[THSt07] *Lars Arne Turczyk, Oliver Heckmann, Ralf Steinmetz:* **File Valuation in Information Lifecycle Management;** Information Resources Management Association 18th IRMA International Conference 2007, Vancouver, Mai 2007.

File Valuation in Information Lifecycle Management

Lars Arne Turczyk
TU Darmstadt
KOM Multimedia Communications Lab
Merckstr. 25
64283 Darmstadt, Germany
Phone: +49 69 797 2778
lars.turczyk@siemens.com

Oliver Heckmann
TU Darmstadt
KOM Multimedia Communications Lab
Merckstr. 25
64283 Darmstadt, Germany
Phone: +49 6151 16 5188
heckmann@kom.tu-darmstadt.de

Ralf Steinmetz
TU Darmstadt
KOM Multimedia Communications Lab
Merckstr. 25
64283 Darmstadt, Germany
Phone: +49 6151 16 6150
ralf.steinmetz@kom.tu-darmstadt.de

ABSTRACT

Information Lifecycle Management (ILM) stores files according to their value. Therefore file valuation is a very important task in ILM environments. In this paper we look how the value of a file can be measured. Instead of traditional methods leading to a classical decimal-value the presented method leads to a valuation in terms of a "probability of further use". The new method's performance is checked using an ILM-simulator.

1. Introduction

ILM is based on the idea that in an enterprise there are different information with different values. Valuable information is stored on systems with high quality of service (QoS). The value changes over time and therefore migration of information to cheaper storage systems with lower QoS is required. Automated migration makes ILM dynamic. Such automation requires storage systems to understand what files are important at what time so that right policies can be applied. Nowadays ILM lacks at this point of information valuation methods and tools.

The question is "How is the value of a file measured?". Storage Network Industry Association (SNIA) states to measure the value as an amount of money [1].

Other methods express the value as a decimal-value [2]. In chapter 3 we show how this type of value can be derived out of a set of metadata. This method of valuation depends on different factors and has to be defined accurately. Getting metadata is not always easy or even possible. Therefore in section 4 we abandon metadata and show how the value can be derived using a probabilistic method. Here the value of a file is calculated out of usage information and expressed as a probability of further use.

This is a new method which allows valuation depending on the future importance of a file.

Section 5 proves the capabilities of the new method using an ILM-simulator.

Section 6 applies and combines both methods. The paper ends with a summary and an outlook on our future work.

The main contribution of this paper is the follows:

1. We present a new method of file valuation
2. We show that this probabilistic method works for ILM systems
3. We combine the new with an "old" method to optimize the performance.

2. Related Work

Usage information is used for valuation in other system domains as well. Google uses PageRank algorithm to rank the importance of a web page [3, 4]. A page is ranked mainly based on how many other pages link to it. Such links represent a form of usage. It indicates how many other pages are using that particular page. Caching algorithms often rely on data usage information to determine what data are important and hence what to cache in buffers in file systems, databases, and storage controllers [5, 6, 7]. These algorithms do not directly apply to our problem due to different design purposes and different target data.

Usage was the focus of Strange, too, who examined the long-term access behaviour on files in an UNIX system [8]. His aim was to identify regularities and patterns which can be applied to automated migration strategies for Hierarchical Storage Management (HSM).

Schmitz has also analyzed the access behaviour on files of a supercomputer to be able to derive an optimal migration strategy [9].

Miller and Gibson examined the access behaviour in further studies in UNIX environments and designed a "file aging algorithm" as a migration rule [10].

The self-* storage system at Carnegie Mellon University aims at automating storage management tasks through self-managing techniques [11]. It describes how one can classify files based on automatic learning of file properties using decision tree algorithms.

Chen focused on the file valuation for ILM. He erects value classes which are characterized by a unique set of attributes [2]. The valuation leads to a decimal value which can be normalized to the interval [0;1].

In contrast to other work the valuation presented in this paper offers a percent-value. It tells how big the probability of future accesses on a specific file is. Based on the percent-value an ILM-system can migrate files when their access probability falls below a predefined threshold.

3. File Valuation using metadata

Metadata are data that describe other data. Therefore in general for accurate valuation the more metadata is useable the better the valuation will be. For ILM relevant factors (but not limited) are [12]:

- Legislation
- Cost
- User
- File size
- File type

This list could be extended by factors representing the value of knowledge and intellectual properties [13, 14]. These business focused valuation methods require intense human interaction and organizational support. Hence they are often hard to implement.

The above mentioned five factors have different characteristics. For example, some are steady others are discrete, some are string variables others are real variables.

To obtain the information a form has to be filled. This needs human interaction and makes it difficult and expensive to receive the metadata. Nonetheless we discuss the parameters represented as a mapping which can be implemented into an online-form. We now look at each single factor.

Legislation: Each file in an enterprise has its own file retention period. In Germany the period can vary between 0 and 10 years. In American healthcare environment, for example, the period can be up to 100 years [15].

Let L(F) be the file retention period of file F given by legislation. L(F) is a discrete function:

$$L(F): F \mapsto L(F) \in \{0,1,2,5,10\} \subset N_0$$

Cost: Each file has a business importance for the enterprise. This importance is related to the cost originated from the missing of this specific file. Business importance is expressed in a currency (e.g. Dollar or Euro). Either the real value or a relative value is used. Real values are difficult to obtain. Here the relative value is used. The value might vary between 0 and 10.000.

Let C(F) be the cost originated from the missing of file F. C(F) is a steady function:

$$C(F): F \mapsto C(F) \in [0;10.000] \subset R_0^+$$

User: Each file is intended to be used by specific users within the enterprise. If this factor shall be used for valuation, the users' importance is distinguished between "low", "medium" and "high".

Let U(F) be the intended user group of file F. U(F) is a discrete function:

$$U(F): F \mapsto U(F) \in \{low,\ medium,\ high\}$$

File size: The size of a specific file can be used as a factor for valuation, too. The intention is to reduce the needed space on the expensive storage hierarchies. Therefore there is a special focus on the big files which represent a big capacity saving potential. The file size is finite and varies between the values "small", "medium", "big" and "very big". Depending on the enterprise the thresholds are set, e.g. between "big" and "very big" it can lie at 1MegaByte or much higher [9].

Let S(F) be the size of file F. S(F) here is a discrete function:

$$S(F): F \mapsto S(F) \in \{small, medium, big, very\ big\}$$

File type: The file type is determined by the application. In the office environment the most common file types are for example "doc", "xls" and "ppt".

A case study conducted in 2004 at an enterprise database identified 21 different file types with the following composition [16]:

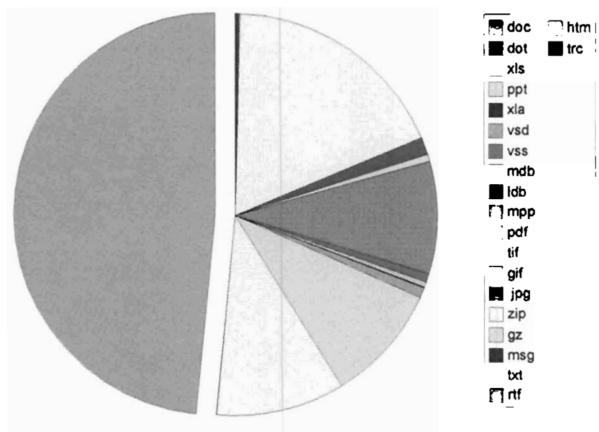

Figure 1 Distribution of file types

Let T(F) be the file type of file F. T(F) is a discrete function:

$$T(F): F \mapsto T(F) \in \{doc, xls, ppt, pdf, rest\}$$

Depending on the enterprise other file types like e.g. "jpg" or "mpg" can be assigned, too.

These mentioned five factors are summarized to a vector to derive the value of a file. Let V(F) be the value of file F. V(F) is a n-dimensional function (here n=5). It takes into consideration the factors "legislation", "cost", "user", "size" and "type" and determines the value as a decimal figure. The value is used to assign the files to the different hierarchies in an ILM-environment.

$$V(F): F \mapsto V(F) =$$
$$V(L(F), C(F), U(F), S(F), T(F)) \in R_0^+$$

Currently V(F) is only a theoretical mapping. The concrete procedure to derive the value out of the vector consists of transformations and normalizations. First the string variables (U(F), S(F) and T(F)) are transformed to real variables. Then the real variables are normalized to [0;1].

At the end the n-dimensional vector is reduced to a one-dimensional figure. This might happen by simple actions like:

V* := max {L*,C*,U*,S*,T*}[1] ∈ [0,1] or

V* := mean {L*,C*,U*,S*,T*} ∈ [0,1]

This shows that the valuation using metadata works. The advantage is of this type of valuation is that no history information is needed. On the other hand this procedure is neither easy nor cheap.

4. File valuation without metadata

Instead of metadata we now uses usage information for the valuation process. This proceeding results from one of the most intuitive metrics for file valuation "if a file is not used for a long time, it is not worthy".

In a case study on a database we provided following results [16]: There were more than 150.000 files on the system and 89 percent of them were not accessed 90 days after creation (see figure 2).

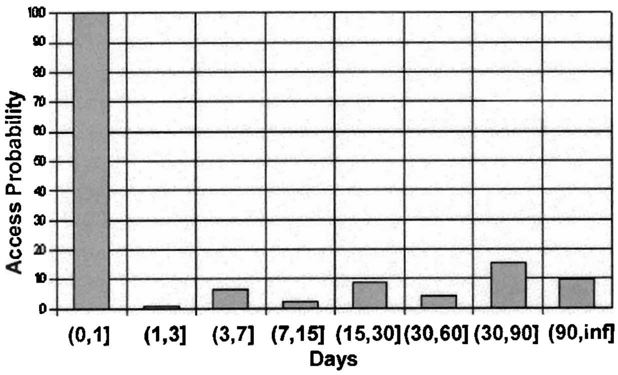

Figure 2: Access probability

[1] * indicates that transformation and normalization have been executed

The intuitive method for valuation would lead to the following policy:

"A file is worthy if it is accessed during the last 90 days and not worthy else."

This is a simple way to measure and by the way often used in HSM (Hierarchical Storage Management) –solutions. This shows that simple history-based valuation without metadata has a wide spread acceptance.

Nonetheless this method does not fit to ILM because it only looks on time limits and not if the file is needed in the business process which is the intention of ILM [1].

4.1 Case Study

We conducted another case study with the intention to get a predictable measure if the file is needed in the future.

Our aim was to derive mathematical distribution functions for file accesses for several file types.

We took a sample of 1000 files of a company's database.

The following tables characterise the random sample by illustrating the number of accesses per file (table 1), the size of the files (table 2), the size of the accesses (table 3), the age of the files (table 4), as well as the file types (table 5) and access methods (table 6).

Table 1: Number of accesses per file

Number of accesses	Number of files
[1;2)	307
[2;3)	152
[3;4)	99
[4;5)	79
[5;10)	209
[10;20)	77
[20;50)	53
[50;100)	14
[100;200)	6
[200;292)	4

Table 2: Size of the files

Size of files	Number of files
[1kB;10kB)	22
[10kB;50kB)	265
[50kB;100kB)	158
[100kB;500kB)	267
[500kB;1MB)	108
[1MB;2MB)	81
[2MB;5MB)	48
[5MB;10MB)	36
[10MB;50MB)	12

Size of files	Number of files
[50MB;115MB)	3

Table 3: Size of the accesses (size of the accessed file in each case)

Size of accesses	Number of accesses
[1kB;10kB)	169
[10kB;50kB)	2357
[50kB;100kB)	1228
[100kB;500kB)	1408
[500kB;1MB)	1458
[1MB;2MB)	440
[2MB;5MB)	426
[5MB;10MB)	322
[10MB;50MB)	65
[50MB;115MB)	38

Table 4: Age of the files (w = week, m = month, y = year)

Age of files	Number of files
[0;1 w)	7
[1w;1m)	37
[1m; 1/4y)	87
[1/4y; 1/2y)	109
[1/2y;1 y)	231
[1y;11/2 y)	247
[11/2 y;2 y)	80
[2 y;3 y)	138
[3 y;4 y)	36
[4 y;5 y)	28

Table 5: File types

File type	Number of files
doc	335
xls	185
ppt	164
pdf	140
zip	41
msg	24
miscellaneous	111

Table 6: Access types

Access type	Number of accesses
Version fetched	3657
View	1519
Version added	1392
Move	438
Reserve	256
Unreserve	247
Permission changed	200
Miscellaneous	202

The 1000 files in the sample were accessed a total of 7911 times between their respective creation and their extraction for the random sample (see table 1). Care must be taken when considering the number of accesses that the first access to a file in the examined database is logged at the time of its creation. As a result, 307 of the 1000 files were not accessed one single time after their creation. After discounting these "unused" files, most of the files, i.e. 152, were accessed only once after the creation date. The file types doc, xls, ppt, pdf and zip are contained most frequently in the sample (see Table 5). The file types avi, cfg, csv, cti, dot, exe, gif, htm, jpg, log, mdb, mmap, mmp, mp3, mpg, mpp, pps, pst, rtf, sql, tif, trc, txt, vsd, vss, wav, wbk, wf2 and xml fall into the category "miscellaneous". Most accesses to files in the sample, i.e. 46.22 % of 7911, are of the "version fetched" type (see Table 6). The access types "View" and "Version added" are represented with 19.20 % and 17.60 % at second and third place. Other frequently occurring access types are "Move", "Reserve", "Unreserve" and "Permission changed". The noticeably more seldom access types under "Miscellaneous" are "Attributes Changed", "Rename", "Copy", "Version Deleted", "Alias Created" and "Generation Created".

We derived distribution functions for file accesses in conjunction with the file type and access history [17].

As distribution function either the Weibull-distribution (W(α;β)) or Gamma-distribution (G(α;β)) were derived (see table 7).

Table 7: Applied distribution functions

Number of accesses / File type	1-6	7-14	15-∞
doc	W(0,35;3,5)	G(0,32;183)	W(0,35;3,5)
xls	W(0,25;1,1)	W(0,25;1,1)	W(0,25;1,1)
ppt	W(0,38;14,3)	W(0,38;14,3)	W(0,38;14,3)
pdf	W(0,35;3,5)	G(0,32;183)	W(0,35;3,5)
other	W(0,46;27,7)	G(0,29;181)	W(0,46;27,7)

With this approach we were able to calculate the future access probability of a file.

The valuation is executed on a percentage basis: "A file is worthy if its access probability is higher than, e.g., 5% and not worthy else".

This is a new quality of valuation using only the file type and the access history. This information is provided by the storage systems and does not need further metadata or user interaction.

5. Proof of concept

We implemented a simulator to analyse that the valuation based on "percentage of further accesses" can be used for ILM. Figure 3 shows an example using 5 hierarchies observed over 4000 days. The lifecycle is illustrated graphically for one single file.

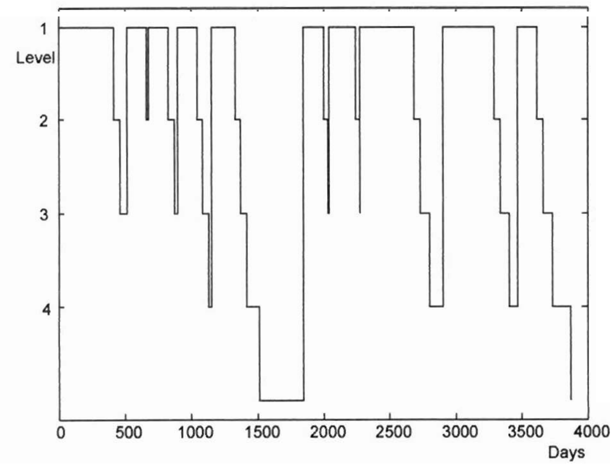

Figure 3: Migration of a file according to its value measured in "% of further accesses"

In figure 3 you can see that between day 1300 and day 1500 after the generation of the file the access probability sinks strongly. Therefore the file is moved on the lowest storage hierarchy in several steps. After approx. 1800 days the file is accessed, so that it must be shifted on the highest storage level again.

After about 3800 days the information is again at the lowest level. The lifecycle follows the typical course of a file with a periodical access sample.

For the simultaneous analysis of several files the graphic evaluation does not make sense. Therefore the examination of the relative capacity-need per hierarchy has to be carried out.

Figure 4 shows how an ILM-system with 3 hierarchies and the described valuation works:

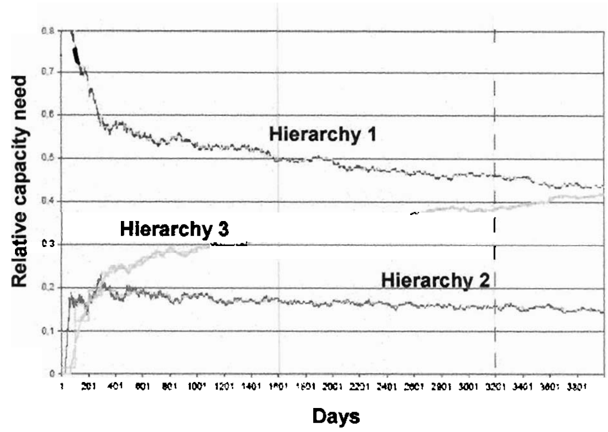

Figure 4: Relative capacity-needs in a 3-dim. ILM system

Since the method looks at the access history it needs some time to stabilize the capacity-need in an ILM system.

6. Combination of the derived valuation methods

When metadata is available we advise to use it in combination with the probabilistic method. Since the effort dealing with metadata is quite high, we advise to use metadata only for the initiation of an ILM scenario.

There are two options to initiate an ILM scenario (see figure 5):

Option 1: At the beginning storing all files on the highest hierarchy.

Option 2: At the beginning valuating all files and presorting them onto the related hierarchy.

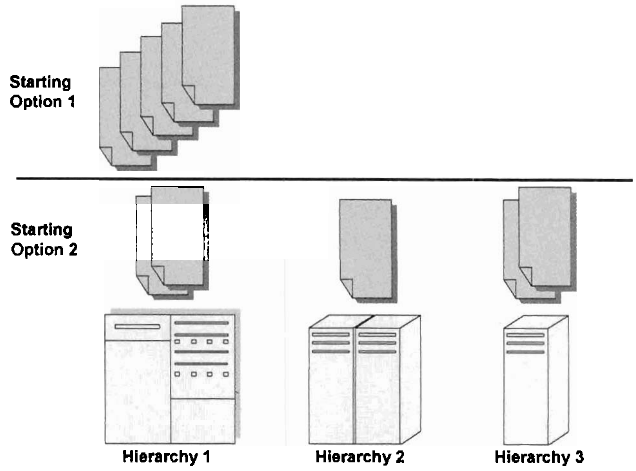

Figure 5: Starting options in an ILM-system

Option 1 does not require metadata, of course. The files are stored on the highest hierarchy. Their valuation is done on bases of file access patterns during the ILM process as shown in figure 4.

The advantage of option 2 compared to option 1 is that the capacity need for hierarchy 1 is below 100% from the beginning. This means that money is saved earlier.

Figure 6 shows a simulation run where the files where presorted according to option 2.

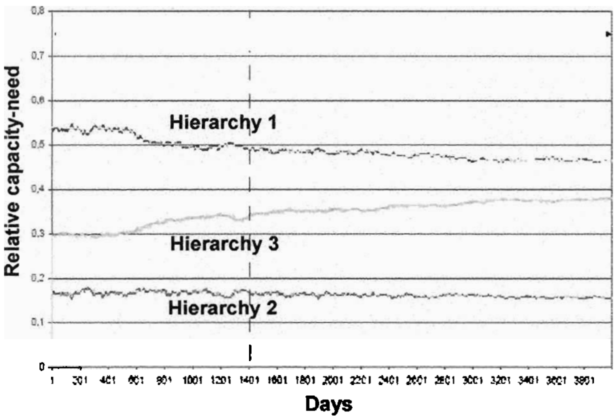

Figure 6: Relative capacity-needs in a presorted 3-dim. ILM system

We see that presorting has positive effects on reaching system stability.

7. Summary and Outlook

Proper information valuation is the first step towards ILM automation. Existing valuation methods either use metadata or look at the history and generate a value in terms of "amount of dollar" or "a decimal-figure within an interval".

We showed that from observed access data the future access of a file can be predicted and that this can be used as a metric for file valuation.

The value of a file is its percentage of further accesses. This is a new way of valuation. The advantages are that it is simple, does not need metadata and fits to ILM-automation. Furthermore it is more sophisticated than the often used simple HSM- time-limit approach.

The application of the new method was shown by means of an ILM-simulator. Therefore the first step towards automation is done.

Since the method looks at the access history it needs some time to stabilize the capacity-needs in an ILM system.

This period can be shortened by presorting the files over the hierarchies. We showed that in combination with an initial valuation according traditional methods using metadata earlier cost gains in ILM-systems can be achieved.

In our future research we further look for ILM automation. We focus on the aspect of policy definition. For comparison of different policies the existing simulator will be extended.

REFERENCES

[1] Peterson, M.: ILM Definition and Scope - An ILM Framework, SNIA Data Management Forum, Version 2.3, July 2004

[2] Chen, Y.: Information valuation for Information Lifecycle Management. In: Proceedings of the Second International Conference on Autonomic Computing (ICAC'05), pages 135-146, 2005.

[3] Page, L., Brin, S., Motwani, R. and Winograd. T.: The pagerank citation ranking: Bringing order to the web. http://dbpubs.stanford.edu:8090/pub/1999-66, 1999.

[4] Ridings, R. and Shishigin, M.: PageRank Uncovered. http://www.voelspriet2.nl/PageRank.pdf, 2002.

[5] Denning, P. J.: The working set model for program behavior. Communications of the ACM, 11(5), 1968.

[6] Denning, P. J. Working sets past and present. IEEE Transactions on Software Engineering, SE-6(1):64–84, 1980.

[7] Effelsberg, W. and Haerder, T.: Principles of database buffer management. ACM Transactions on Database Systems, 9(4), 1984.

[8] Strange, S.: Analysis of Long-Term UNIX File Access Patterns for Application to Automatic File Migration Strategies. Technical Report UCB/CSD-92-700, EECS Department, University of California, Berkeley, 1992.

[9] Schmitz, C.: Entwicklung einer optimalen Migrationsstrategie für ein hierarchisches Datenmanagement System. Technischer Bericht, Forschungszentrum Jülich GmbH, 2004.

[10] Gibson, T. and Miller E.: An Improved Long-Term File-Usage Prediction Algorithm, 1999.

[11] Mesnier, M., Thereska, E., Ganger, G. R., Ellard, D. and Seltzer, M.: File classification in self-* storage systems. In Proceedings of the First International Conference on Autonomic Computing (ICAC-04), New York, NY, May 2004.

[12] Turczyk, L. A.: Information Lifecycle Management: Organisation ist wichtiger als Technologie, Information Wissenschaft & Praxis 2005, Heft 7, S. 371-372

[13] Sveiby, K. E.: The balanced scorecard (bsc) and the intangible assets monitor – a comparison. http://www.sveiby.com/articles/BSCandIAM.html, 2001.

[14] Strassmann, P. A.: The value of computers, information and knowledge, January 1996. http://www.strassmann.com/pubs/cik/cik-value.shtml,

[15] Public Law 104-191, Health insurance portability and accountability act, August 1996. http://www.hipaadvisory.com/REGS/law/index.htm,

[16] Gostner, R., Turczyk, L., Heckmann, O. and Steinmetz, R.: Analyse von Datei-Zugriffen zur Potentialermittlung fuer Information Lifecycle Management, TU Darmstadt KOM Technical Report 01/2005

[17] Turczyk, L. Groepl, M., Heckmann, O. and Steinmetz, R.: Statistische Datenanalyse von langfristigem Dateizugriffsverhalten in einer Unternehmensdatenbank fuer ILM, TU Darmstadt KOM Technical Report 01/2006