated that the difference in the means was statistically significant, $F(1,40)=3083.879$, $p<0.001$, and the null hypothesis was rejected. The rejection of the null hypothesis led to the acceptance of the alternative hypothesis

that there was a difference in CPU use when virtualization is implemented. A comparison of the mean values of CPU utilization between bare metal (M=1.492, SD=0.052) and virtualized (M=2.26, SD=0.057) environments for the SM node indicated an increase of 62.5%. In the case of the TE node, the ANOVA also demonstrated that the difference in the means is statistically significant, $F(1,40)=5027.822$, $p<0.001$, and the null hypothesis can be rejected. A comparison of the mean values of CPU utilization between bare metal (M=2.684, SD=0.049) and virtualized environments (M=4.546, SD=0.110) indicated an increase of 69.4 %.

TABLE I. RESULTS

Perf Measure	SM Virtualized vs. Bare Metal			TE Virtualized vs. Bare Metal		
	Test	p-value	Chg	Test	p-value	Chg
CPU Util	ANOVA	$p < 0.001$	62.5%	ANOVA	$p < 0.001$	69.4%
Memory Util	ANOVA	$p < 0.001$	1.4%	K-W	$p < 0.001$	2.7%
Disk I/O	K-W	$p < 0.001$	37.3%	K-W	$p < 0.001$	N/A
Network I/O	ANOVA	$p < 0.001$	-12.4%	ANOVA	$p < 0.001$	-8.6%
TPS	ANOVA	$p < 0.001$	66.1%			

The SE experienced a 62.5% increase, and the TE, a 69.4% increase in CPU utilization with the additional layer of virtualization software in place. The amount of overhead caused by virtualization can vary based on the VM's workload capable of running directly on a physical processor and the amount requiring virtualization [54]. Tudor [55] found increases in CPU utilization ranging from 38% to 45% with open source RDBMS, attributing the increase to increased I/O wait times. The current research found increases occurring in user and system CPU utilization. A comparison of bare metal and virtualized NoSQL environments found that CPU utilization increased by roughly 29% under mixed (read/write) loads [56]. In experiments using CPU-intensive benchmark applications, Pousa and Rufino [57] found decreases in CPU efficiency due to the existence of an ESXi 6.0 virtualization layer in areas of process creation, disk to RAM transfers, context switching, and system call overhead. The current research indicates that the impact of virtualization on CPU utilization on the NewSQL RDBMS was greater than levels found in open source RDBMSs and NoSQL database testing in the studies referenced. The increased CPU utilization should be strongly considered in the migration or implementation of NewSQL on virtualized systems.

The results of the ANOVA for memory utilization for the SM node indicated that the difference in the means was statistically significant, $F(1,40)=25.746$, $p<0.001$, leading to the acceptance of the alternative hypothesis that there was a difference in memory use when virtualization is implemented. A comparison of the mean values of memory utilization between bare metal (M=4702.894, SD=56.362) and virtualized (M=4766.631, SD=11.696) environments indicated only a

slight increase of 1.4% in the SM node. In the case of the TE node, the assumptions necessary to use ANOVA were not met, and a Kruskal Wallis H test was used. The results of the Kruskal Wallis H indicated that the difference in the means was statistically significant, $\chi^2(1)=23.694$, $p<0.001$, and the null hypothesis could be rejected. A comparison of the mean values of memory utilization between bare metal (M=3163.598, SD=37.606) and virtualized (M=3249.873, SD=38.754) environments indicated an increase of 2.7% in the TE node.

Concerning memory utilization, the difference between the two environments, bare metal and virtualized, indicated an increase in memory utilization in the virtualized environment of 1.4% for the SM and 2.7% for the TE. The additional overhead is present and can be attributed to the addition of the virtualization layer, but given the low percentage increases, adjustments in memory size are not warranted.

In the analysis of disk I/O, the assumptions to use an ANOVA were not met in either the case of the SM or the TE, so a Kruskal Wallis H test was used. The results of the Kruskal Wallis H indicated that the means for the SM node were significantly different, $\chi^2(1)=30.767$, $p<0.001$, and the null hypothesis was rejected. A comparison of the mean values of disk I/O utilization between bare metal (M=3574.681, SD=358.112) and virtualized (M=4907.644, SD=60.220) environments for the SM indicated an increase of 37.3%. In the case of the TE node, the Kruskal Wallis H test yielded $\chi^2(1)=1.339$, $p=0.247$. The data failed to provide the evidence needed to reject the null hypothesis, and it was therefore retained as $p>0.05$.

A statistically significant difference in the disk I/O measurement was reflected in increased disk I/O in the virtualized SM node. Using system benchmark performance tools, Pousa and Rufino [57] found disk performance under ESXi virtualized conditions to be very similar to bare metal. Shirinbab, Lundberg [56] found disk I/O write rates to be the same or higher with virtualized NoSQL databases. Tudor [55] found that disk I/O was higher in virtualized open source RDBMS through the collection of values of OS disk I/O as percentages. The faster data can be moved from disk to RAM, the greater availability of data for the RDBMS. The disk-intensive nature of RDBMSs depends on I/O bandwidth to function properly [58]. Lee and Fox [59] suggested that greater IOPS are good for database systems. The existence of statistical significance in datasets collected on the SM node allowed for the comparison of bare metal and virtualized NewSQL environments and a 37.3% increase in the disk I/O rate was observed in the virtualized environment. The increase in disk I/O can be attributed to the VMware Paravirtualized SCSI controller as noted by Dakic [51], although the increase in disk I/O in this research exceeded the 12% found in that research effort which used vSphere 6.0. Additional statistical analysis on data collected as part of the current research effort found that along with increased disk I/O rate as measured in KB/s, the virtualized SM node had increased I/O operations per second (IOPS). The virtualized SM IOPS (M=175.78, SD=2.12) exceeded bare metal SM IOPS (M=132.36, SD=10.88) by 32.8%, and the difference in the means of the two datasets was shown to be statistically significant via a

Kruskal Wallis H test, $\chi^2(1)=30.767$, $p<0.001$. The data indicates an increase in disk I/O rate under virtualized conditions for the SM node, whose role is to manage and maintain complete, consistent, independent copies of the entire database. In a discussion with a VMware engineer, he stated that the improvements emphasize the importance of disk I/O drivers in the software and that VMware drivers will coalesce disk I/O reads and writes, but the proprietary nature of the software prevented extensive discussion (D. Robertson, personal communication, May 6, 2020). Given disk I/O is often a bottleneck for increased system performance, the increases found is positive. The disk I/O data collected on the TE would not allow the rejection of the null hypothesis that stated there was no effect due to virtualization. It is worth noting that given that the role of the TE is to cache database tables in memory, accept database requests and execute SQL queries, the values of disk I/O KB/s recorded were in the single digits, which would decrease the potential impact on the system overall.

The results of the ANOVA performed on data collected for network I/O measurements indicated the difference in the means were statistically significant both for the SM, $F(1,40)=238.429$, $p<0.001$, and the TE, $F(1,40)=113.81$, $p<0.001$, and the null hypothesis was rejected for both nodes. A comparison of the mean values of network I/O between bare metal ($M=14270.151$, $SD=396.245$) and virtualized ($M=12502.987$, $SD=343.571$) environments indicated a decrease of 12.4% for the SM node. The TE node saw a difference between bare metal ($M=14491.517$, $SD=404.175$) and virtualized ($M=13243.229$, $SD=352.401$) environments corresponding with a decrease of 8.6%.

Both the virtualization of CPU resources and virtualized network adaptors will increase the time to transmit data packets [54]. The components required in the processing of virtualized network I/O are virtual network drivers, known as the vNIC, the vSwitch, the VMkernel, and the physical NIC driver [60]. This results in virtualization overhead impacting three of the four networking-specific components involved. The benchmarking server remained in a bare metal state for all experiments to ensure that virtualization effects were confined to the database nodes. The SM and the TE experienced a decrease in the network I/O (KB/s) of 12.4% and 8.6%, respectively. The statistical analysis of the data collected allowed for the acceptance of the alternate hypothesis that the additional overhead imposed is due to the virtualization of the NewSQL RDBMS. Direct-path I/O was not supported by the NIC used in the systems tested, but if available, would provide a means to minimize these performance decreases.

The fifth and final hypothesis sought to provide focus on the transactional volume of the NewSQL database. Since the necessary assumptions were met, a One-Way ANOVA was performed on the datasets using SPSS. The results of the ANOVA indicated that there was a statistically significant difference in the means, $F(1,40) = 683.821$, $p < 0.001$, indicating the null hypothesis could be rejected and the alternative hypothesis accepted. A comparison of the mean values of TPS between bare metal ($M=16.474$, $SD=1.189$) and virtualized ($M=27.821$, $SD=1.534$) environments indicated an increase of 66.1%.

A relevant, overall throughput metric for an RDBMS is the rate of database transactions, TPS. The TPS values, as reported by the benchmarking software, increased 66.1% in the virtualized environment as compared to bare metal. The improvement in TPS found in this research contradicts the results of Tudor [55], but it must be pointed out that the virtualization software was VMware ESXi 5.0 as compared to version 6.5 used in the current research. Essential to the performance of an RDBMS is the ability to quickly and efficiently read data from disk into memory when needed and to write new or updated data from memory to disk [61]. The 37.3% increase in the disk I/O and the ample CPU cycles present on the SM node provided an environment with increased processing potential. Such conditions could give the system an increased ability for transaction throughput, but additional research will be required for this to be definitively demonstrated.

V. CONCLUSION

Virtualization continues to play an important role in the delivery of IT Services in on-premise data centers and via the cloud [4]. With the virtualizing of computing resources, organizations have been able to enhance and improve the management and utilization of IT resources. New architectural approaches found in NewSQL allow for the use of SQL and adherence to relational database standards, characteristics, and guarantees in the context of the immense volumes of data present in Big Data applications [2]. Just as the use of traditional relational databases and NoSQL technologies in virtualized environments occurred, the use of NewSQL in a virtualized environment is to be expected [25]. The absence of literature reflecting research to quantify the performance impact of virtualization on NewSQL RDBMS served as a motivation for this effort. The work presented allows for a better understanding and quantification of this impact of virtualization, providing benefits to organizations seeking to virtualize NewSQL servers, and represents an effort to fill the gap in the literature on this specific topic.

The evidence of the virtualization penalty in RDBMSs [55] and NoSQL database systems [56] is present in the literature. However, with the advent of NewSQL technology and continuous improvement in virtualization software, existing paradigms should be revisited. In this research, non-trivial virtualization penalties were identified in CPU utilization and network I/O, but memory utilization was only nominally impacted, and both disk I/O and TPS values were improved in the virtualized environment. The performance improvements would indicate that the new architecture NewSQL solutions may involve dynamics different than those present in traditional and NoSQL database solutions. The architecture of the Nuodb NewSQL RDBMS creates a dependency on disk I/O for the SM, but not the TE, which is memory-dependent. Given the virtualization goal of sharing underutilized hardware resources between virtual machines, existing paradigms must be reconsidered in light of ideas such as the complementary nature of the needs of the SM and TE, which might be amendable to separate VMs on the same physical machine. Additional research surrounding such synergies should be explored.

REFERENCES

- [1] O. Ylijoki and J. Porras, "Perspectives to definition of Big Data: A mapping study and discussion," *Journal of Innovation Management*, vol. 4, no. 1, pp. 69-91, 2016, doi: 10.24840/2183-0606_004.001_0006.
- [2] A. Pavlo and M. Aslett, "What's really new with NewSQL?," *ACM SIGMOD Record*, vol. 45, no. 2, pp. 45-55, 2016, doi: 10.1145/3003665.3003674.
- [3] A. Dhanapal, M. V. Saravanakuma, and M. Sabibullah, "Emerging Big Data storage architectures: A new paradigm," *i-manager's Journal on Pattern Recognition*, vol. 4, no. 2, pp. 31-41, 2017, doi: 10.26634/jpr.4.2.13732.
- [4] H. Ur Rahman, G. Wang, J. Chen, and H. Jiang, "Performance evaluation of hypervisors and the effect of virtual cpu on performance," in *2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCOM/IOP/SCI)*, Guangzhou, China, 2018: IEEE, pp. 772-779, doi: 10.1109/SmartWorld.2018.00146.
- [5] I. Pogarcic, D. Krnjak, and D. Ozanic, "Business benefits from the virtualization of an ICT infrastructure," *International Journal of Engineering Business Management*, vol. 4, no. Godište 2012, pp. 4-42, 2012.
- [6] I. Odun-Ayo, O. Ajayi, and C. Okereke, "Virtualization in Cloud Computing: Developments and trends," in *2017 International Conference on Next Generation Computing and Information Systems (ICNGCIS)*, Jammu, India 2017: IEEE, pp. 24-28, doi: 10.1109/ICNGCIS.2017.10.
- [7] M. Wardat, M. Al-Ayyoub, Y. Jararweh, and A. A. Khreishah, "Cloud data centers revenue maximization using server consolidation: Modeling and evaluation," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2018: IEEE.
- [8] R. Kumar and S. Charu, "An importance of using virtualization technology in cloud computing," *Global Journal of Computers & Technology*, vol. 1, no. 2, pp. 56-60, 2015.
- [9] H. Fatima and K. Wasnik, "Comparison of SQL, NoSQL and NewSQL databases for Internet of Things," in *2016 IEEE Bombay Section Symposium (IBSS)*, Baramati, India, 2016: IEEE, pp. 1-6, doi: 10.1109/IBSS.2016.7940198.
- [10] S. Binani, A. Gutti, and S. Upadhyay, "SQL vs. NoSQL vs. NewSQL-a comparative study," *Communications on Applied Electronics*, vol. 6, no. 1, pp. 1-4, 2016.
- [11] J. Oliveira and J. Bernardino, "Newsql databases: Memsql and VoltDB experimental evaluation," presented at the 9th International Conference on Knowledge Engineering and Ontology Development, Funchal, Portugal, 2017.
- [12] K. Kaur and M. Sachdeva, "Performance evaluation of NewSQL databases," in *2017 International Conference on Inventive Systems and Control (ICISC)* Coimbatore, India 2017: IEEE, pp. 1-5, doi: 10.1109/ICISC.2017.8068585.
- [13] M. Murazzo, P. Gómez, N. Rodríguez, and D. Medel, "Database NewSQL Performance Evaluation for Big Data in the Public Cloud," in *Conference on Cloud Computing and Big Data*, 2019: Springer, pp. 110-121.
- [14] D. Bitton et al., "A measure of transaction processing power," *Datamation*, vol. 31, no. 7, pp. 112-118, 1985.
- [15] B. Scalzo, *Database benchmarking and stress testing: An evidence-based approach to decisions on architecture and technology*. New York, NY: Apress, 2018.
- [16] D. Rajesh, A. A. Ahmed, M. I. T. Hussan, and V. Bollapalli, "Virtualization and its role in Cloud Computing environment," *International Journal of Computer Sciences and Engineering*, vol. 7, no. 4, pp. 1131-1136, 2019, doi: 10.26438/ijcse/v7i4.11311136.
- [17] Z. Li, M. Kihl, Q. Lu, and J. A. Andersson, "Performance overhead comparison between hypervisor and container based virtualization," in *2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA)*, Taipei, Taiwan, 2017: IEEE, pp. 955-962, doi: 10.1109/AINA.2017.79.
- [18] H. Fayyad-Kazan, L. Perneel, and M. Timmerman, "Benchmarking the performance of Microsoft Hyper-v server, Vmware ESXi and Xen hypervisors," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 4, pp. 922-933, 2013.
- [19] B. Dordevic, V. Timčenko, N. Kraljević, and N. Davidović, "File system performance comparison in full hardware virtualization with ESXi and Xen hypervisors," in *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*, East Sarajevo, Bosnia, 2019: IEEE, pp. 1-5, doi: 10.1109/INFOTEH.2019.8717664.
- [20] W. Graniszewski and A. Arciszewski, "Performance analysis of selected hypervisors (virtual machine monitors-VMMs)," *International Journal of Electronics and Telecommunications*, vol. 62, no. 3, pp. 231-236, 2016, doi: 10.1515/eletel-2016-0031.
- [21] C. H. Kao, "Testing and evaluation methods for cloud environments: A review," in *Proceedings of the 2017 International Conference on E-commerce, E-Business and E-Government*, Turku, Finland, 2017: ACM, pp. 56-60, doi: 10.1145/3108421.3108435.
- [22] A. Vogelsgesang et al., "Get real: How benchmarks fail to represent the real world," in *Proceedings of the Workshop on Testing Database Systems*, Houston, TX, 2018: ACM, pp. 1-6, doi: 10.1145/3209950.3209952.
- [23] M. Aslett, "What we talk about when we talk about NewSQL," in *Too much information* vol. 2018, ed. https://blogs.the451group.com/information_management/2011/04/06/what-we-talk-about-when-we-talk-about-newsql/: the 451 group, 2011.
- [24] M. Stonebraker, "New SQL: An alternative to NoSQL and old SQL for new OLTP apps," in *BLOG@CACM* vol. 2018, ed. <https://cacm.acm.org/blogs/blog-cacm/109710-new-sql-an-alternative-to-nosql-and-old-sql-for-new-oltp-apps/fulltext>: ACM, 2011.
- [25] K. Grolinger, W. A. Higashino, A. Tiwari, and M. A. Capretz, "Data management in cloud environments: NoSQL and NewSQL data stores," *Journal of Cloud Computing: Advances, Systems and Applications*, vol. 2, no. 1, p. 22, 2013, doi: 10.1186/2192-113X-2-22.
- [26] E. Awasthi, S. Agrawal, and R. Pandey, "A survey on NoSQL and NewSQL data stores for Big Data management," *Global Journal of Engineering Science and Research*, vol. 5, no. 7, pp. 269-277, 2018, doi: 10.5281/zenodo.1313633.
- [27] A. Almassabi, O. Bawazeer, and S. Adam, "Top NewSQL databases and features classification," *International Journal of Database Management Systems (IJDMS)*, vol. 10, no. 2, pp. 11-31, 2018, doi: 10.5121/ijdms.2018.10202.
- [28] J. M. Monteiro, A. Brayner, and J. A. Tavares, "What comes after NoSQL? NewSQL: A new era of challenges in DBMS scalable data processing," in *Tópicos em Gerenciamento de Dados e Informações. Minicursos do XXXI Simpósio Brasileiro de Banco de Dados (SBBDD)*, Salvador, Brazil, 2016, pp. 27-56.
- [29] I. Hrubaru and M. Fotache, "On the performance of three in-memory data systems for on line analytical processing," *Informatica Economica*, vol. 21, no. 1, pp. 5-15, 2017.
- [30] O. Borisenko and D. Badalyan, "Evaluation of SQL benchmark for distributed in-memory database management systems," *International Journal of Computer Science and Network Security* vol. 18, no. 10, pp. 59-63, 2018.
- [31] Apache.org. "In-memory computing platform." <https://ignite.apache.org/> (accessed).
- [32] "What is VoltDB's database product?" <https://www.voltDB.com/product/> (accessed 8/5/19, 2019).
- [33] CentOS. (2019). The CentOS Project, www.centos.org. Accessed: 11/5/2019. [Online]. Available: <https://www.centos.org/>.
- [34] "What Is NuoDB?" <https://www.nuodb.com/product-overview> (accessed).
- [35] R. Verma, "Understanding the technological trends and quantitative analysis of NewSQL databases," Masters, Department of Computer Science and Electrical Engineering, University of Maryland, ProQuest Dissertations & Theses Global, 2017. [Online]. Available: <https://proxy.cecylibrary.com/login?url=https://search-proquest-com.proxy.cecylibrary.com/docview/1940284796?accountid=144789>.
- [36] NuoDB, "NuoDB Architecture." [Online]. Available: <http://go.nuodb.com/rs/099-DVI-451/images/Technical->

- Whitepaper.pdf?aliId=eyJpJoiME9hNmV4OXZyWkrUYzZZbSIsInQiOjJWbWIDMVfQK0V4VFBGalJBM1g2THJ3PT0ifQ%253D%253D.
- [37] NuoDB. "Installing NuoDB on Linux." <http://doc.nuodb.com/Latest/Content/Nuoadmin-Installing-NuoDB-on-Linux.htm> (accessed 12/15/19).
- [38] nmon for Linux. (2017). [Online]. Available: <https://sourceforge.net/projects/nmon/>.
- [39] M. F. Khalid, B. I. Ismail, and M. N. M. Mydin, "Performance comparison of image and workload management of edge computing using different virtualization technologies," *Advanced Science Letters*, vol. 23, no. 6, pp. 5064-5068, 2017, doi: 10.1166/asl.2017.7310.
- [40] M. Kaczmarek, P. Perry, J. Murphy, and A. O. Portillo-Dominguez, "In-test adaptation of workload in enterprise application performance testing," in *Proceedings of the 8th ACM/SPEC on International Conference on Performance Engineering Companion*, 2017: ACM, pp. 69-72.
- [41] D. E. Difallah, A. Pavlo, C. Curino, and P. Cudre-Mauroux, "Otp-bench: An extensible testbed for benchmarking relational databases," *Proceedings of the VLDB Endowment*, vol. 7, no. 4, pp. 277-288, 2013.
- [42] J. Darmont, "Data processing benchmarks," *Encyclopedia of Information Science and Technology*, pp. 1741-1747, 2014, doi: 10.4018/978-1-4666-5888-2.ch167.
- [43] F. Coelho, J. Paulo, R. Vilaça, J. Pereira, and R. Oliveira, "Htapbench: Hybrid transactional and analytical processing benchmark," in *Proceedings of the 8th ACM/SPEC on International Conference on Performance Engineering*, 2017: ACM, pp. 293-304.
- [44] I. Kovacevic and I. Mekterović, "Alternative business intelligence engines," in *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2017: IEEE, pp. 1385-1390.
- [45] D. Seybold and J. Domaschka, "Is Distributed Database Evaluation Cloud-Ready?," in *European Conference on Advances in Databases and Information Systems*, 2017: Springer, pp. 100-108.
- [46] R. Cole et al., "The mixed workload CH-benCHmark," in *Proceedings of the Fourth International Workshop on Testing Database Systems*, 2011: ACM, p. 8.
- [47] F. Faul, E. Erdfelder, A.-G. Lang, and A. Buchner, "G*Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences.," *Behavior Research Methods*, vol. 39, pp. 171-191, 2007.
- [48] M. Raasveldt, P. Holanda, T. Gubner, and H. Mühleisen, "Fair benchmarking considered difficult: Common pitfalls in database performance testing," in *Proceedings of the Workshop on Testing Database Systems*, 2018: ACM, p. 2.
- [49] VMware.com. "VMware.com." VMware.com. <https://www.vmware.com/> (accessed 7/19/2019, 2019).
- [50] G. Chen, "Worldwide Virtual Machine Software MarketShares, 2017: Virtualization Still Showing Positive Growth," *International Data Corporation*, 2018. [Online]. Available: <https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/products/vmware-idx-virtual-machine-market-shares-2017.pdf>
- [51] V. Dakic, "Influence of virtual storage controller type on Microsoft SQL Server 2019 performance," *Annals of DAAAM & Proceedings*, vol. 30, 2019.
- [52] B. Goldsand and C. Brown, "SAP® Sybase® Adaptive Server Enterprise on VMware vSphere® essential deployment tips," 2013.
- [53] VMware, "vSphere Virtual Machine Administration." [Online]. Available: <https://docs.vmware.com/en/VMware-vSphere/6.5/vsphere-esxi-vcenter-server-65-virtual-machine-admin-guide.pdf>
- [54] VMware, *Performance Best Practices for VMware vSphere 6.5*, Palo Alto: VMware, 2017. [Online]. Available: https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/techpaper/performance/Perf_Best_Practices_vSphere65.pdf.
- [55] A. Tudor, "A study of open-source relational database performance in virtual computing environments," *Doctoral Dissertation*, Northcentral University, Ann Arbor, 2014. [Online]. Available: <https://login.ctu.idm.oclc.org/?url=http://search.proquest.com/docview/1506579000?accountid=26967>.
- [56] S. Shirinbab, L. Lundberg, and E. Casalicchio, "Performance evaluation of container and virtual machine running Cassandra workload," in *2017 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech)*, Rabat, Morocco 2017: IEEE, pp. 1-8, doi: 10.1109/CloudTech.2017.8284700.
- [57] D. Pousa and J. Rufino, "Evaluation of type-1 hypervisors on desktop-class virtualization hosts," *Iadis Journal on Computer Science and Information Systems*, vol. 12, no. 2, pp. 86-101, 2017.
- [58] M. G. Xavier, K. J. Matteussi, F. Lorenzo, and C. A. De Rose, "Understanding performance interference in multi-tenant cloud databases and web applications," in *2016 IEEE International Conference on Big Data (Big Data)*, Washington, DC, 2016: IEEE, pp. 2847-2852, doi: 10.1109/BigData.2016.7840933.
- [59] H. Lee and G. Fox, "Big Data Benchmarks of High-Performance Storage Systems on Commercial Bare Metal Clouds," in *2019 IEEE 12th International Conference on Cloud Computing (CLOUD)*, 2019: IEEE, pp. 1-8.
- [60] D. B. Oljira, A. Brunstrom, J. Taheri, and K.-J. Grinnemo, "Analysis of network latency in virtualized environments," in *2016 IEEE Global Communications Conference (GLOBECOM)*, 2016: IEEE, pp. 1-6.
- [61] R. Elmasri and S. Navathe, *Fundamentals of database systems*, 7th ed. Boston, MA: Pearson, 2017.