

# Eine Methode zur Wertzuweisung von Dateien in ILM

Lars Arne Turczyk, Christian Frei, Nicolas Liebau, Ralf Steinmetz

KOM – Multimedia Communications Lab  
Technische Universität Darmstadt  
Merckstr. 25  
D-64283 Darmstadt  
lars.turczyk@siemens.com

**Abstract:** Information Lifecycle Management (ILM) speichert Dateien gemäß ihres Wertes. Somit ist die Wertzuweisung von Dateien eine der Hauptaufgaben in ILM. In diesem Papier betrachten wir wie der Wert einer Datei bestimmt werden kann. Während die bekannten Methoden einen Wert als Dezimalzahl ermitteln, präsentieren wir eine Methode, die den Wert einer Datei als die „Wahrscheinlichkeit zukünftiger Zugriffe“ ermittelt. Die Anwendbarkeit dieser Methode stellen wir mittels Simulation dar. Dabei vergleichen wir die neue Methode mit einer als Benchmark dienenden optimalen Methode, die allerdings nur unter Laborbedingungen funktioniert.

## 1 Einleitung

Das IT- und Speicherumfeld hat sich in den letzten Jahren erheblich verändert. Ein extremes Datenwachstum bei gleichzeitig länger werdenden Aufbewahrungszeiten und ein immer größerer Kostendruck lassen bisherige Ansätze der Speicherverwaltung an ihre Grenzen stoßen [LV03]. Längst geht es nicht mehr darum, einfach immer mehr Speicher bereitzustellen. Gefragt sind vielmehr umfassende, dynamische Konzepte, die sich an Lebenszyklus und Nutzung der Informationen orientieren. Information Lifecycle Management (ILM) verspricht Lösungen für diese drängenden Probleme. ILM ist ein Storage Management Konzept, das Informationen automatisiert entsprechend ihres Wertes auf dem jeweils kostengünstigsten Speichermedium bereitstellt und langfristig sicher aufbewahrt [Tu05]. ILM ist dynamisch und berücksichtigt, dass sich der Wert von Informationen mit der Zeit ändert.

Dazu werden Migrationsmechanismen benötigt, die bei Wertveränderungen die betreffenden Dateien automatisch zwischen den Speicherhierarchien verschieben. Eine derartige automatisierte Migration macht ILM zu einem dynamischen Konstrukt. Die Automatisierung verlangt zu wissen, welche Dateien zu welchem Zeitpunkt wertvoll und somit wichtig sind, damit die passende Migrationsregel angewendet werden kann. In diesem Punkt fehlt es ILM heutzutage an praktikablen Methoden der Wertzuweisung.

Die Hauptfrage bei ILM lautet “Wie lässt sich der Wert einer Datei automatisiert und dynamisch ermitteln?”.

Die Storage Network Industry Association (SNIA) schlägt vor, den Wert als einen Geldbetrag zu ermitteln [Pe04]. Andere Methoden drücken den Wert als Dezimalzahl [Ch05] oder als Zeitspanne seit letztem Gebrauch aus [Ta05].

Wir zeigen hier, wie der Wert mittels probabilistischer Methoden hergeleitet werden kann. Dabei wird der Wert einer Datei aus deren Zugriffsinformationen, Applikation (Dateityp) und Dateialter berechnet. Der Wert entspricht dann der „Wahrscheinlichkeit zukünftiger Zugriffe“. Diese neue Methode ermöglicht die Wertzuweisung in Abhängigkeit von der zukünftigen Bedeutung der Datei für das Unternehmen.

Zur Herleitung der Methode führten wir eine Fallstudie bei einem Dax-30-Unternehmen durch. Dabei wurde das Zugriffsverhalten auf Microsoft® Office-Dateien beobachtet [TG05].

Das vorliegende Paper stellt die Fallstudie und die abgeleiteten wahrscheinlichkeitstheoretischen Ergebnisse vor. Mit diesen Ergebnissen können Migrationsregeln für ILM formuliert werden. Dies geschieht am Ende dieses Papers, wenn konkrete Zugriffswahrscheinlichkeiten berechnet werden.

Der Beitrag dieses Papers ist folgender:

1. Wir präsentieren die Herleitung einer neuen Methode zur Wertzuweisung von Dateien.
2. Wir definieren eine optimale Methode als Vergleich.
3. Wir vergleichen unsere Methode mit der optimalen Methode.

## **2 Verwandte Arbeiten**

Die Migration von Dateien von teurem Speicher auf preiswerten Speicher ist Objekt verschiedener Studien seit den frühen Achtziger Jahren. Damals entwickelten Smith [Sm82] sowie Lawrie, Randal und Barton [LRB82] Datei-Auswahl-Algorithmen, die die Speicherbelegung optimieren sollten. Dabei betrachteten sie als Entscheidungskriterien das Dateialter und die Dateigröße unter der Nebenbedingung, die Anzahl der Rückmigrationen zu minimieren.

Das Langzeit-Zugriffsverhalten, welches auch unseren Untersuchungen zugrunde liegt wurde schon 1992 von Strange [St92] sowie 1998 und 1999 von Gibson et al. [GML98, GM99] und 2004 von Schmitz [Sc04] untersucht. Der Beobachtungszeitraum umfasste zwischen 84 Tagen bei Strange und 280 Tagen bei Gibson et al. Die untersuchten Dateien stammten dabei von UNIX-Systemen in deutschen bzw. amerikanischen Forschungszentren.

Im Gegensatz dazu stammen die in diesem Paper verwendeten Daten von einem Microsoft-Dateisystem eines DAX-30-Unternehmens. Weiterhin ist der Beobachtungszeitraum weitaus größer. Es wurden Lebenszyklen bis zu 1771 Tagen betrachtet.

Strange entwickelte den "least-recently used algorithm". Dieser verschiebt diejenigen Dateien zuerst, die am längsten nicht genutzt wurden. Gibson and Miller untersuchten den sogenannten "file-aging algorithm", der einen Migrationswert ermittelt. Daneben werden die Dateigröße und die Zeit seit letztem Zugriff in die Kalkulation einbezogen [GML98, GM99]. Der Migrationswert soll die Nutzungsintensität in Abhängigkeit von der Zeit repräsentieren. Der Wert steigt, wenn die zugehörige Datei genutzt wird, und fällt mit jedem Tag, an dem kein Zugriff erfolgt.

Die beschriebenen Publikationen bestimmen keine statistischen Verteilungsmodelle, worin sie sich fundamental von diesem Papier unterscheiden.

Nutzungsinformationen zur Bewertung werden auch in anderen Bereichen genutzt. Google© nutzt den "PageRank Algorithmus", um die Bedeutung einer Internetseite einzuordnen [Pa99, RS02]. Eine Internetseite wird hauptsächlich danach beurteilt, wie viele anderen Internetseiten mit ihr verlinkt sind. Diese „Verlinkung“ stellt eine Form von Nutzung dar. Sie zeigt, wie viele andere Seiten diese Seite nutzen.

Auch Caching Algorithmen nutzen oftmals Informationen über die Nutzung von Daten. Ziel ist es, diejenigen wertvollen Daten zu ermitteln, die dann in den Buffern der Dateisysteme, Datenbanken oder Speicher Controller vorgehalten werden sollen [De68, De80, EH84]. Weil beim Caching die Speichergröße fix ist und die Speicherbelegung im Fokus steht, können derartige Algorithmen aber für ILM nicht angewendet werden, weil hier die optimale Speichergröße über die Zeit variiert.

Heutzutage liegt ILM im strikten Fokus der Forschung. Die Hauptergebnisse liegen in den Bereichen der Prozeduren und Migrationsregeln. Auch stellten Speicherhersteller ihre Sicht von ILM dar [Re04]. Turczyk et. al. [Tu06] lieferten eine formale Definition von ILM, welche erlaubt, ILM abstrakt zu untersuchen. Sie lautet (eigene Übersetzung):

*Information Lifecycle Management (ILM) ist die Abbildung der Informationen  $I_1, \dots, I_n$  auf Speicherklassen  $C_1, \dots, C_m$ , gemäß der Werte  $V(I_1), \dots, V(I_n)$  in dem Zeitintervall  $[t_1, t_2]$ .*

Beigi et. al. [Be05] sowie Tanaka et. al. [Ta05] machten Vorschläge wie in ILM statische Migrationsregeln zu erstellen seien. Chen [Ch05] befasste sich direkt mit der Bewertung von Dateien. Sein Ansatz unterscheidet von unserem dadurch, dass er keine Wahrscheinlichkeiten zur Bewertung verwendet.

### 3 Charakteristika von Bewertungsmethoden

Die Bewertung ist das Kernstück von ILM. In diesem Abschnitt betrachten wir, welche Eigenschaften eine Methode haben muss, um in ILM-Szenarien eingesetzt werden zu können.

Laut Definition der Storage Network Industry Association (SNIA) ist ILM wertgesteuert. [Pe04] Die Wertveränderungen sollen automatisch registriert werden und in eine dynamische Speicherbelegung münden. Diese Charakteristika sind von der anzuwendenden Bewertungsmethode sicherzustellen. Das heißt die Methode muss automatisch und täglich den Wert ermitteln. Damit sind Methoden, für die manuell Metadaten erhoben werden, nicht geeignet, weil sie zu aufwendig und zu träge sind.

Dienstleister für Speicherplatz berechnen den benutzten Speicher in der Regel auf Monatsbasis. Dazu wird der Durchschnitt aus verschiedenen Messtagen des Rechnungsmonats herangezogen. Aus diesem Grund kann eine Methode, die täglich den Wert jeder Datei ermittelt und Migrationen bewirkt, direkt auf die Kosten Einfluss nehmen.

Auf der einen Seite soll die Bewertung mehrere Faktoren berücksichtigen, um möglichst realitätsnah den Wert zu ermitteln. Andererseits muss für jede Datei täglich die Bewertung durchgeführt werden.

Daraus ergeben sich für die Auswahl einer Bewertungsmethode drei Hauptcharakteristika:

- „multifaktorielle Wertermittlung“ für möglichst große Realitätsnähe
- „automatische Wertermittlung“ zur Reduzierung des Bewertungsaufwandes
- „dynamische Wertermittlung“ zur Berücksichtigung von Wertänderungen auf Tagesbasis

#### 3.2 Die Methode der „Wahrscheinlichkeiten zukünftiger Zugriffe“

Zu Herleitung unserer Bewertungsmethode führten wir eine Fallstudie an einer Datenbank bei einem Dax30-Unternehmen durch [TG06] Die untersuchte Datenbank enthält ca. 150.000 Dateien und ihre Zugriffsprotokolle. Für die Durchführung der folgenden statistischen Analysen wurde eine Zufallsstichprobe von 1000 Dateien entnommen. Über jede Datei sind folgende Informationen bekannt: Dateityp, Dateigröße, Datum und Uhrzeit (minutengenau) der Dateierstellung sowie Datum, Uhrzeit und Art der einzelnen Zugriffe. Diese Informationen wurden nach der Stichprobenentnahme mit MS-Excel aufbereitet, so dass die für die jeweiligen Analysen benötigten Daten zur Verfügung standen. Für die Datenanalyse selbst wurden die Programme R und MATLAB verwendet.

Die Tabellen 1 bis 3 charakterisieren die Stichprobe, indem sie die Häufigkeitsverteilungen der Anzahl der Zugriffe pro Datei, des Alters der Dateien sowie die in der Stichprobe enthaltenen Dateitypen darstellen.

#Zugriffe	[1;2)	[2;3)	[3;4)	[4;5)	[5;10)
#Dateien	307	152	99	79	209
#Zugriffe	[10;20)	[20;50)	[50;100)	[100;200)	[200;292)
#Dateien	77	53	14	6	4

Tabelle 1: Anzahl der Zugriffe je Datei

Dateialter	[0;1 W)	[1 W;1 M)	[1 M;¼ J)	[¼ J;½ J)	[½ J;1 J)
# Dateien	7	37	87	109	231
Dateialter	[1 J;1½ J)	[1½ J;2 J)	[2 J;3 J)	[3 J;4 J)	[4 J;5 J)
# Dateien	247	80	138	36	28

Tabelle 2: Alter der Dateien (W = Woche, M= Monat, J= Jahr)

Dateityp	doc	xls	ppt	pdf	zip	msg	Sonstige
# Dateien	335	185	164	140	41	24	111

Tabelle 3: Dateitypen

Auf die 1000 Dateien der Stichprobe wurde seit ihrem jeweiligen Bestehen bis zur Stichprobenentnahme insgesamt 7911 mal zugegriffen (siehe Tabelle 1). Bei der Anzahl der Zugriffe ist jedoch zu beachten, dass in der untersuchten Datenbank stets der erste Zugriff auf eine Datei zum Zeitpunkt der Dateientstehung protokolliert ist. Auf 307 der 1000 Dateien wurde demzufolge nach dem Entstehungszeitpunkt kein einziges mal mehr zugegriffen. Diese „nicht verwendeten“ Dateien einmal ausgenommen, wurde auf die meisten Dateien, nämlich 152, nur einmal nach dem Entstehungsdatum zugegriffen. Die Dateitypen doc, xls, ppt, pdf und zip sind am häufigsten in der Stichprobe enthalten (siehe Tabelle 3). Unter „Sonstige“ fallen die Dateitypen 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 und xml.

Um mehr über die Zugriffe der Dateien zu erfahren wenden wir nun wahrscheinlichkeitstheoretische Methoden an. Ziel ist, die Verteilungsfunktionen von Dateizugriffen zu ermitteln. Dazu betrachten wir die Zufallsvariable  $X : \Omega \rightarrow \mathbb{R}$ , wobei  $X(\omega)$  die „Anzahl der Tage seit dem letztem Zugriff“ abbildet. mit  $\omega \in \Omega$ . Hier ist  $\Omega$  die „Menge aller Zugriffe“ und  $\omega$  ein beliebiges Element daraus.

Wenn man nun eine Verteilungsfunktion für  $X$  findet, so folgt daraus direkt die „Wahrscheinlichkeit zukünftiger Zugriffe einer Datei“

Da eine Zeitspanne zwischen zwei Ereignissen modelliert wird, kommen als Verteilungen nur sogenannte Lebensdauerverteilungen in Frage [Ha95 und Sc03].

Mittels Q-Q-Plot Tests lässt sich rasch die Liste potentieller Kandidaten auf zwei reduzieren. Diese sind die Weibull- und die Gamma-Verteilung ( $W(\alpha, \beta)$  und  $G(\alpha, \beta)$ ).

Die nachfolgende Tabelle gibt exemplarisch die Testergebnisse mit gemischter gestutzter Weibull-Verteilung wieder. Dabei lautet die jeweilige  $H_0$ :

$H_0$ : Die Zufallsvariable  $X$  entstammt einer Grundgesamtheit mit einer gemischten Verteilung der gestutzten Gammaverteilungsfunktion mit den Parametern  $\hat{\alpha}$  und  $\hat{\beta}$ .

Dateityp	$\hat{\alpha}$	$\hat{\beta}$	Ablehnungsbereich	Ergebnis
doc	0.38	23.6	$T > \chi_{33;0.001}^2 \Leftrightarrow 91.85 > 63.87$	$H_0$ abgelehnt
xls	0.25	1.1	$T > \chi_{29;0.001}^2 \Leftrightarrow 49.59 < 58.30$	$H_0$ nicht abgelehnt
ppt	0.38	14.3	$T > \chi_{31;0.001}^2 \Leftrightarrow 39.69 < 61.10$	$H_0$ nicht abgelehnt
pdf	0.48	21.9	$T > \chi_{28;0.001}^2 \Leftrightarrow 79.59 > 56.89$	$H_0$ abgelehnt
sonstige	0.46	27.7	$T > \chi_{30;0.001}^2 \Leftrightarrow 38.83 < 59.70$	$H_0$ nicht abgelehnt

Tabelle 4:  $\chi^2$  Tests pro Dateityp

Die Durchführung aller Tests führt zu folgenden Ergebnissen. Es ist uns gelungen, für einige Untergruppen geeignete Verteilungsmodelle zu konstruieren. Die Tabelle 5 gibt einen Überblick über die Testergebnisse.

Kriterium	Klasse	Verteilung
Dateialter	[0 Tage;365 Tage)	W(0.35,3.5)
	[365 Tage;730 Tage)	-
	[730 Tage;1772 Tage)	-
Anzahl der Zugriffe	[1 Zugriff;7 Zugriffe)	-
	[7 Zugriffe;15 Zugriffe)	G(0.32,183)
	[15 Zugriffe;292 Zugriffe)	W(0.36,4.0)
Dateityp	doc	-
	xls	W(0.25,1.1)
	ppt	W(0.38,14.3), G(0.19,221)
	pdf	-
	sonstige	W(0.46,27.7), G(0.29,181)

Tabelle 5: Zusammenfassung der Testergebnisse

Nun können wir direkt Dateien bewerten.

Beispiel 1: Dateityp: doc, Alter: 50 Tage, Zugriffe: 10, Letzter Zugriff vor 5 Tagen.

Diese Datei hat eine Wahrscheinlichkeit zukünftiger Zugriffe von 60.05 %.

Beispiel 2: Dateityp: pdf, Alter: 420 Tage, Zugriffe: 3, Letzter Zugriff vor 230 Tagen.

Diese Datei hat eine Wahrscheinlichkeit zukünftiger Zugriffe von 2.44 %.

Beispiel 3: Dateityp: sonstige, Alter: 30 Tage, Zugriffe: 20, Letzter Zugriff vor 2 Tagen.

Diese Datei hat eine Wahrscheinlichkeit zukünftiger Zugriffe von 67.82 %.

Die gezeigte Methode ist multifaktoriell, weil 4 Faktoren in die Bewertung einfließen. Sie funktioniert automatisch, weil keine Metadaten erhoben werden müssen. Ferner ist sie dynamisch, weil sie ermöglicht, die Wertveränderungen auf Tagesbasis zu ermitteln. Damit erfüllt diese Methode die oben genannten Hauptkriterien einer Bewertungsmethode für ILM

### 3.1 Definition einer optimalen Methode

Bei der nachfolgend definierten optimalen Methode handelt es sich um eine Methode zur optimalen Speicherbelegung. Diese optimale Methode basiert darauf, dass aus der Fallstudie für jede Datei ein komplettes Zugriffsprotokoll erstellt wurde. Nur mit diesem Zugriffsprotokoll je Datei kann man die optimale Methode benutzen, weil diese Methode Wissen über die Zukunft verlangt (Antizipation).

Definition (Optimale Bewertungsmethode der Antizipation))

Eine Datei, die erzeugt wird, wird auf der obersten Hierarchie gespeichert (Wert=1).

Eine Datei, die nicht genutzt wird, wird auf die unterste Hierarchie migriert (Wert=0).

Eine Datei, die am nächsten Tag genutzt wird, wird auf die oberste Hierarchie migriert (Wert=1).

Es geht also darum, die Dateien optimal zu migrieren. Dementsprechend ist die optimale Methode keine tatsächliche Bewertungsmethode, sondern eine Migrationsregel. Der implizite Wert ist entweder 0 oder 1.

Offensichtlich ist diese Methode der Antizipation nur von theoretischer Natur, weil real niemand konkret Dateizugriffe für den nächsten Tag vorhersagen kann. Der Nutzen dieser Methode besteht also nicht in der Anwendbarkeit, sondern darin, dass sie als Vergleich für tatsächlich anwendbare Methoden herangezogen werden kann. Dies möchten wir im nächsten Abschnitt tun. Nachfolgend vergleichen wir unsere Methode der „Wahrscheinlichkeit zukünftiger Zugriffe“ mit der optimalen Methode der „Antizipation“. Dazu nutzen wir einen selbstentwickelten ILM-Simulator.

## 4 Vergleich der beiden Methoden

Wir nehmen eine feste Stichprobe von 10000 Dateien, die mit beiden Methoden über einen Zeitraum von 1000 Tagen simuliert wird. Die Stichprobe hat eine Gesamtgröße von 5,5 GB. Es wird jeweils ein ILM-Szenario mit drei Hierarchien simuliert.

In der optimalen Methode werden per Definition nur die erste und dritte Hierarchie genutzt. Die folgende Grafik zeigt das Ergebnis der benötigten Kapazitäten auf den jeweiligen Hierarchien:

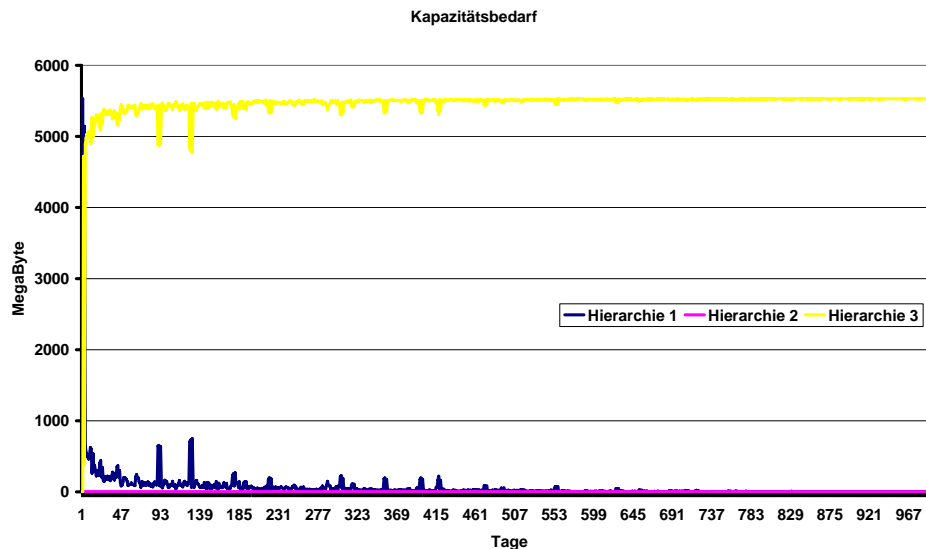


Abbildung 1: Kapazitätsbedarf je Hierarchie bei der optimalen Methode

Man sieht, dass die tatsächlich an einem Tag benutzten Dateien weniger als 1 GB ausmachen. Die ungenutzten Dateien sammeln sich in der dritten Hierarchie und machen im Laufe der Zeit fast den gesamten Kapazitätsbedarf aus.

Berechnet und gewichtet man nun die Flächen unter den Kurven, so ergibt sich die Kostenersparnis im Vergleich zu einem Monolithischen Speichersystem, bei dem alle 5,5 GB auf der obersten (und einzigen) Speicherebene gelagert werden. Im Falle der optimalen Lösung liegt die Ersparnis bei 95%

Betrachten wir nun unsere Methode. Mit derselben Stichprobe wird 1000 Tage simuliert. Eine Datei wird von Ebene 1 auf Ebene 2 migriert, wenn die Wahrscheinlichkeit eines weiteren Zugriffes unter 10% liegt. Der Schwellwert zur Migration von Hierarchie 2 auf Hierarchie 3 beträgt 5 %. Die Schwellwertfestlegung hier ist exemplarisch. In Realität können die Schwellwerte von Unternehmen zu Unternehmen variieren

Man erkennt, dass bei dieser Simulation jede Hierarchie genutzt wird. Zu Beginn lagern alle Dateien auf der ersten Hierarchie. Nach ca. 140 Tagen werden ungefähr 2 GB auf die zweite Ebene migriert. Nach fast einem Jahr befinden sich 4,5 GB auf der zweiten Ebenen, die damit ihren maximalen Kapazitätsbedarf beschreibt. Ab diesem Zeitpunkt füllt sich die dritte Hierarchie stetig bis sowohl die erste als auch die zweite fast leer sind. In diesem Verhalten gleichen sich beide Methoden

Nachfolgend ist das Simulationsergebnis dargestellt:

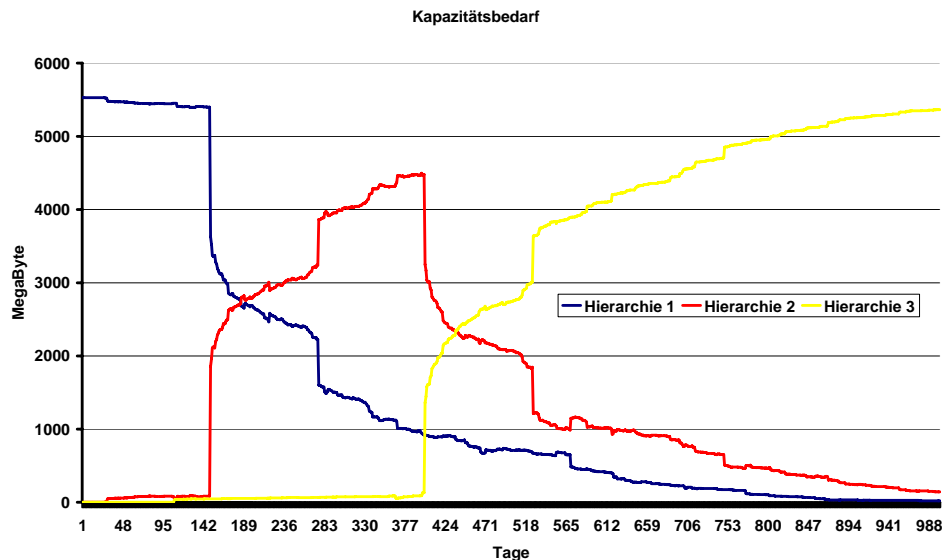


Abbildung 2: Kapazitätsbedarf je Hierarchie bei der Methode der Wahrscheinlichkeit zukünftiger Zugriffe

Betrachtet man nun wieder die Flächen unter den Kurven, so ergibt sich die Kostenersparnis im Vergleich zu einem Monolithischen Speichersystem, von 69%. Es zeigt sich, dass unsere Methode für ILM einsetzbar ist und die von ILM erwarteten positiven Effekte realisiert.

## 5 Zusammenfassung und Ausblick

Die Bewertung von Dateien ist eine Hauptaufgabe in ILM-Umgebungen. Für die Auswahl einer Bewertungsmethode lassen sich drei Hauptcharakteristika ableiten: „multifaktoriell“, „automatisch“ und „dynamisch“. Wir stellten eine Methode vor, die die Wahrscheinlichkeit zukünftiger Zugriffe einer Datei ermittelt. Das Verfahren ist multifaktoriell und funktioniert automatisch, womit es der ILM-immanenten Dynamik gerecht wird. Die Verwendbarkeit dieser Methode wurde mittels Simulation gezeigt.

Im Vergleich mit einer optimalen Lösung zeigt sich, dass unsere Methode bis auf 26 Prozentpunkte an die optimale Methode herankommt. Da dieser Vergleich nur von theoretischer Natur ist, werden wir in weiteren Arbeiten unsere Methode mit realistischen Methoden vergleichen.

## Literaturverzeichnis

- [Be05] Beigi, M., Devarakonda, M., Jain, R., Kaplan, M., Pease, D., Rubas, J., Sharma, U., Verma, A., (2005) "Policy-Based Information Lifecycle Management in a Large-Scale File System". In Proceedings of the Sixth IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY'05) pp. 139-148
- [Ch05] Chen, Y. (2005) Information valuation for Information Lifecycle Management. In: Proceedings of the Second International Conference on Autonomic Computing (ICAC'05), pages 135-146.
- [De68] Denning, P. J., (1968) "The working set model for program behavior". Communications of the ACM, 11(5), 1968.
- [De80] Denning, P. J., (1980) "Working sets past and present". IEEE Transactions on Software Engineering, SE-6(1):64-84, 1980.
- [EH84] Effelsberg, W. and Haerder, T., (1984) "Principles of Database Buffer Management". ACM Transactions on Database Systems, 9(4), 1984.
- [GM99] Gibson, T. and Miller, E. (1999), An Improved Long-term File-Usage Prediction Algorithm, University of Maryland, Baltimore County, USA.
- [GML98] Gibson, T., Miller, E. and Long, D. (1998), Long-term File Activity and Inter-Reference Patterns, 24th International Conference on Technology Management and Performance Evaluation of Enterprise-Wide Information Systems, Computer Measurement Group, Anaheim, California, USA.
- [Ha95] Hartung, J. (1995), Statistik - Lehr- und Handbuch der angewandten Statistik, 10. Aufl., München u.a., Oldenbourg.
- [LRB82] Lawrie, D. H., Randal, J. M. and Barton, R. R., (1982) "Experiments with Automatic File Migration," IEEE Computer 15(7), 1982, pp. 45-55.
- [LV03] Lyman, A and Varian, B.(2003) How Much Information?, University of California, Berkeley, California, USA, 2003
- [Pa99] Page, L., Brin, S., Motwani, R. and Winograd. T., (1999) "The pagerank citation ranking: Bringing order to the web". <http://dbpubs.stanford.edu:8090/pub/1999-66>, 1999.
- [Pe04] Peterson, M (2004), ILM Definition and Scope - An ILM Framework, SNIA Data Management Forum, Version 2.3, July 2004. [www.snia.org/tech\\_activities/dmf/docs](http://www.snia.org/tech_activities/dmf/docs)
- [Re04] Reiner, D. Press, G., Lenaghan, M., Barta, D., Urmston, R., (2004) "Information Lifecycle Management: The EMC Perspective". Proceedings of the 20th International Conference on Data Engineering (ICDE'04) 1063-6382/04
- [RS02] Ridings, R. and Shishigin, M., (2002) "PageRank Uncovered". <http://www.voelspriet2.nl/PageRank.pdf>, 2002.
- [Sa04] Sachs, L. (2004), Angewandte Statistik - Anwendung statistischer Methoden, 11. Aufl., Berlin u.a., Springer.
- [Sc03] Schlittgen, R. (2003), Einführung in die Statistik - Analyse und Modellierung von Daten, 10. Aufl., München u.a., Oldenbourg.
- [Sc04] Schmitz, C. (2004), Entwicklung einer optimalen Migrationsstrategie für ein hierarchisches Datenmanagement System, Forschungszentrum Jülich GmbH, Jülich.
- [Sm82] Smith, A. J., (1982) "Long Term File Migration: Development and Evaluation of Algorithms," Communications of ACM 24(8), 1982, pp.521-532.
- [St92] Strange, S. (1992), Analysis of Long-term Unix File Access Patterns for Application to Automatic File Migration Strategies, University of California, Berkeley, California, USA.
- [Ta05] Tanaka, T., Ueda, R., Aizono, T., Ushijima, K., Naitoh, I., Komoda, N. (2005) Proposal and Evaluation of Policy Description for Information Lifecycle Management Proceedings of the 2005 International Conference on Computational Intelligence for Modelling, Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'05) 0-7695-2504-0/05

- [TG05] Turczyk, L., Gröpl, M. Heckmann, O. und Steinmetz, R., Analyse von Datei-Zugriffen zur Potentialermittlung fuer Information Lifecycle Management, TU Darmstadt KOM Technical Report 01/2005
- [Tu05] Turczyk, L.A.Information Lifecycle Management: Organisation ist wichtiger als Technologie, InformationWissenschaft & Praxis 2005, Heft 7, S. 371-372
- [Tu06] Turczyk, L., Berbner, R., Heckmann, O., Steinmetz, R., (2006) "A formal approach to Information Lifecycle Management". In Proceedings of 17th Annual IRMA International Conference, Washington D.C.