Memory Expansion Technology (MXT™): Competitive Impact

T. Basil Smith, Bulent Abali, Dan Poff, R. Brett Tremaine

IBM T.J.Watson Research Center
P.O. Box 218, Yorktown Heights, NY 10598
{tbsmith@us.ibm.com}

Abstract

Memory Expansion Technology, (MXT) has been discussed in a number of forums. It is a hardware implemented means for software transparent on the fly compression of the content of a computer system's main memory. For a very broad set of workloads it provides 2:1 or better compression. This ability to compress and store data in a fewer number of bytes effectively doubles the apparent capacity of memory at minimal cost. While its clear that a doubling of memory at little cost is going to improve the price/performance of a system that can be offered to our customers, the magnitude or impact of MXT on price/performance has not been quantified. This paper estimates the range of price/performance improvements for typical workloads from available data. To summarize, the results indicate that MXT improves price/performance by 25% to 70%. The competitive impact of such a large step function in price/performance from a single technology are profound. In the competitive market for "PC Servers" this impact is comparable to the entire gross margins in this market.

Introduction

Memory Expansion Technology, (MXT) has been discussed in a number of forums [1, 2, 3, 4, 5, 6, 13]. It is a hardware implemented means for software transparent on the fly compression of the content of a computer systems main memory. For a very broad set of workloads it provides 2:1 or better compression [3,4]. This ability to compress and store data in a fewer number of bytes effectively doubles the apparent capacity of memory at minimal cost. Since memory cost is frequently the single most costly core component in server systems, its clear that a doubling of memory at little cost is going to improve the price/performance of a system. The magnitude or impact of MXT on price/performance has not been quantified or fully appreciated. This paper uses available data on workloads and pricing data to estimate the range of price/performance improvements. To summarize, the results indicate that MXT improves price/performance by 25% to 70%. The competitive impact of such a large step function in price/performance from a single technology are profound. No known alternate technologies or approaches exist that would allow such a large delta between two otherwise equivalent implementations using otherwise identical technology. To overcome the price/performance advantage with a simple pricing

---

adjustment would require the sacrifice of most if not all of the gross margins common to this market. You simply cannot compete profitably against this technology in this market.

Method for Quantifying the Price/Performance Impact of MXT

Extensive evidence exists, and is discussed elsewhere, that MXT can essentially double the capacity of an installed memory to hold instructions and data [2, 3, 4, 13]. It has also been extensively verified that the performance of the MXT memory doubled system is essentially identical to that of a stock system with twice the installed memory of the MXT system [3, 4]. For example, an MXT enabled system with 1 GB of installed memory will perform to within a few percent of a stock system with 2 GB of memory installed. Since memory is frequently the single most costly component of the core system, its is not surprising that this improves the price/performance of systems.

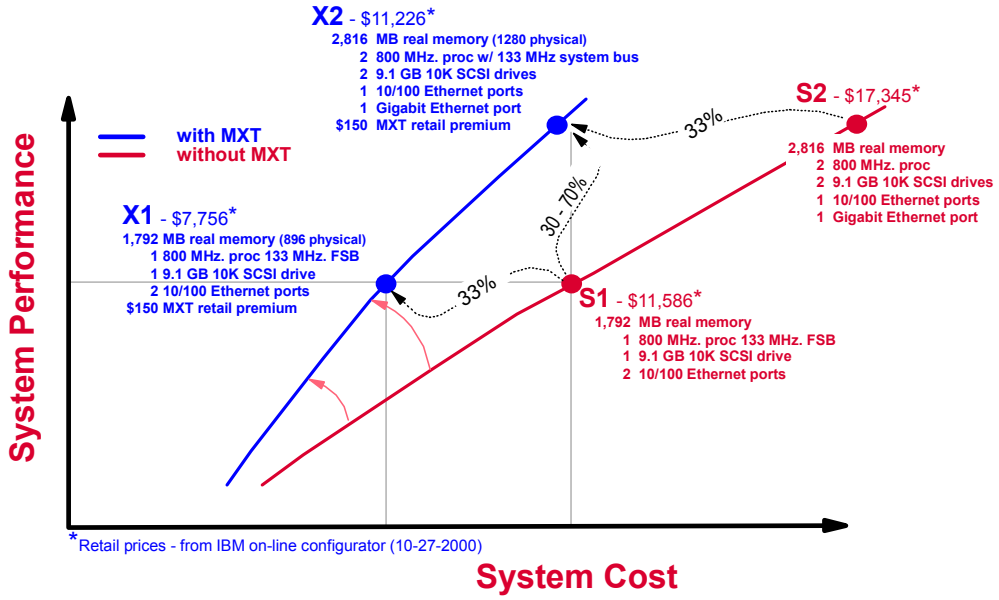
The method chosen is an attempt to quantify the improvement in the price/performance and the competitive impact on typically configured ***core server***s (the base system with configured processor and memory). The configured core was chosen as the relevant price/performance domain for several reasons. The most important of these is that the target "PC Server" market is now highly disaggregated. Full systems include many components which are separately purchased, either by systems integrators or by the end-user customer when he functions as his own systems integrator. The interfaces or boundaries between system components are standardized, making mix and match systems construction the norm in this market. Individual purchase decisions are thus made on a component basis, and all other factors being equal, it is the norm to purchase the component with the best price/performance.

For "PC Servers" the core server is a relevant customer purchase or decision domain, and consists of the base or entry system plus additional memory, additional processors, and/or optional I/O infrastructure upgrades which must be purchased with to base system to configure it for the workload. This excludes PCI or other I/O adapters, which can generally be purchased from a number of competing sources, but would include any I/O infrastructure augmentation, such as options for additional PCI buses that are specific to the base system and would need to be purchased as part of the base configuration. This also excludes disks and network components. Again, these components all compete in their own highly competitive submarkets. Retail price was chosen as the most convenient underlying metric in computing price/performance. Margins and discounts from retail are consistent enough across the industry to expect that the fundamentals and conclusion would remain the same if cost or discounted retail were used instead.

Using the above definition for price/performance, we then define the convenience concept of a "**performance twins**". Performance twins are two server systems which differ only by one being MXT enabled and the second being a stock or non-MXT enabled system but with twice the installed memory of the MXT enabled system. Other components of the systems are identical or equivalent. As noted above, evidence has been presented elsewhere that shows that two systems differentiated only in this fashion have essentially identical performance [3,4]. A similar concept, that of "**price twins**" also exists. Price twins are two systems each balanced for peak price/performance within the same cost constraint; one system is MXT enabled and the other is not. Price twins cost the same, but have different performance or throughput. They also may

have a somewhat different balance of resources (processor, memory and I/O) given the effective halving of the marginal effective cost of memory in an MXT enabled system.

The graph below illustrates these concepts:



In this drawing the S1 and X1 systems are real world examples of performance twins - systems which are very close clones of one another. The X1 system is MXT enabled and configured with 896 MB of physical memory that is expanded to appear as 1,792 MB. The S1 system is a stock system configured with 1,792 Mbs of memory, twice that of the MXT enabled system. These two performance twins system have nearly identical performance characteristics, despite the significant price difference between them.

In contrast, the X2 system is the MXT enabled price twin of the S1 system. These differ considerably in their performance and have a somewhat different balance of components, but are closely priced. The degree to which they differ in performance is strongly a function of workload, indeed the actual optimization of the component or resource balance in each system for peak price/performance is a function of workload.

The performance twin of the X2 system is the S2 system, in general most MXT configurations can be thought of as having both performance and price twins. That is the X2 system can be thought of as either a cost reduced version of S2 (performance twin), or a more capable same cost version of S1 (price twin).

The actual prices in the above example are 10-31-2000 prices for an IBM e-server xSeries 330™ pictured below. Prices were as published [16]. The xSeries 330 is a 1U (1.75") high, rack mount dense server, designed to be packaged with up to 42 servers in a single rack. Such dense packaging has strong market appeal, but illustrates another import point.

In principal its should always possible to find a performance twin for any configuration, but in practice, such a system may be impossible to configure. Because of its dense packaging constraints, an xSeries 330 cannot be configured past 4 GB of installed memory. An MXT enabled configuration exists with 8 GB of apparent memory (4 GB installed memory, expanded to 8 GB), but its performance twin with 8 GB of installed memory cannot be configured. The largest MXT configurations frequently fail to have real world performance twin. This ability to configure an MXT system beyond the largest stock configuration can provide real world performance advantages that are not fully captured in the analysis that follows. While current 32-bit software constraints may currently limit the ability to exploit memory beyond 4 GB, thereby limiting the MXT advantage of being able to configure really big system, software vendors, driven by the considerable performance advantages of increased memory sizes, are rapidly removing these 32-bit system constraints.

IBM ^ xSeries 330 1U Rack Mount Server

The concept of performance twins and price twins suggest two ways to measure the difference in price/performance that could be attributed to MXT. The most natural measure is using price twins. This can be thought of as similar to the increase in performance from simply turning MXT on without otherwise changing the configuration. Basically this is a direct answer to the

question; "Given a fixed number of dollars to spend how much extra performance do I get from MXT"?. Since the answer is very workload dependent the answer tends to cover a range of values for a range of workloads. For price twins the improvement in is measure is given as:

$$a = \frac{MXT_{throughput}/MXT_{price}}{STOCK_{throughput}/STOCK_{price}} - 1$$

Where MXT and STOCK prices are equal, yielding:

$$\alpha = \frac{MXT_{throughput}}{STOCK_{throughput}} - 1$$

For performance twins the most closely related metric is:

$$\beta = \frac{MXT_{throughput}/MXT_{price}}{STOCK_{throughput}/STOCK_{price}} - 1$$

Where MXT and STOCK throughputs are equal, yielding:

$$\beta = \frac{STOCK_{price}}{MXT_{price}} - 1$$

$\boldsymbol{\beta}$ is less sensitive to workload, as the performance of performance twins is nearly identical for a broad range of workloads. The impact of the extra memory for price twins is in contrast highly dependent upon workload and the base size of memory, making $\boldsymbol{\alpha}$ highly dependent upon workload.

Available Data and Workloads

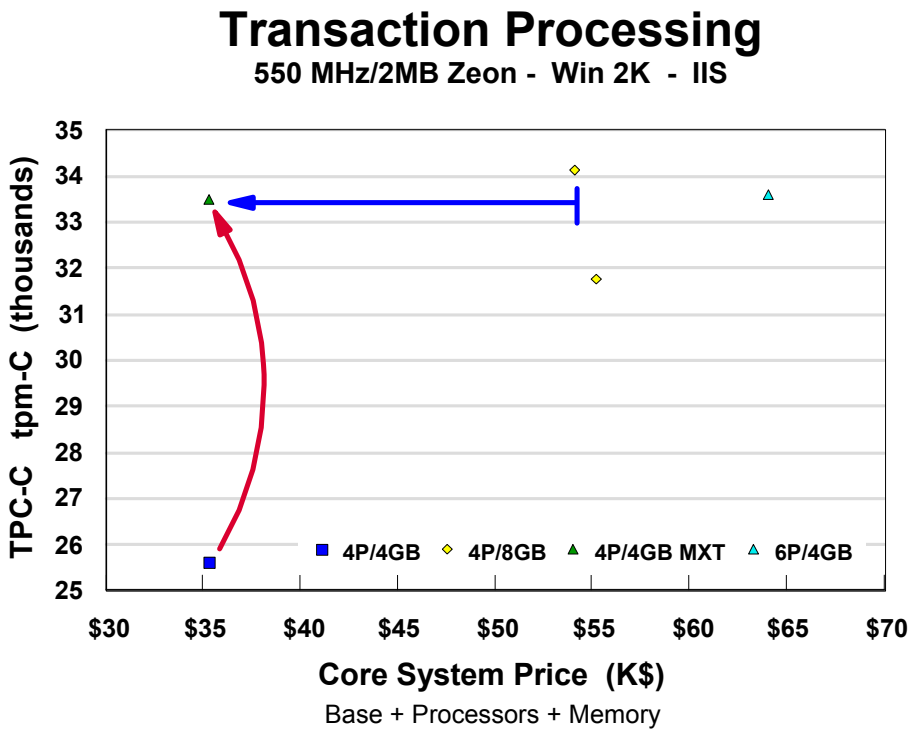
Coarsely, the price/performance impact of MXT technology can computed from real world data that provides insight on the performance differences that can be attributed to variations in memory size for representative configurations and workloads. As noted the general compressibility of workloads in general and the equivalence of physical memory in stock systems to apparent or expanded memory in MXT systems has already been established and documented [3,4].

Several benchmark data bases were examined, looking for insight from instances of where an industry significant workload or benchmark was run on similarly configured "PC Server"

systems, but with enough variations in memory sizes to suggest the competitive impact of MXT. Sets of data which met this criteria were found in the Transaction Processing Performance Council's TPC-C™ reported results, and for also reported results for SPECweb99™. The data comparisons are by virtue of this methodology only approximate.

Transaction Processing

The TPC-C results for the core "PC Server" components is presented in the graph below.



source: http://www.tpc.org October 12, 2000

Several similarly configured systems are shown. All systems were multiprocessors systems using 550 MHz Intel Xeon™ processors with 2 MB L2 caches, running Microsoft Windows 2000 Server™ and IIS 5.0™. Results were reported for a 4 processor, 4 GB's memory configuration; two 4 processor, 8 GB's memory configurations; and one 6 processor, 4 GB memory configuration. These results were also informally compared other configurations in the database to establish that all of the results reported here are within norms, and are not anomalous data points. A performance twin was then postulated; this is a 4 processor with 4 GB's of memory installed, that is MXT expanded to 8 GB's. This performance twin was assumed to be intermediate in performance between the two known stock 4 processor, 8 GB's memory configurations being reported. The core price of this performance twin was assumed to be the approximately the same as the 4 processor with 4 GB's system being reported. Price data is acquisition price for the core system components as reported in the TPC-C™ executive summary

for each reported result. Using this data it is then possible to infer approximate improvement in price/performance:

$$\alpha = \frac{MXT_{throughput}}{STOCK_{throughput}} - 1 = 31\%$$

$$\beta = \frac{STOCK_{price}}{MXT_{price}} - 1 = 56\%$$

These metrics suggest a 30% to 60% improvement in price performance, a staggeringly large number in the PC Server market.

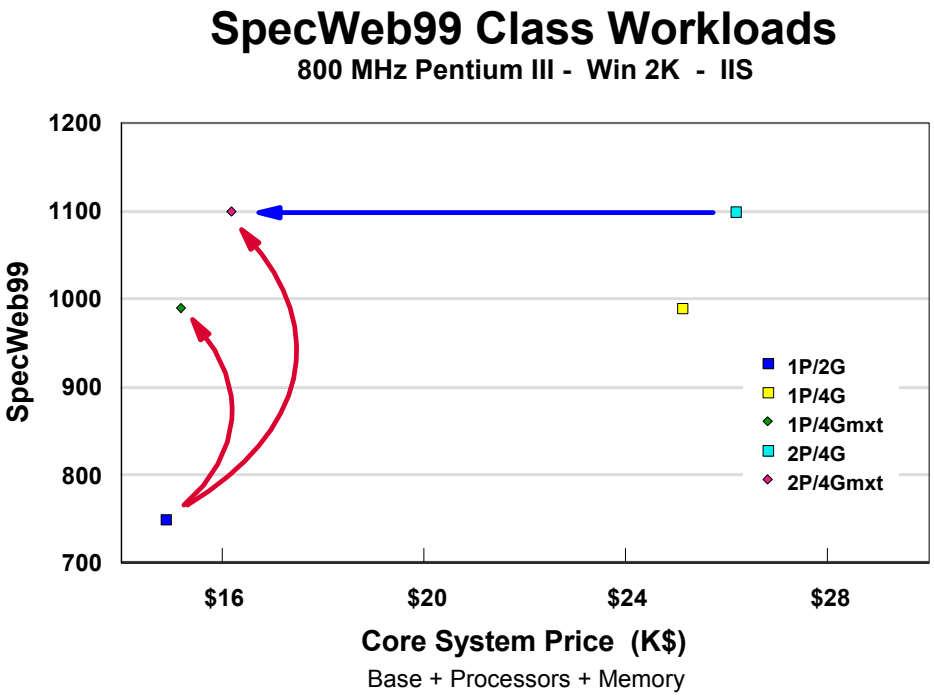
It is interesting to note that the reported results for a 6 processor, 4 GB memory configuration suggests that adding 4 GB's of memory to the 4 processor, 4 GB memory system is a more price effective means to improve performance than is the addition of 2 processors to this same system. Unfortunately, there were no reported results for a 6 processor, 8 GB's of memory configuration.

As an additional measure of price/performance benefit, all of the top 10 price/performance reported results for the TPC-C benchmark were examined. The impact on core system throughput/price for performance twins were constructed for each. The average improvement in throughput/price was:

$$\beta = \frac{STOCK_{price}}{MXT_{price}} - 1 = 33\%$$

Web Serving Workloads

Another set of reported results is for the Specweb99™ benchmark, a web serving applications. The base configurations for these systems are similar in character to IBM's e-Server xSeries 330 1U dense server or the comparable Compaq DL360™. Both systems are also quite similar to the MXT prototype [13]. Prices on all configurations are for a comparably configured 1U servers [16]. All systems were 800 MHz Pentium III™ processors, running Microsoft Windows 2000 Server™ and IIS 5.0™. The data from this database are marked below using square markers. There are published data points for a single processor with 2 GB's of system memory, a single processor with 4GB's of system memory, and a dual processor with 4 GB's of system memory. Two hypothetical price twins were constructed, a single processor with 2 GB memory expanded to 4 GB apparent., and a dual processor with 2 GB memory expanded to 4 GB apparent. Price twin prices were computed using the list price for a comparably configured stock product, assuming a retail $200 premium for MXT.



This results in price/performance metrics of:

$$\alpha = \frac{MXT_{performance}}{STOCK_{performance}} - 1 = 32\%$$

$$\beta = \frac{STOCK_{price}}{MXT_{price}} - 1 = 66\%$$

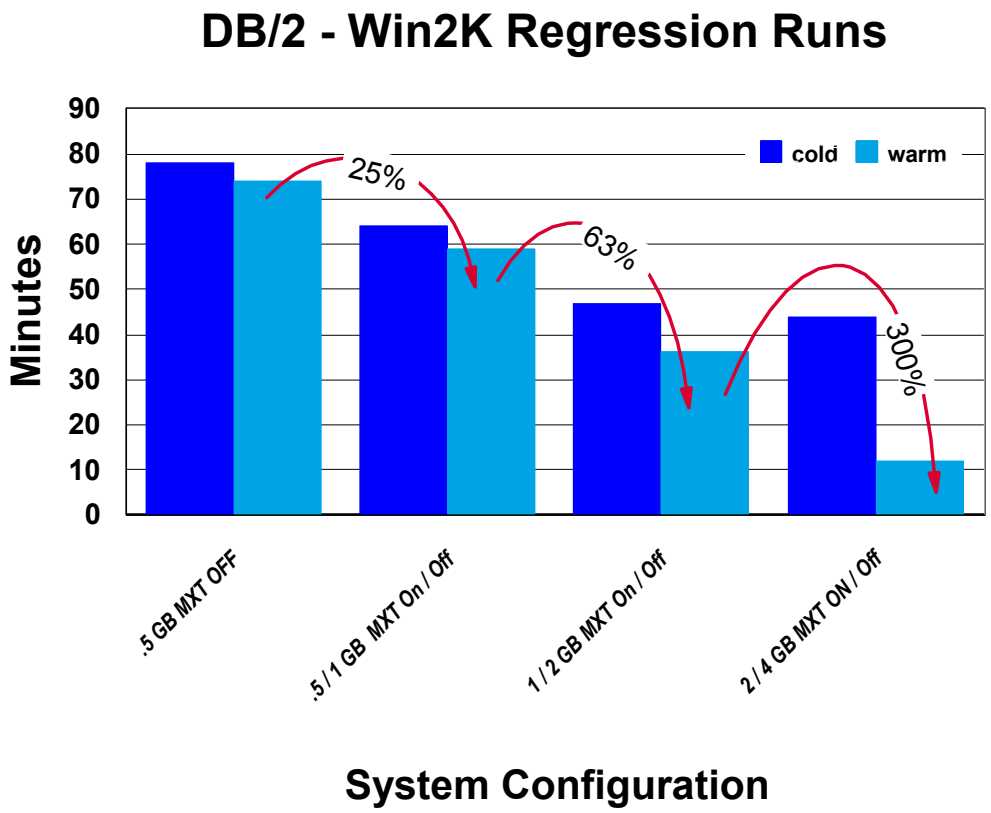
Again both metrics suggest a large (30% to 70%) improvement in price/performance.

It is interesting to note that in this case the purchase of an additional processor appears to be very cost efficient. Indeed the ideal configuration would appear to be a 2 processor 2 GB installed memory system expanded to 4 GB with MXT.

Measured Results

As a final result, the prototype MXT system has been measured running an extract of a commercial company's production database. This configuration is primarily used within IBM as a quick look regression test for ascertaining the impact of DB/2 design changes. It is substantially less costly and quicker to run than complex benchmarks such as TPC-C, and is another coarse indication of the general performance characteristics that might be expected. Several configurations were run on the prototype hardware: 512 MB with MXT off, the same 512 MB with MXT on (1 GB expanded), a 1 GB with MXT off configuration, and the same 1

GB with MXT on (2 GB expanded), and finally a 2 GB configuration with MXT off and on (4 GB expanded). Multiple runs were made for each configuration, a cold run where the DB/2 buffers are initially empty, and following that a warm run, where the DB/2 buffers had been "warmed" by the preceding cold run. The cold run give you an indication of the overhead in the initial load from disk. The warm runs illustrate the advantage of operating out of memory caches once they have been warmed. In all runs there was essentially no significant difference between runs with 2xN GB's of installed memory with MXT off or N GB's of installed memory with MXT on (e.g., the 1 GB system with MXT off performed exactly like the .5 GB system with MXT on). The arks below indicate the percent throughput gain for warm runs between when comparing the same system with MXT off and on (e.g., there is a 63% throughput increase when MXT is turned on for a system with .5 GB of physical memory installed - these are essentially comparisons between price twins).



For the warm run comparison between .5 GB/1 GB expanded MXT off/on price twins:

$$\alpha = \frac{MXT_{performance}}{STOCK_{performance}} - 1 = 25\%$$

Similarly the 1GB/2GB MXT off/on comparison is a comparison between price twins:

$$\alpha = \frac{MXT_{performance}}{STOCK_{performance}} - 1 = 63\%$$

Finally for the 2GB/4GB MXT off/on comparison price twins:

$$\alpha = \frac{MXT_{performance}}{STOCK_{performance}} - 1 = 300\%$$

It is interesting to note that the benefit of larger memory is are more pronounced for this workload for larger memory sizes, and is indicative that both the smaller 512 MB memory and 1 GB memory configurations are being memory starved.

For this workload, system price/performance is improved by close to 70%. The 2 GB memory size is close to the observed sweet spot for this class of dense servers.

Conclusion

Memory Expansion Technology, (MXT) has been discussed in a number of forums. It is a hardware implemented means for software transparent on the fly compression of the content of a computer system's main memory. For a very broad set of workloads it provides 2:1 or better compression. This ability to compress and store data in a fewer number of bytes effectively doubles the apparent capacity of memory at minimal cost. While its apparent that a doubling of memory a little cost is going to improve the price/performance of a system that can be offered to our customers, the magnitude or impact of MXT on price/performance has not been quantified nor fully appreciated. Available benchmark and workload data suggests that typical throughput for price performance improvements of from 30% to 70% can coarsely be expected. The competitive impact of such a large step function in price/performance from a single technology are profound. In the competitive market for "PC Servers" this impact is comparable to the entire gross margins in this market.

References:

[1] Arramreddy, S., Har, D., Mak, K., Smith, T.B., Tremaine, B., Wazlowski, M.: "IBM X-Press Memory Compression Technology Debuts in a ServerWorks NorthBridge," *HOT Chips 12 Symposiu*m, Aug.13-15, 2000.

[2] Abali, B., and Franke, H.: "Operating System Support for Fast Hardware Compression of Main Memory", Memory Wall Workshop, *Intl. Symposium on Computer Architecture (ISCA2000),* Vancouver, B.C., July 2000.

[3] Abali, B., Franke, H., Poff, D., Saccone, R., Herger, L., Smith, T.B., "Memory Expansion Technology (MXT): Software Support and Performance*",* Submitted to *IBM Journal of Research and Development*

[4] Abali, B., Franke, H., Poff, D., Smith, T.B., "Performance of Hardware Compressed Main Memory"

[5] Benveniste, C, Franaszek, P., Robinson, J.: Cache-Memory Interfaces in Compressed Memory Systems, Memory Wall Workshop, *Intl. Symposium on Computer Architecture (ISCA2000),* Vancouver, B.C., July 2000.

[6] Chen, J., Har D., Mak, K., Schulz, C., Tremaine, B., Wazlowski, M., "Reliability-Availability-Serviceability Characteristics of a Compressed-Memory System", *International Dependable Systems and Networks - 2000* (DSN-2000), June 2000, New York, New York.

[7] Franaszek, P, Robinson, J., Thomas, J. "Parallel Compression with cooperative dictionary construction," In *Proc. DCC'96 Data Compression Con*f., pp.200-209, IEEE 1996.

[8] Franaszek, P., Robinson, J., "Design and Analysis of Internal Organizations For Compressed Random Access Memory," IBM Research Report RC21146, Yorktown Heights, NY 10598.

[9] Franaszek, P., Heidelberger, Wazlowski, M.: "On Management of Free Space in Compressed Memory Systems", *Proceedings of the ACM Sigmetric*s, 1999.

[10] Franaszek, P., Heidelberger, Poff, D., Robinson, J., "Algorithms and Data Structures for Compressed Memory", *IBM Journal of Research and Development,* this issue.

[11] Hovis et al., "Compression architecture for system memory application," US Patent 5812817, 1998.

[12] Kjelso, M, Gooch, M., Jones, S.: "Empirical Study of Memory Data: Characteristics and Compressibility," In *IEEE Proceedings of Comput. Digit. Tec*h, Vol 45, No. 1, pp 63-67, IEEE, 1998.

[13] Tremaine, R.B., Smith, T.B., Wazlowski, M., IBM Memory eXpansion Technology (MXT), Submitted to *IBM Journal of Research and Development*

[14] Vahalia, U: "Unix Internals, The New Frontiers", Prentice Hall, ISBN 0-13-101908-2, 1996

[15] Wilson, P, Kaplan, S., Smaragdakis, Y.: "The Case for Compressed Caching in Virtual Memory Systems", *USENIX Annual Technical Conferenc*e, 1999.

[16] http://www.pc.ibm.com/eservers/xseries/ October 29, 2000

[17] http://www5.compaq.com/products/servers/platforms/ October 12, 2000.

[18] http://www.tpc.org October 27, 2000.

T. Basil Smith IBM Thomas J. Watson Research Center, Yorktown Heights, New York 10598 (tbsmith@us.ibm.com). Dr. Smith has been a member of the technical staff at IBM T. J. Watson Research Center since 1986 where he is now a Senior Manager responsible for research into exploitation of high leverage server innovations, and manages the Open Server Technology Department. . His work has been on memory hierarchy architecture, reliability, durability, and storage efficiency enhancements in advanced servers. He has received both IBM Outstanding Innovation Awards and Outstanding Technical Achievement Awards for his contributions in these fields at IBM. Previous to his joining IBM in 1986, he worked at United Technologies Mostek Corp. in Dallas and at the Charles Stark Draper Laboratory in Cambridge, Massachusetts. He holds over 20 patents in computer architecture and reliable machine design. Dr. Smith received his Ph.D. in computer systems, and his S.M. and S.B. from MIT. He is an IEEE Fellow and a member of the IEEE Computer Society Technical Committee on Fault-Tolerant Computing and active in that community. Most recently he was General Chair of the Dependable Systems and Networks Conference (DSN-2000) held in New York City, June 2000.

Bulent Abali IBM Thomas J. Watson Research Center, Yorktown Heights, New York 10598 (abali@us.ibm.com). Dr. Abali has been a Research Staff Member at IBM T. J. Watson Research Center since 1989, where he is now a manager responsible for system software and performance evaluation of advanced memory systems. He has contributed to numerous projects on parallel processing, high speed interconnects, and memory systems, including RS/6000 SP and MXT. Dr. Abali received his Ph.D. degree in electrical engineering from the Ohio State University.

Dan E. Poff, IBM Thomas J. Watson Research Center, Yorktown Heights, New York 10598 (poff@us.ibm.com). Mr. Poff is a System Programmer at the IBM TJ Watson Research Center, where he designs and developes MXT software compression controls. Before joining the Watson Research Center in 1982, he programmed logic chip testers at IBM East Fishkill, NY. At Research, he began with a group doing IBM's 1st port of Unix to the 1st Risc machine. Then joined two other people porting CMU's MACH to an early SMP Risc machine; subsequently, with one other person, ported MACH to RS/6000. In the early 90s joined a group porting Windows NT to IBMs PowerPC. He has received an Outstanding Technical Achievement Award. Mr. Poff received a MA degree in History and Philosophy of Science from Indiana University, 1969, and a BS degree in Physics from the University of Cincinnati, 1964. He has 5 patents pending and several publications, and he is a member of ACM.

R. Brett Tremaine IBM Thomas J. Watson Research Center, Yorktown Heights, New York 10598 (afton@us.ibm.com). Mr. Tremaine is a Senior Technical Staff Member at the IBM TJ Watson Research Center, where he is responsible for commercial server and memory hierarchy architecture, design and ASIC implementation. Before joining the Watson Research Center in 1989, he had been at IBM's Federal Systems Division in Owego, NY, since 1982. He has led several server architecture and ASIC design projects, many with interdivisional relationships, and he has received two Outstanding Technical Achievement Awards and several division awards for his contributions. Mr. Tremaine received a MS degree in computer engineering from Syracuse University in 1988, and a BS degree in electrical engineering from Michigan Technological University in 1982. He has 11 patents pending and several publications, and he is a member of the IEEE.