# Performance Benchmark Results For Automated Tape Library High Retrieval Rate Applications -Digital Check Image Retrievals-

John Gniewek
IBM Storage Systems Division
Tucson, Arizona

George Davidson
Check Solutions
Charlotte, NC

Bowen Caldwell
Check Solutions
Charlotte, NC

## Abstract

Benchmark tests have been designed and conducted for the purpose of evaluating the use of automated tape libraries in on-line digital check image retrieval applications. This type of application is representative of digital library applications requiring service to multiple concurrent users with service request response time identified as a very important metric. The benchmark test design was guided by a simplified theoretical queuing performance model developed to predict average response time as a function of request rate, object size retrieved, hardware characteristics, and library configuration. An exponential interarrival time distribution was used to simulate the multiple user environment, thus corresponding to an $M/E_k/c$ queue. A series of nine benchmark test runs, each consisting of nominally 1000 retrievals, was conducted for average request rates of 50, 100, 150, 200, and 250 requests per hour and for library configurations varying from two to six drives. The resulting experimental response time distributions were analyzed by fitting the data to a Shifted Erlang distribution function and also by determining the time for which the response corresponded to cumulative percentiles ranging from 10% to 95%. Good agreement was obtained between the modelled and experimentally determined average response time. For the hardware and software systems used in these tests, an average response time of approximately 30 seconds has been achieved under conditions of Poisson arrivals at average rates up to approximately 150 requests per hour. The use of timing trace data has identified the potential for further response time improvements.

## Introduction

The migration to digitally stored data from alternative media types such as paper and film has created opportunities for greatly increased data storage capacity. A very demanding application is check image storage and retrieval. At some of the larger centers, the requirements for long term archive will approach capacities of several hundred terabytes to a petabyte. The requirement for active 45-90 day storage will approach several tens of terabytes to 100 terabytes. However, high economical capacity is only one aspect of the demands. For the 45-90 day storage, this large capacity must also provide reasonable response times for image retrieval at high request rates. In order to evaluate the suitability of tape storage hardware for such applications, a performance benchmark test was developed to quantify response time performance for a variety of hardware configurations and request rate work loads. The experimental data are combined with queuing response time analysis to enable predictive performance projections for various configurations and operational characteristics of the tape storage hardware. This analysis thus allows end users and system integrators to select those applications where the appropriate tape storage hardware can meet the required performance targets with the economical advantages of lower cost tape storage.

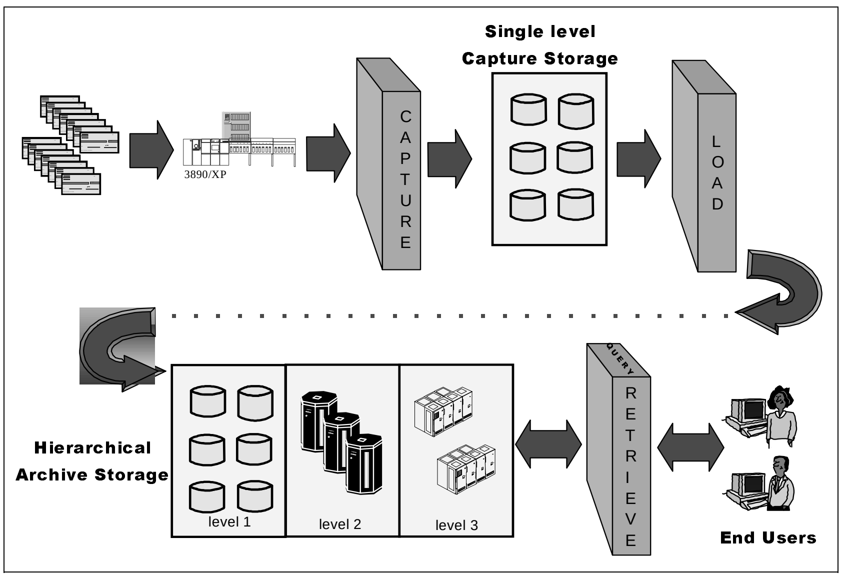
## System description

The hardware and software components used in these benchmark studies are all commercially available and part of an integrated solution [1] developed for check image storage and retrieval applications within the banking industry. The processor used was one node of an IBM multinode RS/6000 Scalable POWERParallel (SP) system. Application software combines Check Solutions' Check Image Enterprise Archive (CIEA) custom code with IBM's On-Demand for AIX, DATABASE 2-Universal Data Base (DB2-UDB), and ADSTAR Distributed Storage Manager (ADSM). Tape storage hardware consisted of an IBM 3575 L18 library with up to six 3570 Model C drives.

This paper concentrates on the random retrieval of images, which is only one of several applications required for the total solution; other applications include check image capture, loading the archive, and bulk image retrieval. Capture involves optically scanning the paper checks at high speed while they are being sorted.

Compressed images of front and back views of the checks are captured and stored as objects on magnetic disk. It is from this Capture Storage that the multi-level Archive Storage hierarchy is created. This first level is typically directed to magnetic disk systems to provide for image retrievals within one to two seconds. Thus, there exists a 'load' operation where images are moved from Capture Storage to the active first level of Archive Storage. Because of the very large capacities required, a reduced demand on retrievals over time, and economical reasons, two levels of tape storage are defined behind the first level disk storage. Traditional tape processing operations including back-up, restore, and migration from first level disk (typically 3-7 day lifetime) to second level fast access tape (30-90 day lifetime) to long term archive tape (7 year lifetime) are also employed. To provide economical scaling and modular growth, a multi-node processor system is used with high-speed interconnection. For larger installations, several million objects per day are introduced into the system. The benchmark tests reported here were designed to evaluate the response time retrieval performance of the fast access tape storage layer (level 2) under conditions simulating a multi-user customer environment. Figure 1 illustrates the Check Image Enterprise Archive system.

Figure 1. Check Image Capture, Storage and Retrieval System.




## Performance benchmark definition

### Desired output

The desired output for the benchmark tests is the response time distribution as a function of hardware configuration and request rates in an environment simulating a large number of users randomly accessing a common automated tape data storage library. The multi-user simulation thus indicated the use of a Poisson process for the arrival of requests. This corresponds to an exponential distribution for the request interarrival times. In order to construct the random spatial nature of where the image objects were located, ten 5000MB cartridges were written to capacity with 26KB image objects. The cartridges were then placed at various strategic locations in the 180 cell library to represent different distances from cartridge slot to the drive locations. Within the cartridge, 100 locations were indexed starting with 1 at the logical beginning of tape and 100 at the logical end of tape. Thus, each retrieval request was defined by specifying three parameters: (A) a time, defined by adding a time interval determined by a random number chosen from the exponential distribution of interarrival times; (B) a library slot location, determined by the cyclic identification of the cartridge number from 1 to 10; and (C) a within cartridge location, determined by a random number between 1 and 100 chosen from a uniform distribution. An example of the retrieval pattern generated for 15 requests (out of a sequence of 1000 requests) and the measured response time is shown in Table 1. This example corresponded to an average request rate of 150 per hour with 6 or 4 drives in the library. The response time includes the complete operation of unloading a cartridge from the drive (if necessary), getting and loading a new cartridge, searching to the object, and reading and sending the object back to the requester. Rewind occurs automatically at the completion of the service request and is not included in the response time. It does, however, factor into the queuing delay if all drives are occupied. The test sequence was defined such that there was only one request serviced per cartridge mount, that is there were no second hits to a mounted cartridge.

### Early learning

Initial performance test results compared with predictions from a queuing response time model [2] were less than expected. Timing traces were then obtained at the system level to identify delay components of the response time for each retrieval. This learning resulted in modifications introduced to both the data management software and the device driver code to improve the overall response time at the system level. In addition, at the component level, firmware changes were introduced with the C drive to also improve response time. All of these changes were incorporated in the benchmark tests reported in this paper. The code changes provided substantial system level performance improvement as measured by the response time distribution. Additionally, following the software modifications, good agreement was obtained between the experimental results and the queuing performance model. Delays due to system and application software overhead were incorporated into the queuing model input parameters noted later.

Table 1. A sequence of 15 retrieval requests illustrating the combination of a random location within the cartridge (1-100) chosen from a uniform distribution function with a random interarrival time chosen from an exponential distribution function. Response time results are from 150 requests per hour with either 6 or 4 C drives.

| Event | Tape Location | Interarrival Time | 6 Drives | 4 Drives |
|---|---|---|---|---|
| 1 | 9 | 0 | 30 | 26 |
| 2 | 4 | 45 | 37 | 33 |
| 3 | 2 | 7 | 25 | 29 |
| 4 | 70 | 18 | 32 | 29 |
| 5 | 55 | 40 | 28 | 23 |
| 6 | 27 | 13 | 32 | 32 |
| 7 | 26 | 21 | 35 | 35 |
| 8 | 13 | 5 | 33 | 37 |
| 9 | 38 | 11 | 34 | 34 |
| 10 | 79 | 4 | 43 | 40 |
| 11 | 17 | 51 | 35 | 32 |
| 12 | 36 | 8 | 27 | 26 |
| 13 | 69 | 16 | 26 | 25 |
| 14 | 94 | 1 | 29 | 37 |
| 15 | 97 | 4 | 40 | 30 |

**System level optimization is required**

Figure 2 shows an example of the response time distribution for the experimental conditions of 150 requests per hour with a configuration of 4 drives in the library. The mean and variance of the response time for this test case improved from 48 seconds and 455.8 to 37.5 seconds and 254.8 when the new data management software changes were introduced. These performance improvements were measured with the identical hardware configuration. The data management code influences the drive query status and also the sequence by which the drive 'unload' and robot 'acquire' operations were handled. Additionally, device driver code changes were introduced earlier. Those changes resulted in more frequent queries of drive 'ready' status to effectively reduce the drive 'load-to-ready' time. Trace data were used to identify the request sequences with long response times. After making the code changes, the remaining identified causes included closely batched request arrival times, all drives busy during this time, and a drive going into an automatic cleaning cycle. The majority of the long times associated with the 150-4 test case (average request rate = 150 per hour; 4 drives) resulted from 5 request sequences that ranged in

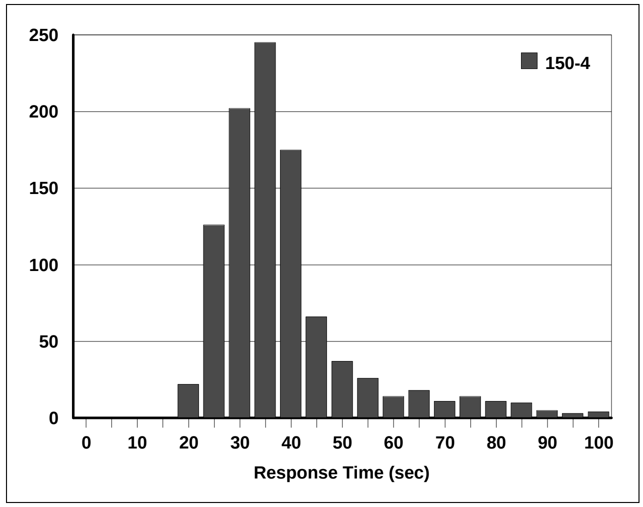



Figure 2. Test case 150-4. Histogram of response time distribution.

length from 14 to 18 requests at an effective rate ranging from 280 to 475 requests per hour. Such events are to be expected in normal mutiple user production environments and must be taken into account in configuring the system to meet performance expectations.

**Test conditions**

Nine separate performance benchmark tests were conducted. Test cases are designated by the number of requests per hour and the number of drives in the automated library. Thus, test case 150-6 corresponds to a mean request rate of 150 per hour with 6 drives in the library. Each test consisted of nominally 1000 retrievals, although there were a few instances where this number was reduced slightly as a result of known operational anomalies. In no case was the number less than 960 retrievals. The nine test cases reported here are: 50-2, 50-4, 50-6, 100-4, 100-6, 150-4, 150-6, 200-6, and 250-6. These test cases correspond to an average utilization of the robot ranging from 0.14 at 50 mounts per hour (50 MPH) to 0.69 (250 MPH), and an average drive utilization ranging from 0.12 (50-6) to 0.62 (250-6). The second highest drive utilization value is for test case 150-4 at 0.56. The average reponse times are ordered according to the queuing delays corresponding to these utililization factors.

## Data Analysis

The arrival time distribution is defined as a Poisson process. Hence, this corresponds to an M type in queuing theory nomenclature [3]. The number of servers, (c), is defined as 1 for the robot, and 2, 4, or 6 for the number of drives. The service time distribution may, in general, be represented by an Erlang distribution, $E_k$, where k represents the number of phases. For k = 1, the service time distribution would be exponential. For k approaching infinity, the distribution approaches a constant with ever-decreasing variance. For an M/M/c type queue the average response time may be expressed analytically. However, exact analytical expressions for the response time distribution are not available.

The nine different data sets were analyzed in two ways. One method compares the experimental mean response time to the calculated mean response time from the simplified queuing theory model. This is useful for assessing the capability of the model to make projections beyond the configurations that were experimentally measured. The second method empirically fits the experimental data to a shifted Erlang distribution function, analogous to a shifted exponential distribution [4], with Erlang parameters, k, and $\lambda$, and a shift factor, F. F is

$$f(t) = \frac{(k\cdot\lambda)^k}{(k-1)!}\cdot(t-F)^{k-1}\cdot e^{-k\cdot\lambda\cdot(t-F)}$$

Shifted Erlang Distribution Function

given in seconds and enters the equation by replacing (t) with (t-F). For each of the 9 conditions, the matched distribution function is thus characterized by (k, $\lambda$, F). Figures 3 through 5 show the results. The equation used is shown subject to the constraints that t > F and k is an integer.

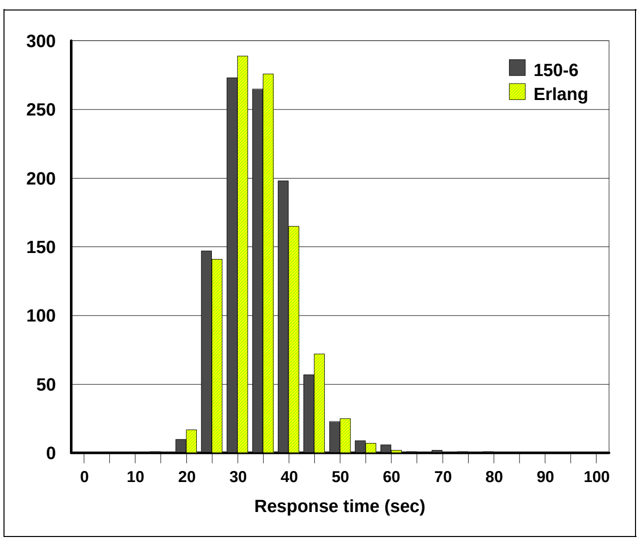



Figure 3. Response time distribution. Test Case 150-6.

Included in the graphs of the response time distribution are histograms of the experimental data together with the distribution obtained by a least mean squares best fit to the Shifted Erlang distribution with variables F, k, and $\lambda$. The histogram is constructed with 5 second bins. The times are designated so that the bar at a given time represents the number of events that occurred between that time and 5 seconds less. For example, in Figure 4, the bar showing 318 events at 30 seconds represents the number of responses that occurred within the five second interval from 25 to 30 seconds. A comparison of Figure 2 with Figure 3 shows the improved response times with an increase in the number of drives from 4 to 6 for a request rate of 150 per second.

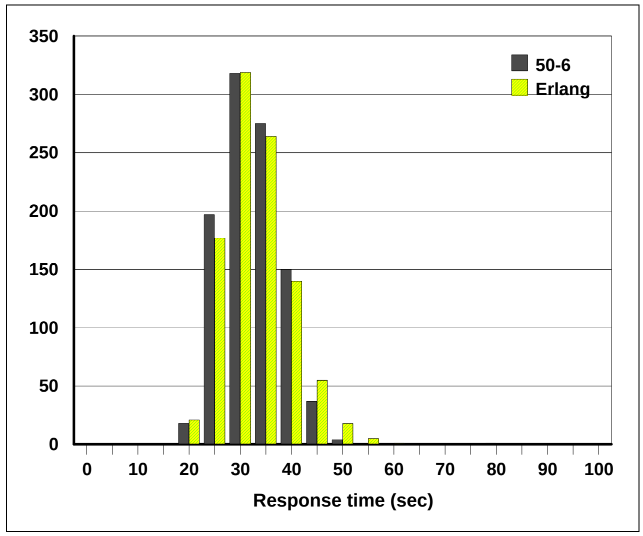



Figure 4. Response time distribution. Test case 50-6.

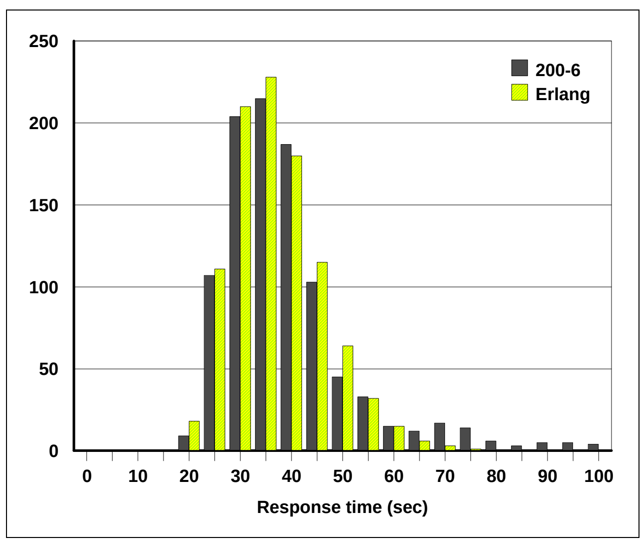



Figure 5. Response time distribution. Test case 200-6.

A summary of the best fit Erlang function parameters for all test cases is given in Table 2. For high values of k, several sets of parameters yield only small differences in the least mean square values. The optimum set is shown.

Table 2. Summary of best fit Erlang distribution parameters for each of nine test cases. $\lambda$ is calculated as a mean (FM) for direct comparison to the experimental mean (EM). See text.

| CASE | F | k | FM | EM |
|---|---|---|---|---|
| 50-2 | 15 | 5 | 32 | 34.6 |
| 50-4 | 9 | 12 | 31 | 30.6 |
| 50-6 | 13 | 8 | 31 | 30.4 |
| 100-4 | 10 | 12 | 31 | 31.8 |
| 100-6 | 6 | 16 | 31 | 31.2 |
| 150-4 | 13 | 6 | 34 | 37.5 |
| 150-6 | 11 | 10 | 32 | 32.5 |
| 200-6 | 13 | 6 | 35 | 38.6 |
| 250-6 | 20 | 2 | 43 | 47.3 |

The value of $\lambda$ for the Erlang distribution may be calculated from the values FM and F as $\lambda$=1/(FM-F). The use of the value FM in this manner thus allows direct comparison to the experimentally determined mean value, EM. Units for F, FM, and EM are in seconds. The Erlang constant, k, is dimensionless.

**Comparison with analytical queuing model**

A simplified queuing response time model [2] has been developed in Mathcad 6.0. The model is analytical and provides estimates of average response times only. It has been used to provide estimates for many conditions of configurations, application requirements, and hardware characteristics. The value of such a model, even though approximate, has been illustrated by the improvements made to the code when initial experimental results failed to meet modelled expectations. Additionally, it provides a means to predict performance for significantly different digital library applications. The experimental results described here provide an opportunity to compare the model with experiment and 'tune' the model empirically by adjustment of some of the queuing delay parameters. A comparison of the predicted and measured average response time for each of the test conditions is shown in Table 3. The hardware parameters used as input to the model are shown in Table 4.

Table 3. Comparison of experimentally determined mean response time, EM, and queuing model predicted mean response time, QM. Time in seconds.

| CASE | EM | QM |
|---|---|---|
| 50-2 | 34.6 | 36 |
| 50-4 | 30.6 | 30 |
| 50-6 | 30.4 | 30 |
| 100-4 | 31.8 | 32 |
| 100-6 | 31.2 | 31 |
| 150-4 | 37.5 | 39 |
| 150-6 | 32.5 | 33 |
| 200-6 | 38.6 | 37 |
| 250-6 | 47.3 | 51 |

Table 4. Hardware parameters used in queuing response time performance model. *11 seconds includes 3.5 seconds system and software delays (hardware component alone is 7.5 seconds). **13 seconds corresponds to average for all tape locations including 1 second system overhead (9 seconds corresponds to the average for cartridge filled to ½ capacity).

| OPERATION | MEAN SERVICE TIME (SECONDS) |
|---|---|
| Drive Unload | 6 |
| Robot Put | 5 |
| Robot Get | 5 |
| Drive Load | 11 (7.5)* |
| Search/Read | 13 (9)** |
| Rewind | 13 |

The analytical queuing model is only capable of estimating the mean response time. For production environments, the response time distribution is also important. The experimentally derived distributions are obtained for a single interarrival time distribution for a given average request rate. Thus, the 50-2, 50-4, and 50-6 test cases used the same interarrival time distribution function. However, an independent randomly generated interarrival time distribution is used for each of the other mean request rates. Comparisons across different mean request rate test cases should keep in mind the statistical variations that could exist for a retrieval sample size of only 1000 requests. As mentioned previously, for an average request rate of 150/hour, several burst sequences were found with effective request rates between 280-475/hour for durations between 14-18 requests.

**Response time distribution**

The experimental response time distribution may be represented by plotting the time for which a given percent of the requests are serviced. A plot showing results for several of the test cases is shown in Figure 6. While the median (50 percentile) shows only minor variations among these chosen cases, the divergence is greater at the higher percentile values. User satisfaction will be determined by not only the average response time, but by the variance observed. The mean, median, and variance of the response times for each of the test cases are shown in Table 5.

| TEST CASE | MEAN | MEDIAN | VARIANCE |
|---|---|---|---|
| 50-2 | 34.6 | 31 | 192.6 |
| 50-4 | 30.6 | 30 | 34 |
| 50-6 | 30.4 | 30 | 33.7 |
| 100-4 | 31.8 | 31 | 53.6 |
| 100-6 | 31.2 | 31 | 34.7 |
| 150-4 | 37.5 | 34 | 254.8 |
| 150-6 | 32.5 | 32 | 52.8 |
| 200-6 | 38.6 | 35 | 278.1 |
| 250-6 | 47.3 | 39 | 576.3 |

Table 5. Statistical parameters associated with each of the test cases.

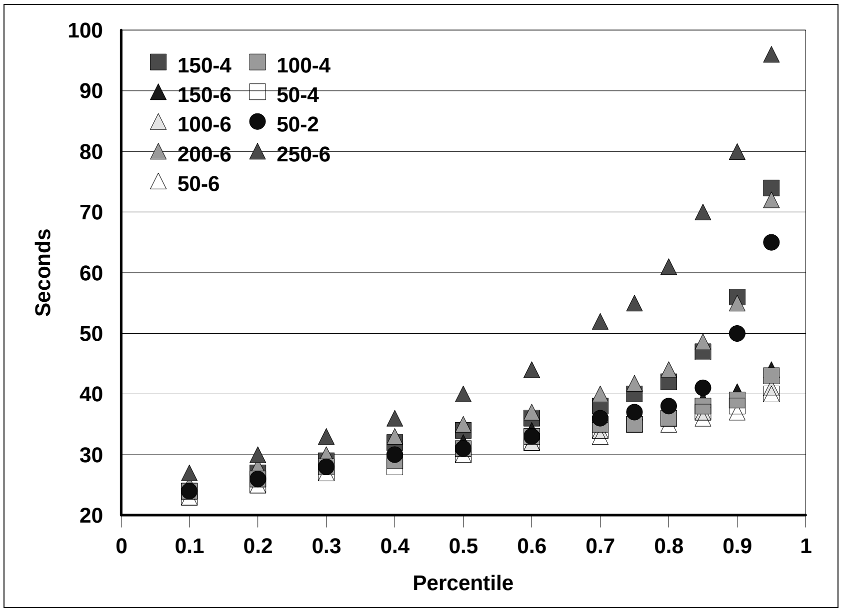



Figure 6. Response time for various cumulative percenti le values.

## Discussion

The response time dependence on work load in a multi-user environment is complex and highly non-linear. Additionally, there are non-linearities and unique behaviors in characteristics such as search time and rewind time [5] associated with different types of tape storage devices. The optimization of retrieving multiple files from a cartridge (that is no robot move invoked) for one such device was reported by Hillyer and Silberschatz [6]. Lake [7] described the system level tuning involving software and hardware that was required to improve the throughput performance of a large data ingest facility. The work reported in this paper echos the need for system level tuning albeit in a retrieval application.

The storage hierarchy chosen is dependent upon the specific requirements of the application. For the check image storage and retrieval application described here, the combination of high capacity and high retrieval rates dictates a three level hierarchy: disk, fast response tape, and archive tape. Even though the percentage of objects retrieved from tape tends to be low, < 0.6%, the absolute numbers are quite high. For example, a bank that archives 10 million checks per day may expect requests for 60,000 of these during their lifetime, and on a given day retrievals may number between 15,000 and 20,000 from the total tape archive. The economic trade-offs are dynamic as disk and tape storage costs decrease with improved technology and as new devices emerge with improved characteristics developed for these kinds of demanding applications.

The performance benchmark tests designed, conducted, and reported on here are a practical means to measure how well different types of devices may operate in a production environment under 'loaded' conditions. At the system level, interaction of the hardware with the software may sometimes reveal delays not anticipated and provide the motivation to tune the system for improved performance. The availablitity of a predictive performance model provides the means to monitor the experimental results of the current test, and then apply the results to other configurations and other applications.

## Conclusions

The benchmark tests described here may be used to provide a common set of conditions for evaluating diverse hardware configurations and characteristics. Use of the identical random number distributions for the exponential interarrival time and the uniform tape location sequences would enable comparison with sequence lengths shorter than 1000 retrievals. Alternatively, independent random number sequences may be used provided the number of retrievals used is on the order of 1000.

The use of a performance model in combination with the experimentally obtained response times and timing trace data has resulted in improved performance characteristics. These test results have demonstrated the capability of currently available tape library systems to provide average response times of about 30 seconds under high rate random retrieval applications. Several areas where further improvements may be possible have been identified. It is expected that products under development will incorporate additional features that will further enable the use of automated tape library storage systems in this type of digital library application.

## Acknowledgements

J.Ayres, F.Baca, C.Bell, L.Durazo, J.Hammar, L.Jesionowski, R.Kamens, J.Koski, A.Syed, F.Tanner, J.Thomas, and W.Wilson assisted in the completion of these tests.

## References

[1] Check Image Enterprise Archive Solution: Introduction and Planning Guide, Document Number GC31-4023, Program Number: 5799-QJX, 5799-FZQ, 5799-FZR, 1998. (http://www.checksol.com)

[2] J. Gniewek, "Evolving Requirements for Magnetic Tape Data Storage Systems", NASA Conference Publication 3340. Fifth NASA Goddard Conference on Mass Storage Systems and Technologies, pp.477-491, September, 1996.

[3] J. White, J. Schmidt, and G. Bennett, "Analysis of Queuing Systems", Academic Press, New York (1975).

[4] H.C. Tijms, "Stochastic Modelling and Analysis: A Computational Approach", John Wiley & Sons, Great Britain (1986).

[5] T. Johnson and E.L. Miller, "Benchmarking Tape System Performance", NASA Conference Publication GSFC-1998-206850, Sixth Goddard Conference on Mass Storage Systems and Technologies, pp. 95-112, March, 1998.

[6] B.K. Hillyer and A. Silberschatz, "Scheduling Non-Contiguous Tape Retrievals", NASA Conference Publication GSFC-1998-206850, Sixth Goddard Conference on Mass Storage Systems and Technologies, pp. 113-123, March, 1998.

[7] A. Lake, "Performance Tuning of a High Capacity/High Performance Archive for the Earth Observing Systems Project", NASA Conference Publication GSFC-1998-206850, Sixth Goddard Conference on Mass Storage Systems and Technologies, pp. 373-383, March, 1998.