

# A Mathematical Framework for Modeling and Analyzing Migration Time

Feng Luan

Dept. of Computer and  
Information Science,  
Norwegian University of Science  
and Technology,  
Trondheim, Norway  
+47 735 91457

luan@idi.ntnu.no

Mads Nygård

Dept. of Computer and  
Information Science,  
Norwegian University of Science  
and Technology,  
Trondheim, Norway  
+47 735 93470

mads@idi.ntnu.no

Thomas Mestl

Research and Innovation  
Det Norske Veritas (DNV),  
Høvik, Norway  
+ 47 67579900

thomas.mestl@dnv.com

## ABSTRACT

File format obsolescence has so far been considered the major risk in long-term storage of digital objects. There are, however, growing indications that file transfer may be a real threat as the migration time, i.e., the time required to migrate Petabytes of data, may easily spend years. However, hardware support is usually limited to 3-4 years and a situation can emerge when a new migration has to be started although the previous one is still not finished yet. This paper chooses a process modeling approach to obtain estimates of upper and lower bounds for the required migration time. The advantage is that information about potential bottlenecks can be acquired. Our theoretical considerations are validated by migration tests at the National Library of Norway (NB) as well as at our department.

## Categories and Subject Descriptors

C.4 [Performance of system]

H.3.2 [Information Storage]

H.3.7 [Digital Library]

## General Terms

Management, Measurement, Design,

## Keywords

Long-term preservation, Migration, Storage, Process modeling, Performance

## 1. INTRODUCTION

The rapid development of and the acceptance in the mass market of digital capturing devices such as PDA, smart phones, digital cameras, digital video cameras, mp3 players, together with increased usage of computers and digital sensors in industry has

lead to what is commonly called an “information explosion”. Gantz et al. [1] calculated that in 2006 a staggering 161 billion GB of new information was stored. The total amount of digital data worldwide was about 281 Exabyte<sup>1</sup> in 2007, with an annual growth rate of 60%. They predicted that this would be 6-folded in 2010. If this trend continues, then 10 years later (by 2020) the amount of data generated annually will be 200 times higher than in 2008, about 70 Zettabyte.

The majority of these data, i.e., 70%, belongs to individuals and is stored de-centralized on private computers, memory cards, tapes, mp3-players, etc. Certain industries such as oil companies (seismic data), biotech companies (protein & DNA sequences), data centers (e.g., Google) as well as public institutions (e.g., libraries), will experience a similar explosion-like growth in their amount of data. For instance, in 2007 the National Library of Norway (NB) holds about 1 Petabyte of unique data (the actual number is 3 times as higher due to two additional backup copies) with an annual growth rate of 750 Terabyte. All library items (books, maps, letters, photographs, films, radio, television, and web hosting) are to be digitized - so far only 5% has been converted. At completion in 2018, the expected required storage capacity will have to be at least 37 Petabyte. This number will however not contain all the new material that will be submitted in the mean time and the total volume of unique data will rather be closer to 50 Petabyte in 2018.

Various preservation strategies [2-6] have been suggested: computer museum, emulation, encapsulation, UVC, and migration. Interestingly, there is a significant concern that all these data may become inaccessible over time as the file formats become obsolescent, e.g., [7-11]. However, at present it seems that file format obsolescence is a less important issue than hardware obsolescence – especially storage hardware. In the case of NB, every 3-4 years they have to transfer all their data from their old storage system to a new one. The reason is not hardware breakdown or the need of higher performance - but simply the limited support time offered by the hardware vendors - usually restricted to 3 or 4 years. According to the latest experience of NB, when the old storage system was replaced, it took more than three months to copy all data into the new storage system,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

JCDL '10, June 21–25, 2010, Gold Coast, Queensland, Australia.  
Copyright 2010 ACM 978-1-4503-0085-8/10/06...\$10.00.

<sup>1</sup> 1 Zettabyte = 1024 Exabyte = 1024<sup>2</sup> Petabyte = 1024<sup>3</sup> Terabyte = 1024<sup>4</sup> Gigabyte = 1024<sup>7</sup> Byte

constituted by one disk array with RAID5 and two tape robotic systems. As the amount of digital information rises exponentially, there is a real concern that the time required for the next system replacement will exceed by far the support time, i.e., the previous copying process of all data is still not finished before the next has to be initiated.

Such huge volume of preserved data will affect a number of aspects, such as the preservation strategy, hardware performance, software performance and even storage architecture. The expected data migration time could be a handy measure for comparing different technology choices, modifications of systems, or variations on system designs.

The objective of this paper is to address the time challenge in the migration of huge volumes of data. In the following, we will construct process models for migration, as well as derive mathematical equations for estimation of the upper and lower bounds of the required migration time. The theoretical approach will be evaluated by migrating data in two experimental settings.

This paper is structured as followed: the next section addresses the concept of migration. Section 3 describes the main parameters involved in migration. In Section 4, mathematical equations for each stage in the migration process are derived. In Section 5, two evaluation experiments are carried out in order to compare the theoretical results with real migration measurements and we end with some comments and discussions in Section 6.

## 2. WHAT IS MIGRATION?

There is some confusion and a clear lack of precise definition of what is actually meant by migration. Throughout this paper, we use a general definition of Garrett [12], i.e., [Migration is] "... a set of organized tasks designed to achieve the periodic transfer of digital materials from one hardware/software configuration to another, or from one generation of computer technology to a subsequent generation".

Others definitions can be found in Wheatley et al. [13] and OAIS [14]. Wheatley et al. broke migration into six specific cases: 1) Minimum Preservation: preserve the bits of the original digital objects; 2) Minimum Migration: convert the original digital objects to very general formats requiring little technical work; 3) Preservation Migration: preserve digital objects in the same formats as minimum migration and some annotations, e.g., screenshots, comments to the screenshots, and video clips; 4) Recreation: replace the original object with a new object; 5) Human Conversion Migration: recreate all objects derived from that original object by manual operations; 6) Automatic Conversion Migration: convert digital objects to ones suitable for the current environment.

On the other hand, OAIS separates migration into four categories: Refreshment, Replication, Repackaging and Transformation. Refreshment is to copy data from one digital storage medium to another without changing the original bit stream (also called media migration), whilst replication is to preserve many copies in geographic distributed places. Repackaging will modify the structure of digital records whereas transformation is the same as format conversion (or format migration), where the file's format is changed.

There are five main reasons for initiating migration:

**Failure of digital storage media.** Storage media, especially with moving parts, e.g., hard disk drive, are sensitive to the environment, such as temperature, humidity, shake and use frequency. They may gradually evolve towards an unacceptable error level. Crespo & Garcia-Molina [15] believe that for long-term storage systems, higher reliable storage media can effectively reduce the cost of preservation and increase the reliability of data.

**Hardware/software obsolescence.** Hardware and software of a preservation system have become so old that vendors will no longer offer support. Although the system would still work perfectly for many years, it has to be replaced for assuring continual warranty or support. This is the main reason why NB replaces its digital storage system every 3-4 years.

**Performance obsolescence.** The latest hardware or software is more sophisticated, i.e., faster processing speed and higher reliability, or frees floor space by more compact hardware. It often offers advanced management techniques and better performance, or it overcomes logic failures in the old system. Both of them can efficiently reduce the maintenance cost, increase reliability of digital information and speed up the access speed.

**A change in requirements of digital repositories.** The owners of a digital repository may have to change the configuration of a digital repository due to new laws or regulations, change of business model or due to cost reduction requirements.

**Changing hardware/software vendors.** The cooperation between owners of digital repositories and hardware/software vendors may also change. Normally, each vendor uses its proprietary product, which may be incompatible with other vendors' products.

Whatever reason there are for replacing the storage hardware or software, it is generally wise to prepare a migration event in detail. A general migration process is illustrated in Figure 1.

## 3. MIGRATION PARAMETERS

This paper aims at estimating the required migration time for a given amount of data based on specified technical constraints, i.e., file sizes, read/write bandwidth, processing speed, network transfer bandwidth, and the number of replicas. Table 1 lists the parameters for calculating migration time.

**Table 1. Migration Parameters**

Parameters	Abbreviation
Average size of a single digital record	$x$
Average read or write transfer bandwidth	$B_R$ or $B_W$
Average file processing speed	$B_P$
Average network transfer bandwidth	$B_N$
The number of replicas	$K$

**Average size of a single digital record** ( $x$  in MB): A digital repository holds thousands or millions of files with varying sizes, e.g., image or video files can easily be many Mega- or Gigabytes, whereas plain text files are typically only Kilobytes. The total volume of all files is denoted by  $S$  and the total number of files  $N$ .

**Average read or write transfer bandwidth** ( $B_R$  or  $B_W$  in MB/s): The read transfer bandwidth  $B_R$  is the bandwidth of transferring data from the old storage to its host, while  $B_W$  is the write

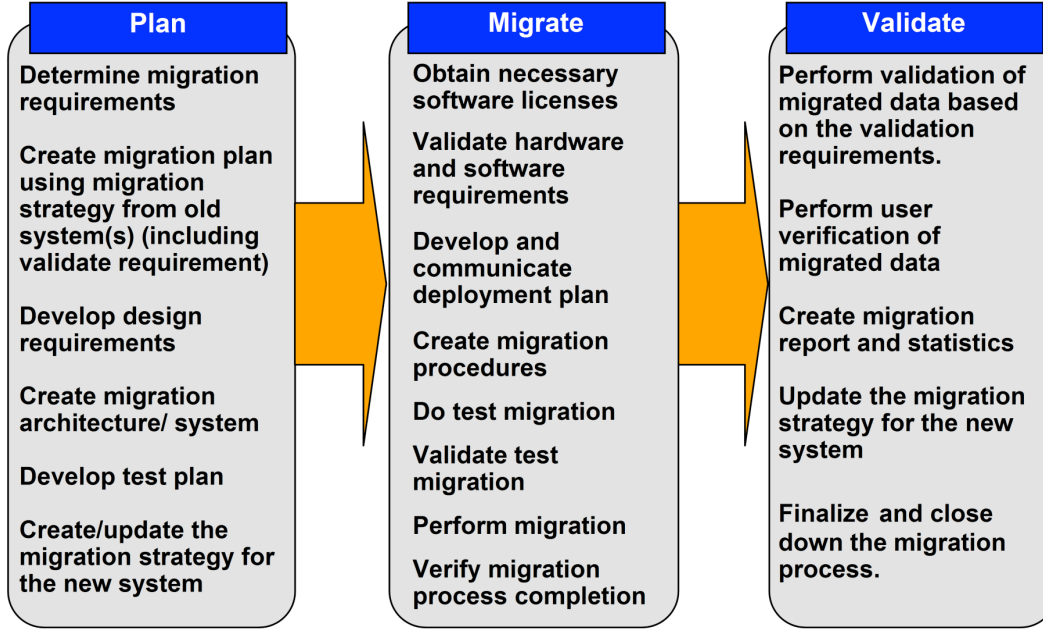


Figure 1. An overview of processes involved when performing a complete migration [16]

bandwidth of transferring from the host to its new storage. This transfer happens in two phases: 1) the storage drive reads bits from the storage media into its internal storage buffer. This speed is called internal transfer bandwidth,  $B_{IR}$ ; 2) the buffer sends off the bits to the host via an interface, e.g., IDE, SATA and USB. This speed is called the external transfer rate,  $B_{ER}$ . Correspondingly, for the write process there are  $B_{IW}$  and  $B_{EW}$ , respectively.

For typical transfers, the internal buffer restricts the speed to the slower of these two phases. In most situations, the external transfer bandwidth is much higher than the internal transfer bandwidth. For example, the external read bandwidth of the first generation of SATA is about 150MB/s while the internal read transfer bandwidth of current normal hard disks is about 50-60MB/s. Hence the read transfer bandwidth for those hard disks is equal to the internal read transfer bandwidth, i.e., 50-60MB/s.

**Average file processing speed ( $B_P$  in MB/s):**  $B_P$  refers to the processing speed when file modifications or other changes are required, e.g., format conversion and repackaging.  $B_P$  is affected by the host hardware configuration and the processing algorithm's complexity.

**Average network transfer bandwidth ( $B_N$  in MB/s):**  $B_N$  is the transfer bandwidth in a given network. Curators often use network to do data redundancy and storage management.

**The number of replicas ( $K$ ):** Migrations sometimes involves creating several replicas in different hosts or storages, usually tapes. The number of replica  $K$  will impact the performance of the whole migration progress.

## 4. MIGRATION MODELING

In our modeling approach, we will begin with the most basic form of migration (i.e., basic migration). We will then successively expand this basic migration model by integrating various procedures, processing (i.e., migration with processing), replication (i.e., extension with replication), and verification (i.e., extension with verification).

In the following, we will derive mathematical equations for the migration time depending on the involved processes. An upper bound for the migration time is obtained by assuming all involved processes are in series, whereas a theoretical lower bound is obtained by assuming total parallelism.

### 4.1 Serial process model: upper migration time limit

#### 4.1.1 Basic migration (BM)

A file is read from the storage A, loaded into the CPU in the host, and written to storage B. If the storage B is in another host, it can be sent off via a network. A and B do not have to be of the same type.

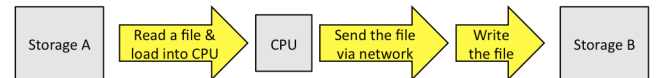


Figure 2. The BM model

If all processes are strictly serial, then the time required for transferring  $x$  MB is:

$$T_{MB}^U \approx \left( \frac{1}{B_R^A} + \frac{1}{B_N^B} + \frac{1}{B_W^B} \right) \cdot x \quad (1)$$

where  $B_R^A$  stands for the read bandwidth of the storage A,  $B_N^B$  and  $B_W^B$  are network transfer and write bandwidth of storage B.

#### 4.1.2 Migration with processing (MP)

After loading a file into the processor, some modification or conversion tasks are performed. Format conversion and archives' repackaging are two examples of MP. Any processing of a file may result in a change of the file size  $x \rightarrow x'$ .

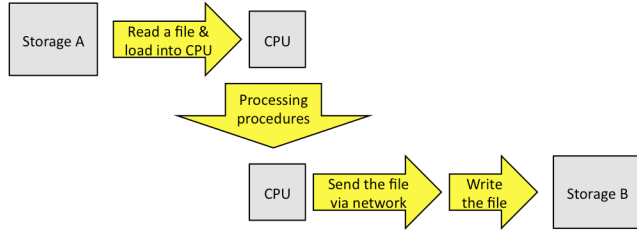


Figure 3. The MP model

Equation (2) gives an approximation of the migration time when a file processing occurs in the host A.

$$T_{MP}^U(x) \approx \left( \frac{1}{B_R^A} + \frac{1}{B_P^A} \right) \cdot x + \left( \frac{1}{B_N^B} + \frac{1}{B_W^B} \right) \cdot x' \quad (2)$$

where  $B_P^A$  is the processing bandwidth of the host that retrieved data from the storage A, and  $x'$  gives the new file size after processing.

#### 4.1.3 Extension with replication (ER)

A file is not only sent to storage B but also to  $K$  other storage media  $B^i$  ( $1 \leq i \leq K$ ). Assuming the replicas are successively written, the additional time required for each replica is:

$$T_{ER}^i(x) \approx \left( \frac{1}{B_N^i} + \frac{1}{B_W^i} \right) \cdot x \quad (3)$$

where  $i$  identifies the additional backup storage media.

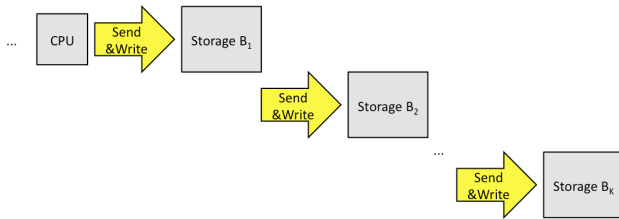


Figure 4. The ER model

If the file is copied to all of the backup storages one by one, the total time of replication is

$$T_{ER}^U(x) = \sum_{i=1}^K T_{ER}^i(x) \quad (4)$$

#### 4.1.4 Extension with verification (EV)

Verification determines whether a file has successfully been written to the new storage or backup storages, e.g., no bit error has occurred. If a file does not pass verification, the whole migration procedure for this file has to be repeated.

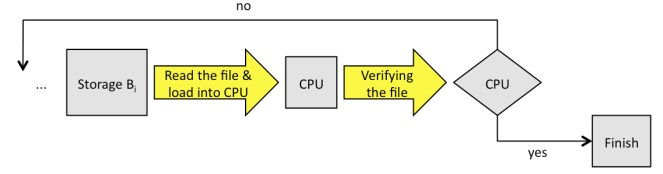


Figure 5. The EV model

For a storage  $B^i$ , the verification time  $T_{EV}^U$  is calculated by Equation (5).

$$T_{EV}^U(s) \approx \left( \frac{1}{B_R^i} + \frac{1}{B_V^i} \right) \cdot x \quad (5)$$

### 4.2 Parallel process model: lower migration time limit

So far, we have assumed a serial occurrence of all the individual migration processes. In reality, many of the involved processes will happen in parallel that in principal can lead to considerable time savings.

#### 4.2.1 Basic migration (BM)

A file is often divided into several blocks that can be processed in parallel. Through simplifying Equation (1), the new equation is:

$$T_{MB}^L(x) \approx \max\left(\frac{1}{B_R^A}, \frac{1}{B_N^B}, \frac{1}{B_W^B}\right) \cdot x \quad (6)$$

Equation (6) implies that the slowest of the read, networking or write processes could be considered as the bottleneck in parallel migration.

#### 4.2.2 Migration with processing (MP)

Depending on the modification required, a file might first have to be completely loaded into the CPU. The file should not be sent off until the corresponding processing has finished. In the MP model, it is assumed that only the transfer process and write process can overlap. Therefore, Equation (2) can be rewritten as:

$$T_{MP}^L(x) \approx \left( \frac{1}{B_R^A} + \frac{1}{B_P^A} \right) \cdot x + \max\left(\frac{1}{B_N^B}, \frac{1}{B_W^B}\right) \cdot x' \quad (7)$$

#### 4.2.3 Extension with replication (ER)

The replication procedure can be scheduled in parallel, like the transfer and write process in the BM model. Hence, the replication time for a given storage  $B^i$  becomes:

$$T_{ER}^i(x) \approx \max\left(\frac{1}{B_N^i}, \frac{1}{B_W^i}\right) \cdot x \quad (8)$$

Moreover, all backup storages can get the file from the host A at the same time, or the host A can send the file off to all backup storages at the same time. Hence, the replication time for all storages is revised as:

$$T_{ER}^L(x) = \max \left[ T_{ER}^i(x) \right] \quad (9)$$

#### 4.2.4 Extension with verification (EV)

Similar to the MP model, the EV extension requires that any file must be completely loaded into the CPU of the host before the verification can start. Hence, the read and the file verification processes cannot run in parallel.

To sum up, Equation (1) - (5) for the serial process model provide approximate equations for upper migration time limits, whereas Equation (6) – (9) provide lower time limits.

## 5. Migration Experiments

Two different sets of migration experiments were performed: one at the NB, the other between our personal computer systems at our department. The main objective is to test our models, so that we can identify the difference between theory and practice.

### 5.1 Migration at the NB

The Nation Library of Norway is considered the nation's memory. They store information about Norway's cultural heritage through a variety of media. The NB's long-term storage system is constituted by two tape arrays and one hard-disk array. Each tape array is a tape robotic system that stores up to 100 tapes, whereas the hard disk array adapts the RAID5 technique giving data redundancy and better I/O performance. All storage media are connected and managed under a Storage Area Network (SAN).

Our experiments were also carried out in this SAN environment including one server named Zarepta and two folders in two different hard disk arrays, DiskArray1 and DiskArray2. The task of Zarepta is to manage DiskArray1 and DiskArray2. Zarepta has four processors with Redhat Enterprise Linux. It directly connects to SAN via a fiber bus with 4GB/s transfer bandwidth. DiskArray1 and DiskArray2 are Sun StorageTek 6140 Arrays<sup>2</sup> with several 1TB hard disks with 7200rpm installed in it. They also connect to SAN via four GB/s fiber buses.

In order to compare various migration setups we first collected benchmark data for DiskArray1 and DiskArray2 by running Bonnie++<sup>3</sup> 10 times on each of them. The results are given in Table 2. All migration will be from DiskArray1 to DiskArray2, denoted as A and B respectively in our equations.

**Table 2. Benchmark of storage media**

Storage media	Read transfer bandwidth (MB/s)			Write transfer bandwidth (MB/s)		
	Best	Worst	Ave.	Best	Worst	Ave.
DiskArray1	300.8	220.73	264.32	131.78	89.88	112.05
DiskArray2	301.67	209.06	239.9	128.2	96.196	116.44

Three different migration setups were set up.

- Basic migration (BM)
- Basic migration and extension with verification (BM+EV)
- Migration with processing (MP)

Due to the technical limitation, we could not test all migration models, i.e., the situation with replication was discarded.

#### 5.1.1 Basic migration (BM)

In the BM case, a collection of video files was migrated. In order to reduce the complexity of the experiment, we created a folder containing 12 sub-folders each with 100 video files with the same size of 432 MB. In the experiments, different volumes of data were copied with the cp command in the Linux operating system. Upper and lower bounds for the migration time are given by Equation (1) and Equation (6) respectively.

Table 3 presents the experimental results for BM. We find that the measured migration time is a linear function in terms of the total data volume since the parameters will not be changed once the hardware configuration has been decided on.

**Table 3. BM's experiments. '+' or '-' means that the estimated time is higher or lower than the measured migration time respectively.**

Total data volume in MB	Average read/write transfer bandwidths		Worst read/write transfer bandwidths	
	Upper bound's rel. error in %	Lower bound's rel. error in %	Upper bound's rel. error in %	Lower bound's rel. error in %
86400	+23.62%	-15.84%	+48.65%	+1.87%
172800	+25.84%	-14.33%	+51.32%	+3.70%
259200	+22.88%	-16.35%	+47.77%	+1.26%
345600	+22.79%	-16.41%	+47.65%	+1.18%
432000	+20.38%	-18.05%	+44.75%	-0.81%
518400	+20.26%	-18.13%	+44.61%	-0.91%

If we use average read/write transfer bandwidths, as expected the lower bound underestimates the migration time, whereas the upper bound overestimates the required migration time. However, if we use the worst read/write transfer bandwidths, the estimates are both  $\pm$ . Thus, it cannot be used as a lower bound. However, we obtain more precise results for the lower bound since the relative error rate is -0.91% - 3.70%. This may be because other background programs decreased the storage's performance. It can therefore be concluded that storages should not be selected in terms of its best or average benchmark, but focus should be on the worst benchmark.

#### 5.1.2 Basic migration and extension with verification (BM+EV)

The basic migration experiment from the latest section (BM) was extended by adding a verification process after the copy process has finished. The verification program that was used is the Linux command named md5sum that uses an encryption algorithm MD5 to verify a specified file's integrity. Both writing and reading are carried on DiskArray2. In order to avoid conflict, we designed a schema to coordinate those two tasks. We assumed any file that has been written will be read immediately and cannot be interrupted by other writing tasks. Combining Equation (1) and

<sup>2</sup> [http://www.sun.com/storage/disk\\_systems/midrange/6140](http://www.sun.com/storage/disk_systems/midrange/6140)

<sup>3</sup> <http://www.coker.com.au/bonnie++/>

Equation (5) will provide the upper bound, whereas combining Equation (6) and Equation (5) will give the lower bound.

The experimental results are listed in Table 4. BM+EV generates similar results as BM. For example, the measured migration time is a linear function in terms of the total data volume, the lower bounds are close to the real migration time, and the worst read or write transfer bandwidths provided the most precise estimates. The reason why BM+EV and BM have similar phenomenon is that those migrations have the same bottleneck, i.e., DiskArray2.

**Table 4. BM+EV's experiments. '+' or '-' means that the estimate time is higher or lower than the measured migration time respectively.**

Total data volume in MB	Average read/write transfer bandwidths		Worst read/write transfer bandwidths	
	Upper bound's rel. error in %	Lower bound rel. error in %	Upper bound's rel. error in %	Lower bound's rel. error in %
86400	+23.62%	-15.84%	+128.10%	-15.11%
172800	+25.84%	-14.33%	+120.91%	-11.48%
259200	+22.88%	-16.35%	+105.76%	-3.84%
345600	+22.79%	-16.41%	+103.25%	-2.57%
432000	+20.38%	-18.05%	+106.53%	-4.23%
518400	+20.26%	-18.13%	+99.26%	-0.55%

Compared to BM, the migration time increases in BM+EV as the read and write transfer bandwidth of DiskArray2 is the bottleneck. If the new storage is fast enough, the bottleneck can be shifted to the old storage, e.g., DiskArray1. In terms of the equation for the lower bound, the total migration time will then be determined by the read transfer bandwidth of the old storage. Thus, instead of no increase for migration time, the reliability for digital objects can be improved since this verification task ensures bits integrity.

### 5.1.3 Migration with processing (MP)

In the MP experiment, all video files are converted from mp4 file format to avi file format with the open source program ffmpeg. We generated 200 identical mp4 video files with a file size of  $x=432\text{MB}$  which after conversion had a file size of  $x'=195\text{MB}$ . Equation (2) and Equation (7) can again be used to derive equations for the upper and lower bounds for the migration time respectively.

Table 5 summarizes the experimental results. Due to time limitations, we were restricted to transform a fixed amount of data only. The results clearly indicate that the file conversion process constitutes the bottleneck. In order to increase the performance, a faster program or computer should be chosen.

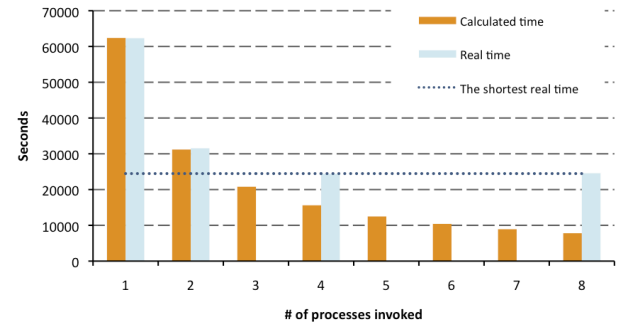
By running different processes on different CPUs, it should be possible to significantly decrease the migration time. An approximate equation for the processing time when running  $c$  parallel processes with  $C$  number of CPUs in a computer system is:

$$T(c) = \frac{T(1)}{c}, \quad 1 \leq c \leq C \quad (10)$$

**Table 5. MP's experiments. '+' or '-' means that the estimated time is higher or lower than the measured migration time respectively.**

Volume before conversion in MB	Volume after conversion in MB	# of processes invoked	Upper bound's rel. error in %	Lower bound's rel. error in %
86400	39000	1	+0.14%	-0.94%
86400	39000	2	-1.03%	-2.10%
86400	39000	4	-36.03%	-36.71%
86400	39000	8	-68.24%	-68.58%

Figure 6 presents that Equation (10) is a good approximation for the migration time when  $c \leq C-2$ . This has three reasons: 1) Zarepta has enough processors and memory space, so that parallel conversion processes can be run simultaneously. 2) In MP's experiment, the bottleneck is the file processing speed rather than the read/write transfer bandwidths. Hence, we can run several file conversion processes simultaneously. 3) When mp4 files are converted to avi, ffmpeg does not need too much memory to load the whole file. In practice, it converts one frame after the other. Therefore, under those conditions it is recommended to use parallel conversion processes to increase migration speed. The shortest migration time (the dotted line in Fig. 7) lies in the range of the migration time corresponding two and three parallel conversion processes. In general, the achievable processing time may be close to  $T(C-2)$  or  $T(C-1)$  as background processes will steal considerable processing power.



**Figure 6. Migration time for parallel migration processes. Increasing the number of processes invoked leads to a decreasing migration time proportional to  $1/c$  as long as  $c \leq C-2$ .**

## 5.2 Migration between personal computers

These experiments were performed between two personal computers. The objective of this test was again to validate the mathematical equations, and to explore the performance of different storage media. Only BM migration was done, i.e., no verification, no replication and no file processing involved.

Three storage media were tested: the hard disk (Disk1) in the computer A, the hard disk (Disk2) in the computer B, and the external hard disk (Disk3) using UBS2.0. The performance of the storage media was evaluated with HD Tune 2.55 and ATTO Disk Benchmark v2.34. The benchmarks are listed in Table 6.

Test setup and test scenarios:



**Case 1:** Transfer data from Disk1 to Disk2. Total data volume: 32427.7MB with 76705 files.

**Case 2:** Transfer data from Disk2 to Disk3. Total data volume: 30252.5MB with 72762 files.

**Case 3:** Transfer data from Disk1 to Disk2 via a 100M Internet. Total data volume: 3940.2 MB with 2617 files.

**Table 6. Benchmark of storage media**

	Seagate ST9160827AS (Disk1)	Samsung SP0812C (Disk2)	Hitachi IC25N040ATM R04 (Disk3)
<b>Capacity</b>	160GB	80GB	40GB
<b>Read transfer bandwidth</b>	36.04MB/s	42.49MB/s	7.51MB/s
<b>Write transfer bandwidth</b>	32.14MB/s	42.74MB/s	8.36MB/s

For each case, the actual migration was done by FastCopy, which is a multithread copy software application. According to the documentation of FastCopy, its copy speed should be 30%-50% faster than the copy command in Windows XP.

An estimate for the upper bound of the migration time is given by Equation (1) and for the lower bound by Equation (6). Table 7 lists the estimates for the migration times and the actual measured times.

**Table 7. Comparison of migration time when using FastCopy and calculated estimates. ‘+’ or ‘-’ means that the estimated time is higher or lower than the measured migration time respectively.**

	FastCopy time in s	Upper bound’s rel. error in %	Lower bound’s rel. error in %
<b>Case 1</b>	4467.97	+13.7%	-3.4%
<b>Case 2</b>	3524.66	+22.9%	-2.6%
<b>Case 3</b>	932.91	+22.2%	-0.5%

For all of the three test scenarios, the estimation of migration time based on the parallel model (lower bound) was quite close to the real observed migration time. As no tasks overlap in the sequential model, its time estimates (upper bound) grossly overestimate the required migration time. In this respect, the parallel model for estimating the migration time seems to be a viable and justifiable way.

In addition, personal computers are less suitable for migration due to their relative slow performance. However, several personal computers could be combined to achieve a migration speed similar to advanced computer systems. For example, the write transfer bandwidth was about 116.44MB/s in the storage array of NB. When combining three personal computers a read/write bandwidth in their hard disks of 30-40 MB/s (Table 6), we would get a migration speed close to 120MB/s. Thus, cheaper systems can reach a migration speed similar to an expensive system.

### 5.3 Summary of experimental results

Based on our experiments’ results, we can conclude that:

First, our mathematical equations provide reasonable estimations for upper and lower bounds for the expected migration time in all of the experiments. The experiments show that the actual migration time is quite close to the lower bound when the worst benchmarks for storages are used.

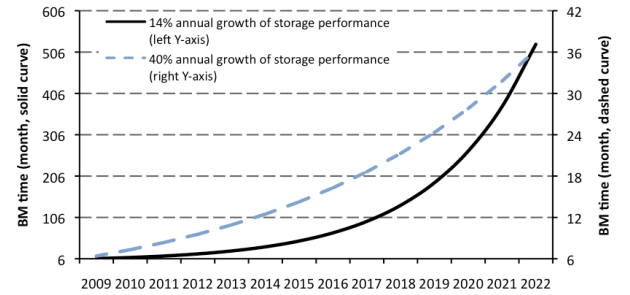
Second, it is important to analyze the bottleneck in migration, so that curators can design a better migration plan. For example, if the old storage constitutes the bottleneck in BM+EV, curators can do verification without adding time. In the MP experiment the processing program was the bottleneck. By adding enough empty processors and memory space, curators could increase the speed by running processing programs in parallel. Curators could obtain a similar migration performance as advanced computer systems but with less cost by combining several personal computers.

## 6. Discussion

So far preservation systems did relative easily cope with the growth in digital volume by applying the newest technology developments. The storage cost per GB is steadily falling. However, the expected exponential growth in data will crystallize two main challenges: file migration and file processing.

### 6.1 The challenge for BM

Basic migration may simply take too long time. At the NB, there are totally 1.8PB data in the end of 2009. Assuming a 60% annual growth in required storage volume [1], and an annual 40% increase in storage performance [17-18], the expected basic migration time is shown as the dashed curve in Figure 7. There may however be indications that a 40% annual improvement may be unrealistic in the long run [19]. A more realistic assumption of the annual growth rate may be closer to 14% [20] resulting in longer migration times (see the solid curve in Figure 7).

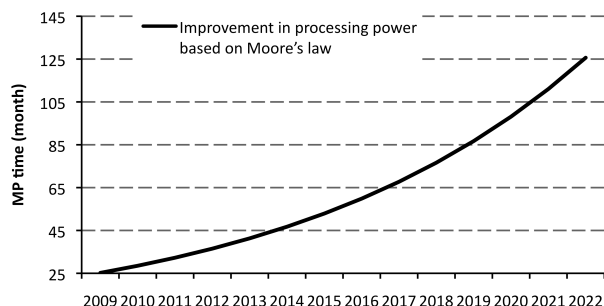


**Figure 7. Estimated migration time of BM at the NB. With a low improvement (14%) in storage performance migration will be increasingly longer (solid line, left Y-axis), whereas with the higher improvement rates (40%) migration will only grow moderately (dashed line, right Y-axis).**

Figure 7 indicates that the migration time may be over 50 months (4 years!) already from 2015 if the annual storage development is only at 14%. Obviously, the 4 year migration time is however unacceptable for preservation systems. With a 40% growth rate, this situation is postponed until 2022 when the migration time will last about 36 (3 year!) months.

### 6.2 The challenge for MP

In general, MP is very time consuming. For instance, the NB has 88 TB of video materials. If ffmpeg is used to do the format conversion, it would currently take about 25.2 months to process all video files. Assuming that Moore’s law [21] still holds in the near future, i.e. a doubling of processor performance in two years ( $\approx 41.4\%$  annual growth) then with a 60% annual growth in video volume [1], the expected migration time with file processing would be more than 125 months (over 10 years!) in 2022, see Figure 8.



**Figure 8. Estimated migration time of MP at the NB**

Moreover, video files constitute only a minor part at the NB. If all files (image, web, text, audio and video) have to be transformed, the total time might be out of the range of a decade.

### 6.3 Suggestions

As we see it, huge data bring challenges not only on data organization but also on data preservation. We suggest that curators use five lines of attacks:

Firstly, the simplest strategy for curators would be to dampen the growth in digital volume. Although many preservation institutions are asked to preserve everything that they receive, they should try to keep the volume growth at a relative slow rate compared to technology development rate. For example, they should remove redundancy data and only keep the original version and current version for digital objects.

Secondly, vendors should design more reliable storage solutions, so that they can provide longer support time for their solutions. The root cause for the migration is not the data volumes or processing speed but rather the relative short support agreements for storage systems, usually restricted to 3-4 years. If the support time can be longer, curators have more time to schedule their preservation tasks.

Thirdly, more advanced and faster storage solutions are needed. In the long run, the current disk technology may not be suitable for large-scale preservation systems due to their relative slow performance. Flash storage has in principle faster performance than disks, such as Solid-State Disks (SSDs), but still struggle with some quality issues. According to an investigation [22], the annual growth of SSD's performance is 12%.

Fourthly, curators may gain increased performance by utilizing distributed computing. An ideal distributed computing technology can disperse preservation tasks equally to different independent computers. Thus, migration time will be significantly reduced as more computers are involved.

Fifthly and finally, curators should deploy automated tools in migration. Currently, there are some projects to provide automated migration service, such as [10-11, 23]. However, curators usually lack programs to examine the result of a migration and to create metadata that record changes or lost parts.

## 7. CONCLUDING REMARKS

In this paper, a mathematical framework was derived for estimating the required migration time. The framework is based on a process modeling perspective. The parameters used are usually easily obtainable from data sheets of hardware vendors or benchmark software, so that several equations for the upper and

lower bounds of migration time can be derived for various migration tasks. We believe that our framework offers the following contributions:

First, the equations derived give reasonable estimates for the upper and lower bounds. Our validation experiments also indicate that the lower bounds are realistically close to the real migration times since relative error is between -18.13% (see Table 3) and -0.5% (see Table 7).

Second, our suggested equations allow curators to pinpoint potential migration bottlenecks. The bottlenecks can be different depending on the situation. Sometimes, it can be the read or write transfer bandwidth; sometimes the processing speed, and in other situations other parameters should be taken into account. However, in terms of current technologies, it seems that the read/write transfer bandwidth is the most likely bottleneck in BM, whereas the bottleneck could be the processing speed in MP, especially as with the case of format conversion of the video files.

Finally, we believe that this paper turns the attention of curators to a rather neglected aspect of preservation, which is the increasingly longer time required to migration. We hope that this framework will be used in preservation planning to derive or validate technical specifications.

## 8. ACKNOWLEDGMENTS

Research on this subject has been funded by the Norwegian Research Council and our industry partners under the LongRec project. We would thank the National Library of Norway, especially Lars Gaustad, for helping and providing experiment environments and technique supports.

## 9. REFERENCES

- [1] J. F. Gantz, et al., "The Diverse and Exploding Digital Universe: An updated forecast of worldwide information growth through 2011", 2008.
- [2] The Digital Preservation TestBed, "Migration: Context and Current Status", 2001.
- [3] K.-H. Lee, et al., "The State of the Art and Practice in Digital Preservation", Journal of Research of the National Institute of Standards and Technology, vol. 107, pp. 93-106, 2002.
- [4] K. Thibodeau, "Overview of Technological Approaches to Digital Preservation and Challenges in Coming Years", CLIR Reports, Conference Proceedings of The State of Digital Preservation: An International Perspective, 2002.
- [5] S. Granger, "Emulation as a Digital Preservation Strategy", D-Lib Magazine, vol. 6, 2000.
- [6] R. A. Lorie, "A methodology and system for preserving digital data", in Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries, Portland, Oregon, USA, 2002, pp. 312-319.
- [7] P. Mellor, et al., "Migration on Request, a Practical Technique for Preservation", in Proceedings of the 6th European Conference on Research and Advanced Technology for Digital Libraries, 2002, pp. 516-526.
- [8] N. Lormant, et al., "How to evaluate the ability of a file format to ensure long-term preservation for digital information?", in Ensuring Long-term Preservation and



Adding Value to Scientific and Technical data, Edinburgh, 2005.

- [9] S. Strodl, et al., "How to choose a digital preservation strategy: evaluating a preservation planning procedure", in Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries, Vancouver, BC, Canada, 2007, pp. 29-38.
- [10] J. Hunter and S. Choudhury, "PANIC: an integrated approach to the preservation of composite digital objects using Semantic Web services", International Journal on Digital Libraries, vol. 6, pp. 174-183, 2006.
- [11] M. Ferreira, et al., "A foundation for automatic digital preservation", Ariadne, vol. 48, 2006.
- [12] D. Waters and J. Garrett, "Preserving Digital Information. Report of the Task Force on Archiving of Digital Information", 1996.
- [13] P. Wheatley, "Migration—a CAMiLEON discussion paper", Ariadne, vol. 29, 2001.
- [14] Consultative committee for Space Data Systems, "Reference Model for an Open Archival Information System (OAIS) ", 2002.
- [15] A. Crespo and H. Garcia-Molina, "Cost-driven design for archival repositories", in Proceedings of the 1st ACM/IEEE-CS joint conference on Digital libraries, Roanoke, Virginia, United States, 2001, pp. 363-372.
- [16] T. Mest, K. Nymoen, J. I. Bøyum, L. Gaustad "Survivability of Digital Records", DNV Technical Record 2007-1623.
- [17] E. Grochowski and R. F. Hoyt, "Future trends in hard disk drives", IEEE Transactions on Magnetism, vol. 32, pp. 1850-1854, 1996.
- [18] E. Grochowski and R. D. Halem, "Technological impact of magnetic hard disk drives on storage systems", IBM Syst. J., vol. 42, pp. 338-346, 2003.
- [19] S. Gurumurthi, "Should disks be speed demons or brainiacs?", SIGOPS Oper. Syst. Rev., vol. 41, pp. 33-36, 2007.
- [20] S. Gurumurthi, et al., "Disk drive roadmap from the thermal perspective: a case for dynamic thermal management", in Computer Architecture, 2005. ISCA '05. Proceedings. 32nd International Symposium on, 2005, pp. 38-49.
- [21] G. E. Moore, "Progress in digital integrated electronics", in 1975 International Electron Devices Meeting, 1975, pp. 11-13.
- [22] Web-Foot Research, Inc., "Solid State Drives (SSD) Markets and Applications 2006-2011", 2006, pp. 68-70.
- [23] M. Ferreira, et al., "An intelligent decision support system for digital preservation", International journal on digital libraries, vol. 6, pp. 295-304, 2007.