Transaction Performance vs. Moore's Law: A Trend Analysis

Raghunath Nambiar[1] and Meikel Poess[2]

[1] Cisco Systems, Inc., 3800 Zanker Road, San Jose, CA 95134, USA
+1- 408-527-3052
rnambiar@cisco.com
[2] Oracle Corporation, 500 Oracle Pkwy, Redwood Shores, CA 94065, USA
+1-650-633-8012
meikel.poess@oracle.com

Abstract. Intel co-founder Gordon E. Moore postulated in his famous 1965 paper that the number of components in integrated circuits had doubled every year from their invention in 1958 until 1965, and then predicted that the trend would continue for at least ten years. Later, David House, an Intel colleague, after factoring in the increase in performance of transistors, concluded that integrated circuits would double in performance every 18 months. Despite this trend in microprocessor improvements, your favored text editor continues to take the same time to start and your PC takes pretty much the same time to reboot as it took 10 years ago. Can this observation be made on systems supporting the fundamental aspects of our information based economy, namely transaction processing systems?

For over two decades the Transaction Processing Performance Council (TPC) has been very successful in disseminating objective and verifiable performance data to the industry. During this period the TPC's flagship benchmark, TPC-C, which simulates Online Transaction Processing (OLTP) Systems has produced over 750 benchmark publications across a wide range of hardware and software platforms representing the evolution of transaction processing systems. TPC-C results have been published by over two dozen unique vendors and over a dozen database platforms, some of them exist, others went under or were acquired. But TPC-C survived. Using this large benchmark result set, we discuss a comparison of TPC-C performance and price-performance to Moore's Law.

Keywords: Trends in System and Database Performance, Benchmark Standards.

1 Introduction

Intel co-founder Gordon E. Moore described in his famous 1965 paper [1] that the number of components in integrated circuits had doubled every year from the invention of the integrated circuit in 1958 until 1965, and predicted that the trend would continue for at least ten years. Around 1970 Caltech professor, VLSI pioneer, and entrepreneur Carver Mead coined the term the "Moore's Law". Moore slightly altered his formulation of the law over time. In 1975, Moore refined his projection to a doubling every two years [2]. Later, David House, an Intel colleague, who factored in the

R. Nambiar and M. Poess (Eds.): TPCTC 2010, LNCS 6417, pp. 110–120, 2011.

increase in performance of transistors, concluded that integrated circuits would double in performance every 18 months. History has shown that the capabilities of many digital electronic devices, such as processors, memory, etc. are strongly linked to Moore's law. All of these are improving at exponential rates. His prediction has proven to be very accurate. Some suggest that this is because the law is used in the semiconductor industry to guide long-term planning and to set targets for research and development. Figure 1 shows the CPU transistor count starting from 1971 to 2008. The graph is also annotated with the major milestones in the history of the Intel processor product line.

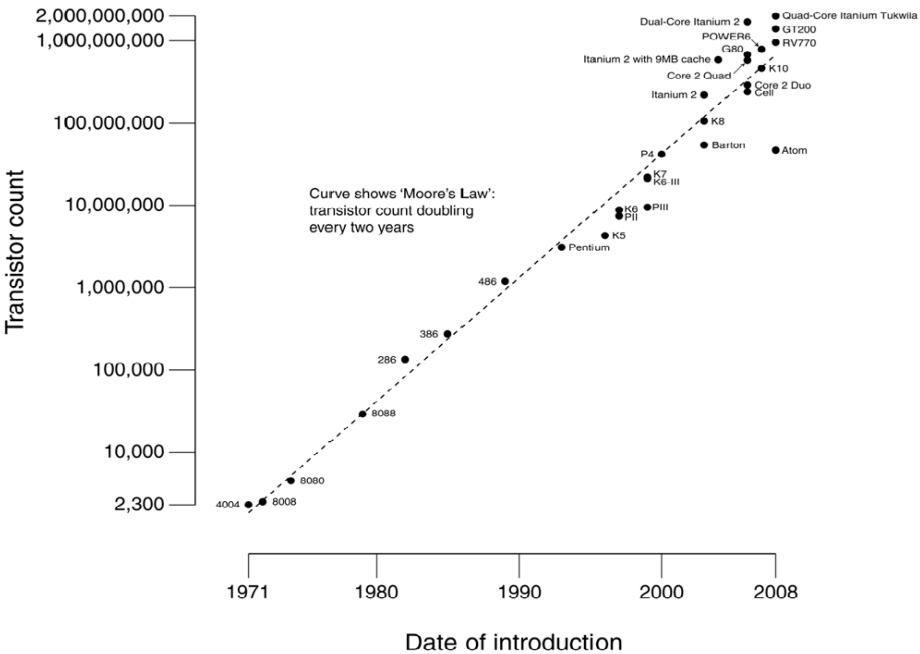

Fig. 1. Number of transistors on a single CPU (source Wikipedia.org)

But, does the exponential transistor growth, predicted by Moore’s Law, also translate into exponentially greater performance of systems supporting the fundamental aspect of our information based economy, namely transaction processing systems? Many suggest it doesn’t. Especially complex parallel applications that involve a diverse set of components are difficult to scale with processor performance because of their non-processor bottlenecks such as disks, software components, storage and connectivity.

In this paper we look at transaction processing systems, which are directly and indirectly involved in virtually every moment of our lives, e.g. phone calls, bank transactions, stock trades and flight reservations. As processor speeds have increased over the last two decades, performance of other critical components of transaction processing systems, such as disk latencies, I/O controllers and network latencies, have lagged far behind. Additionally, traditional algorithms may not be designed to scale to transaction processing systems with a larger number of processors, large amount of memory and large number of disks. In this paper we look at performance improvements of transaction processing systems, not just one component but total system that includes database server, storage, connectivity and software including database management systems and application middle-tier.

In order to show relative performance gains of large complex application systems over time and to draw a comparison to Moore's Law, one needs consistent, verifiable performance data over a long period of time across a diverse set of platforms. The two most prominent industry standard benchmark organizations to publish benchmarks since the late 1980's are the Transaction Processing Performance Council (TPC), established 1987 and the Systems Performance Evaluation Corporation (SPEC), established 1988. The TPC's focus has been total system performance and price-performance under database workloads, including: server, storage, connectivity and software. All results have a price-performance metric audited by an independent TPC certified auditor. Like the TPC, the SPEC develops suites of benchmarks intended to measure system level performance. These suites are packaged with source code and tools and are extensively tested for portability across platforms before they are released. Unlike the TPC results, the SPEC results are peer audited. While the SPEC has been revising their benchmarks frequently the TPC has long lasting specifications. During its over 20 year long history the TPC has created and maintained significant benchmarks, such as TPC-A, TPC-B, TPC-C, TPC-H, TPC-W and TPC-E. Its flagship OLTP benchmark, the TPC-C, was first introduced in June 1992. Since then it has undergone several modifications. In its two decades of existence there have been over 750 results on dozens of hardware and software platforms. All major hardware and database vendors of yesterday and today have published TPC-C benchmarks, some of them are in business while others went out of business or were acquired. TPC-C survived, tracking the evolution of processor architectures (MIPS, RISC, CISC etc.), server architectures (rack mounted, SMP, clusters, blades etc.) and database technologies. There is no single benchmark standard that comes close at claiming such industry acceptance and life span. These factors make TPC-C the ideal candidate for conducting a performance and price-performance trend analysis that compares one of the most important factors that touch every second of our lives in the information era, namely transaction performance to Moore's law.

The remainder of this paper is organized as follows. Section 2 gives a brief overview of those parts of the TPC-C benchmark specification that are necessary to understand the subsequent sections. It also includes a history of the different revisions of TPC-C and explains why results from its different revisions can be used for the purpose of our comparison to Moore's Law. Section 3 demonstrates the performance and price-performance trends of TPC-C results from 1993 to 2010 and compares these trends to Moore's predictions.

2 Background of TPC's OLTP Benchmarks and Why TPC-C Is Suitable for Long Term Trend Analysis

TPC's first OLTP benchmark specification, TPC-A, was published in November 1989. Built upon Jim Gray's DebitCredit benchmark TPC-A formalized the rules, which all vendors had to obey in order to publish a benchmark result. About one year later, TPC-B was developed. TPC-B was a modification of TPC-A, using the same transaction type (banking transaction) but eliminating the network and user interaction components. The result was a batch transaction processing benchmark. After two years of development, in June 1992, TPC's third OLTP benchmark specification, TPC-C, was approved.

Most TPC-C systems are implemented in three tiers mimicking typical transaction processing systems. A TPC-C benchmark implementation is also referred to as the System Under Test (SUT):

1. Tier: Driver System
2. Tier: Client
3. Tier: Database Server

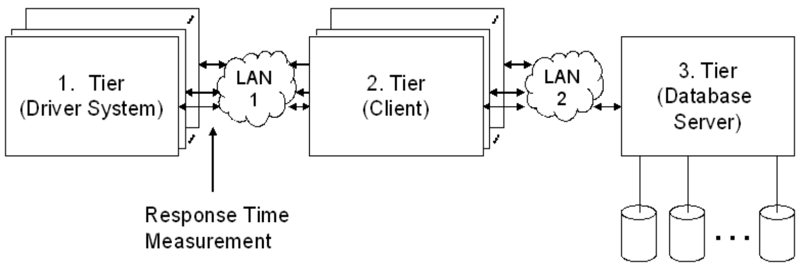

Fig. 2. Typical TPC-C System Setup, SUT (conceptual)

Compared to previous OLTP benchmarks, the TPC-C has a more representative database schema, several complex transactions exercising fundamental database functionalities and strict implementation requirements. It simulates a complete computing environment where a population of users executes transactions against a database. These transactions include entering and delivering orders, recording payments, checking the status of orders, and monitoring warehouse stock levels. In order to maintain TPC-C's applicability to systems of differing capacity, TPC-C implementations must scale both the number of users and the size of the database proportionally to the computing power of the system to be measured. The unit used in TPC-C to measure performance specifies the number of processed new-order transactions per minute (tpmC) while fulfilling the rest of the TPC-C transaction mix workload. There is response time threshold for each transaction: 90% of each type of transaction must have a response time of at most 5 seconds, except stock-level, which is allowed to be at most 20 seconds. TPC price/performance metric ($/tpmC) is the ratio of 3 year cost of ownership to tpmC. In addition to performance, dependability aspects of a fully production-ready system are also tested, by requiring full ACID properties (Atomicity, Consistency, Isolation and Durability).

2.1 History of TPC-C Revisions

In the first 18 months after approval TPC-C underwent two major revisions (Revision 2 and 3). Both revisions were comparable due to their minimal effect on existing results. Revision 4 failed to get the necessary support from TPC members. In October 2000 Revision 5 of TPC-C was approved. This revision contained substantial changes with a large impact on existing results, which made it non-comparable to previous revisions under the rigid TPC rules. Since benchmark sponsors[1] wanted to protect their investments in the large set of results that were already published across many hardware and software platforms, the TPC created a set of requirements and guidelines that allowed a onetime upgrade of results from Revision 3 to Revision 5. The following section highlights the changes in the different revisions and reasons why the results can be compared.

Revision 2.0, released October 20th 1993

Revision 2.0 included additional wording in the General Implementation Guideline section. This section defines the requirement of a benchmark implementation and specifically disallows "benchmark specials". Although obvious to the TPC, rules to explicitly disallow hardware and software that was written specifically and solely to run TPC-C was not included in the TPC-C specification until this major revision. Additionally new disclosure requirements including a numerical quantity summary was added to the specification. Revision 2.0 requires the most relevant numerical data to be included as part of the executive summary of the benchmark publications and not scattered in the Full Disclosure Report.

Revision 3.0, released February 15th 1995

Revision 3.0 added new transaction monitor requirements for routing transactions. The first results published under revisions 1 and 2 revealed that only limited transaction monitor functions were exercised. In order to avoid unrealistic implementations of these functions in the operating system, new requirements were added that require a full featured transaction monitor in case such functions were used. As it became apparent that the distinction between Wide Area Networks and Local Area Networks technologies in the context of TPC-C was becoming fuzzy, the distinction between the two was removed. Careful examination of results revealed that vendors were pricing the lowest cost terminal available on the market. However, terminals accounted for a substantial portion of the reported price. As a consequence, terminals were excluded from the pricing to refocus the price on more crucial components of the configuration.

Revision 5.0, released October 18th 2000

The changes in Revision 5 can be divided in pricing related changes and benchmark run rule related changes. The pricing related changes had no impact on the transactions-per-minute metric (tpmC), but changed the total cost of ownership (TCO) and hence impacted the price-performance metric. They reduced the maintenance support period from five to three years and increased the weekly uptime requirements from eight hours, Monday through Friday to 24-hours seven days a week. The changes also removed the requirement to price the terminal connectivity network (hubs, switches),

[1] A benchmark sponsor is a company that publishes a TPC benchmark result.

which refocuses the benchmark on the client-tier, database-tier and their connectivity. They also allowed pricing quotes from web pages and print materials, which accommodates growing acceptance of online and direct purchase models. This changed the source of pricing, but had no impact on actual pricing. The last pricing related change was the reduction of the disk space requirements from 180 to 60 days. This is the disk space sufficient to store and maintain the data generated during a period of 60 days (instead of 180 days) of activity with an average of 8 hours per day at the reported tpmC rate. The intention was to reduce the cost for running benchmarks, but it made the total cost of ownership non comparable to results from previous revision.

The run rule change had no impact on performance or on price-performance. It increased the measurement interval from 20 minutes to 2 hours ensuring that the system under test can sustain the reported tpmC and guaranteeing that modified database records are written to durable media every 30 minutes for two hours through check point mechanisms. The TPC allowed upgrading Revision 3 to Revision 5 results if benchmark sponsors re-priced their systems to accommodate the pricing changes explained above. The two increased measurement interval requirement was waved.

2.2 Why TPC-C Is a Good Candidate for Technology Trend Analysis

The previous section has shown that the TPC-C specification evolved to remain as representative as possible of current practice without fundamentally changing the workload and pricing model. There were changes on implementation requirements and pricing, but the benchmark fundamentals including data population, transactions, execution rules and metric remained unchanged. This makes the TPC-C workload the ideal candidate for performance and price-performance trend analysis.

The pricing changes impacted the overall price of a system. For example in Revision 2 the price of terminals accounted for a substantial portion of the reported price, reducing maintenance support pricing to 3 years down from 5 years, and reducing the disk space requirements to 60 days from 180 days. Although the impact of these changes to TPC-C performance and price-performance is low, the strict TPC rules do not allow comparisons of results across major revisions. This is because benchmark sponsors usually compete over single digit percent differences. For our analysis in the next sections we use the performance and price-performance metrics without any adjustments because the changes had very little impact from a historical trend perspective. First of all, the fundamentals of the benchmark did not change. Secondly, due to the significant improvements in performance while cost of almost every component dropped, even the unadjusted performance metric and price-performance metric reflected overall industry trend. And finally, such changes as mandating commercially available transaction monitor and removing the terminal requirements are reflections of general changes in the industry. In the early 90's customers implemented their own home grown transaction monitors, which were eventually replaced by feature rich commercial TP monitors. In those days OLTP users accessed the servers over directly connected terminals, those were later replaced by remote web users.

3 Comparison of TPC-C Metrics with Moore's Law

In this section we analyze whether TPC-C performance improvements over the last 20 years agree with Moore's Law. We analyze both TPC-C's primary performance [tpmC] and price-performance [$/tpmC] metrics.

TPC-C benchmarks are published using a large range of system configuration sizes, which range from single server with one processor and a few disks to large cluster of servers with hundreds of processors and thousands of disk drives. Consequently, performance ranges from hundreds to millions tpmC. In order to compare the performance across various system sizes, we normalize the TPC-C performance metric by dividing the reported tpmC number by the number of processors (sockets) that are configured in the 3rd tier, i.e. the database server. We refer to this normalized metric as the NtpmC. For each year we then compute the average NtpmC for all TPC-C publications. Since the price-performance metric already accounts for the higher performance by an increased system cost, we do not need to adjust the price-performance with the number of processors.

3.1 Comparison of TPC-C Performance with Moore's Law

In this section we compare NtpmC, which is TPC-C's primary performance metric, normalized to the number of processors, to the adjusted[2] Moore's Law. The triangle markers in Figure 3 indicate the average Ntpmc per year from 1993 to 2010 on a logarithmic scale (base=10). The triangles are annotated with the major milestones in TPC-C publications. The dotted line shows TPC-C performance improvement using Moore's law as amended by David House, i.e. performance doubles every 18 months. As a starting point we use the average NtpmC of 1995, as the number of results available for 1993 and 1994 is too small to indicate a representative performance. We calculate the dotted line with the following function:

$$f(y) = f(y_0) * 2^{\frac{2}{3}*(y-y_0)} \quad (1)$$

$f(y)$ is the performance projection for year y and $f(y_0) = 426.65\ NtpmC$ is the base performance with $y_0 = 1995$.

Interestingly, the performance graph calculated using Moore's Law almost superimposes the data points taken from TPC-C publications [NtpmC]. There are three areas where NtpmC slightly over and under-performs Moore's Law.

In the years 1993 and 1994 and from years 1996 to 2000 Moore's Law slightly underestimates NtpmC. There are only a few results available for the first years, one result for 1993 and four results for 1994. Hence, the numbers obtained for these years are unlikely to be representative for performance improvements of technology in those years. The years between 1995 and 1999 saw a respectable number of benchmark results (between 21 and 76). Hence, the average of these results can be viewed as a very close representation of the performance that was achievable in these years.

[2] Adjusted as by David House: processor performance doubles every 18 months.

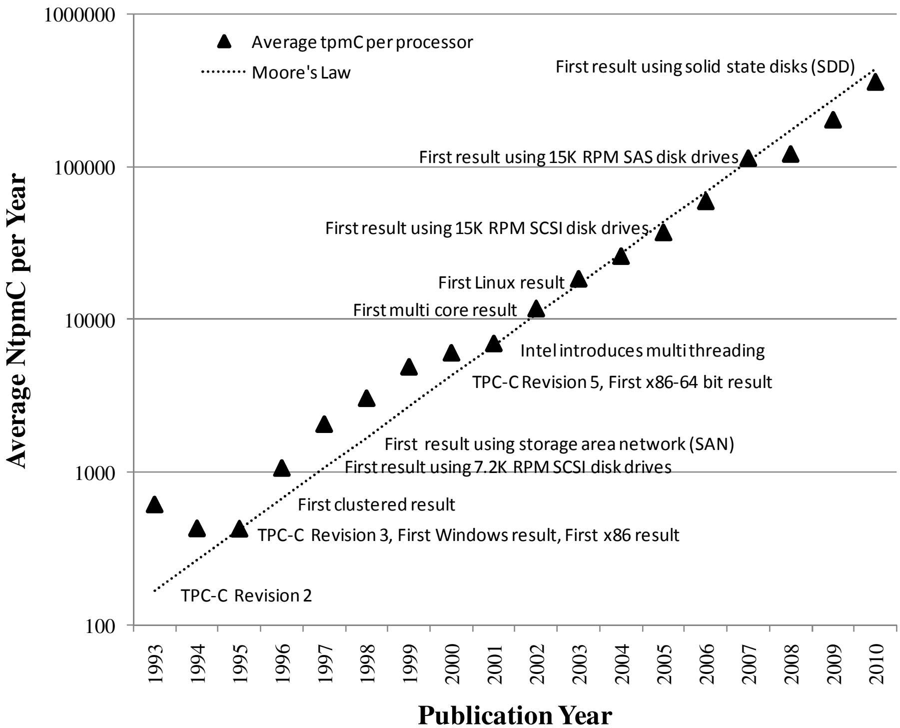

Fig. 3. TPC-C primary performance metric trend vs. Moore's Law

During the years 1996 to 2000, Moore's Law slightly underestimates NtpmC, while starting with 2008 it slightly overestimates NtpmC. In the period between 1996 and 2000 the publications used a very diverse combination of database and hardware platforms (about 12 hardware and 8 software platforms). Between 2001 and 2007 Moore's Law estimates NtpmC pretty accurately. In this period most publications were on x86 platforms running Microsoft SQL Server. Starting in 2008 Moore's Law over estimated NtpmC. At the same time the number of Microsoft SQL Server publications started decreasing. 2008 saw 21 benchmark publications, 2009 saw 6 benchmark publications and, so far, 2010[3] has seen 5 benchmark publications. This is mostly because of Microsoft's increased interest in the new TPC-E benchmark, introduced in 2007, while Oracle's DBMS took the lead in a number of TPC-C publications. Also, during this period the number of cores per processor for x86 processors increased to six. Another reason for the decline in the number of TPC-C publication is the increased benchmark implementation cost, mainly due to an increased number of spindles required to cope with increase in processor speed.

Ignoring the two time periods during which NtpmC differed slightly from Moore's Law, TPC-C performance increases are remarkably similar to the processor performance improvements. This suggests that TPC-C performance is attributed solely to

[3] As of June, 2010.

processor improvements, i.e. other system components, including system software, have not contributed much to the overall performance improvement of TPC-C results. However, this does not necessarily indicate that other system components and system software haven't improved over time – quite to the contrary. The ongoing improvements of processor speeds cause challenges on other system components, such that they do not become the next system's bottleneck. Hence, their performance also needed to be increased constantly. Components whose performance improvements were slower than those of processors were replicated. For instance, the number of disks required per processor increased from a dozen in 1993 to over 100 in 2010.

In case of software such replication is not necessarily possible. Especially the DBMS of the 3rd tier cannot be replicated because of the locking nature of the benchmark application. An attempt was made to use federated databases. However, the idea was abandoned shortly after it was introduced[4]. The development of multi-core processors imposed challenges on the operating systems as well as the DBMS because of the increased number of processes that need to be scheduled to occupy all processor cores. Similarly, the increased number of processes challenged shared memory access and semaphore handling mechanisms. Similarly, the dramatic increase in memory density challenged DBMS because the design of algorithms that were previously optimized for using disk I/O, such as sort operations, hash-joins, needed to be revisited to assure that they work optimally with large amounts of memory. The dramatic increase in processor cores and the resulting increase in TPC-C database size/number of database users (see Section 2) challenged the scalability capabilities of DBMS. The fact that TPC-C applications still scale with the vast increase in processor performance is an indication that DBMS scaling capabilities improved at rate of processors performance improvements. With every DBMS release its code path increases due to the introduction of additional features. These features are not necessarily introduced to improve TPC-C, but improve the overall usability of the product, such as security and manageability features.

3.2 Comparison of TPC-C Price-Performance with Moore's Law

In this section we compare TPC-C's price-performance metric with the adjusted[5] Moore's Law. TPC-C's price-performance metric is defined as the ratio of the total system price of the SUT divided by the transactions per minute [$/NtpmC].

The triangle markers in Figure 4 shows the price per NtpmC per year from 1993 to 2010 on a logarithmic scale (base=10). The dotted line shows Moore's law as amended by David House, i.e. performance doubles every 18 months. Similarly to the previous section, as a starting point we use the price per NtpmC in 1995 as the number of results available for 1993 and 1994 is too small to indicate representative performance for those years. We calculate the Moore's Law graph with the following function:

$$f(y) = f(y_0) * 2^{-\frac{2}{3}*(y-y_0)} \quad (2)$$

[4] Reasons for the abandonment of this technology would be pure speculation.

[5] Adjusted as by David House: processor performance doubles every 18 months.

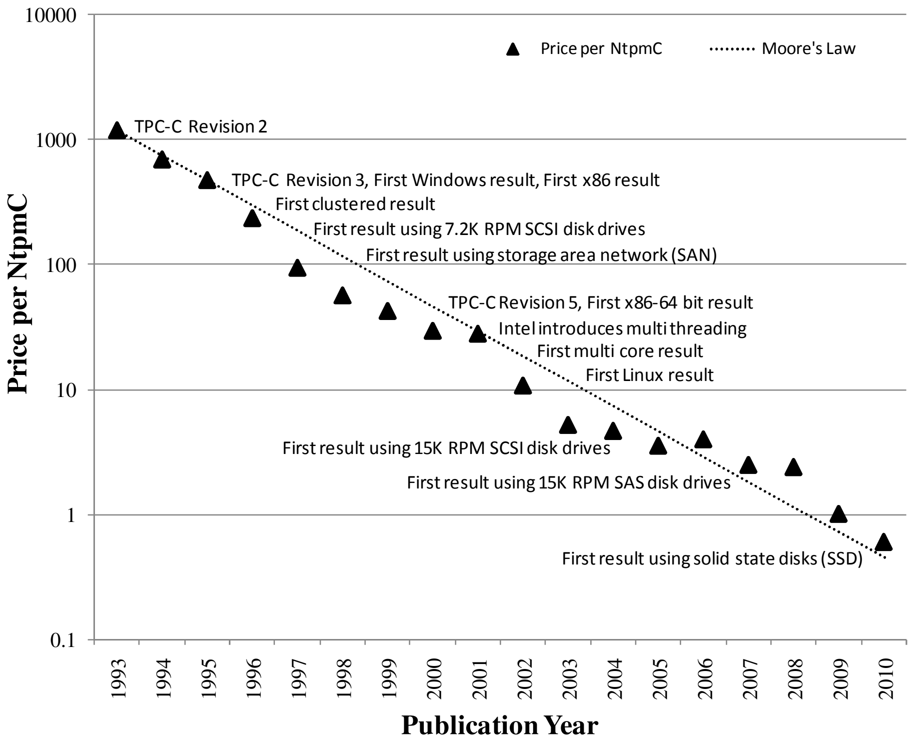

Fig. 4. TPC-C price-performance vs. Moore's Law

$f(y)$ is the price-performance projection for year y and $f(y_0) = 465.41\frac{\$}{Ntpmc}$ is the base price-performance with $y_0 = 1995$.

The trend in Figure 4 demonstrates that the gradual decline in the cost for transaction performance in line with Moore's Law. Interestingly, we saw in the comparison of TPC-C's normalized performance metric NtpmC with the calculated Moore's Law graph, we also see a phenomenal similarity between the price-performance trend and the price-performance predicted by Moore's Law.

There are three areas where the price-performance graph slightly over and under-performs the predictions of Moore's Law. Between the years 1996 and 2000 and between 2002 and 2005 Moore's Law slightly over-calculates the average transaction price as reported by benchmark sponsors. Contrary, between 2006 and 2010 Moore's Law slightly under-calculates the actual transaction cost. However, the overall trend of the actual graph and the calculated graph are phenomenally similar.

We had a closer look at the variations but the reasons were inconclusive, except that since 2006 the price-performance have been above the expectations potentially due to the extreme drop in system component pricing caused by competition, business efficiency and factory efficiency. This trend is likely to continue for the next several years.

4 Conclusion

Many research papers and technical articles have been published about Moore's Law and its applicability to hardware performance. This paper is a first of its kind attempt to analyze Moore's theorizations in respect to the performance and the cost for performance of complex application systems like transaction processing systems. The analysis involves not just the processor perspective, but the total system perspective, including hardware and software of database servers, storage subsystems, middle tiers and component connectivity. Our analysis shows that performance and price-performance trends of over two decades of TPC-C results exhibit close resemblance to Moore's predictions.

Acknowledgements

The authors would like to thank the past and present members of the TPC committee for their commitment in producing and maintaining the TPC-C benchmark specification, which continues to be the most popular yardstick for comparing transaction processing performance and cost for performance. A special thank is given to Charles Levine for providing the historical data used in the paper. The authors would also like to thank Satinder Sethi and Ray Glasstone for their support in writing this paper, and Brean Campbell, Tony Harvey, Ven Immani and Venkatesh Kamat for their valuable comments and feedback.

References

1. Moore, G.E.: Cramming more components onto Integrated Circuits. Electronics Magazine (April 19, 1965)
2. Moore, G.E.: Progress in digital integrated electronics. In: International Electron Devices Meeting, pp. 11–13 (1975)