A Formal Approach to Information Lifecycle Management

Lars Arne Turczyk, Oliver Heckmann, Rainer Berbner, Ralf Steinmetz
TU Darmstadt
KOM Multimedia Communications Lab
Merckstr. 25
64283 Darmstadt, Germany
Phone: +49 69 797 2778
Fax: +49 69 797 3437
e-mail: lars.turczyk@siemens.com

December 8, 2005

Abstract

In this paper we present a framework for simulating Information Lifecycle Management (ILM) scenarios. The framework is derived from a formal approach to ILM which offers concrete mathematical terminology.

1 Introduction

Information Lifecycle Management (ILM) is one of the great trends in the context of information storage. It has its roots in hierarchical storage management (HSM), which was popularized with mainframe storage management strategies in the early 1980s. ILM is a process for managing information through its lifecycle, from conception until disposal, in a manner that optimizes storage and access at the lowest cost. ILM is based on the idea that in an enterprise there are different information with different values. The different information will be stored on different storage devices. A similar way of thinking is found in the area of operating systems at page swapping scenarios using virtual memory: RAM memory is more expensive than hard disc memory, therefore currently unused memory pages are swapped to the hard disc when memory becomes scarce [1]. The same principle is employed with ILM for storage systems.
ILM manages information according to its value. Valuable information is stored on systems with high Quality of Service (QoS) [2]. The value changes over time and therefore migration of information to cheaper storage systems with lower QoS is required. Automated migration makes ILM dynamic. By correctly establishing migration rules, the organization would see little to no delay in information access (keeping frequently accessed information or data requiring instant access regardless of age nearline), but would save significantly by conserving precious disk subsystem space and eliminating disk subsystem purchases to support growth.

In this paper we build a framework for testing the quality of migration rules by simulation.

2 SNIA's Definition of ILM

ILM as a concept is not easy to handle. Therefore in 2004 the Storage Networking Industry Association (SNIA) gave a new generally accepted definition [3]:

Definition 1 (ILM SNIA) *Information Lifecycle Management is compromised of the policies, processes, practices, and tools used to align the business value of information with the most appropriate and cost effective IT infrastructure from the time information is conceived through its final disposition. Information is aligned with business processes through management policies and service levels associated with applications, metadata, information and data.*

This definition forms the basis for an accurate occupation with ILM. Nevertheless it is general and has limitations when applied to specific cases. Therefore a formal approach will lead to specific results for the employment of ILM as we will show.

We start with deriving a mathematical approach for defining ILM. In the mathematical environment the simulations will be prepared.

The following definition is not meant as a contradiction to the SNIA definition. It aims to help creating a best practices framework for ILM and ILM evaluations.

3 Formalized Definition of ILM

The formalized definition of ILM is derived canonically. To get a common understanding we refer to SNIA's definition of information [3]:

Definition 2 (Information) *Information is data that is exchanged, expressed or represented within a context such as an application or a process.*

This means the application offers the context for data.

Information in ILM has a granularity and the number of information in an enterprise is finite. Depending on the storing task the granularity can be, for example, a file, a database table, an email inbox or an object that is shared between applications that participate in a business process.

An access is based on operational reasons. This shows that the information has a certain importance for business. This importance is defined as value of the information.

Definition 3 (Value of Information) *The value of information V(I) describes the importance of the information I for the business. The value of information can be expressed in money. V(I) is initiated with the creation of the information I. The time of creation is defined as* $t_0 \geq 0$.

The value of an information changes over time. It is a function of t: V(I(t)). Therefore it is necessary to migrate the information during its lifecycle to adequate storage systems.

How is the value determined? This is a complex question in ILM. The easy answer is: It depends on the the business processes. To be more specific it comes, for example, from the administrator, the end user or the CIO or from the application. External regulations and laws can determine the value, too, e.g. Sarbanes Oxley Act (SOA). Furthermore the value can be derived from the usage of information. Files with many accesses are more valuable than files with few accesses, which are more valuable than files with no accesses.
Therefore observing access patterns is one way to determine the value. There is a long tradition in looking for access patterns [4][5].
In section 4 we refer to our own study, which was conducted in 2005 [6].

When the value is determined, the information are grouped into information classes according to their value. All items of an information class have similar values. Values change over time, so the constellation within an information class varies. It is dynamic. One strength of ILM is to take this dynamic into account. Information class can be defined as follows:

Definition 4 (Information Class) *An information class C is a set of all information* I_1,...,I_m, *whose values* $V(I_i(t))$ *lie at the time t in a predefined (value-)interval.*

$$C := C_{i,j} := \{I_i(t_j) \mid a \leq V(I_i(tj)) < b; a, b \in \mathbf{R}\}$$

An information class is a set of information which have similar values. "Similar" means the value lies within the interval $[a, b)$. Different information classes have different (disjunct) intervals.
The intention is to store the content of an information class on the same type of storage devices. Therefore all storage devices in the enterprise are grouped, too.

Definition 5 (Storage Class) *A storage class S is a set of storage devices with similar properties, i.e. Quality of Service (QoS) and cost.*
QoS summarizes especially security, backup frequency, access speed [7].

The storage classes represent the hierarchies in the ILM solution. The information classes are mapped onto the storage classes. It is a fix mapping which does not change after being established.

In chapter 4 we pick up on this point and show how information classes can be defined and how the mapping can be arranged.

The content of an information class is not static. The value of each information might change beyond the value interval $[a, b)$. Then an information becomes an element of a different information class which is mapped to a different storage class. The information will be migrated. That is the migrating process of ILM.
The changes in value of an information are dynamic and define a "lifecycle". The following definition puts the lifecycle in a formula.

Definition 6 (Lifecycle) *Let $0 < t_1 < t_2$. The lifecycle L of information I is the mapping of the value of I between time t_1 and t_2.*

$$L(I) := \{V(I(t)) \mid t_1 \leq t \leq t_2\}$$

The change in the value of information during a period of time represents a lifecycle.

The lifecycle is dynamic for each information. The question of administrators is "What happens to the system when all the dynamic lifecycles are reflected to the storage environment?". To avoid a bad experience on the real storage environment the dynamic behaviour of ILM has to be simulated.

Before simulating we summarize the formal approach in a formal definition of ILM.

Definition 7 (ILM) *Information Lifecycle Management (ILM) is the mapping of the information $I_1, ..., I_n$ on Classes $C_1, ..., C_m$ according to their values $V(I_1), ..., V(I_n)$ in time interval $[t_1, t_2]$.*

With the derived formalism the framework for simulation is established.

4 Application of the formal approach

ILM is a dynamic process affecting the whole IT. The effect of an ILM solution depends on the quality of the migration rules. To improve the quality of the rules simulations are helpful. The formal definition offers the framework for simulations.

Step 1: V(I), the value of information I is determined.

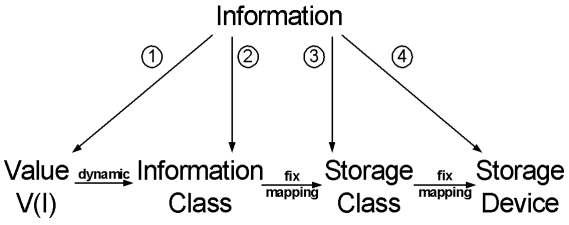

Figure 1: Framework for simulations

Step 2: I becomes element of an information class according to its value.
Step 3: The determined information class has a fixed related storage class.
Step 4: I is stored on a storage device related to the determined storage class.

Steps 2 and 3 refer to a fix mapping. They were inserted in order to cluster information and storage devices. When dealing with a great number of information it is easier to apply migration rules to the whole set of information than to single information.
What is a migration rule exactly? A migration rule takes the value of an information and determines whether a migration is to be executed or not. In the framework the migration rule is a part of the information class. By defining the value interval $[a, b)$ of an information class the migration rule is defined, too. Therefore the migration rule is part of determinating the value, too. The more complex the rule, the more complex the metric of the value and vice versa. Again, valuation is not easy. The framework can handle both simple and complex valuations.
Now we will apply the formal approach to a real database. We start with a case study made in 2005.

4.1 Case study

In a case study on a database we provided following results [6]:
There were more than 150,000 files on the system and 89 percent of them were not accessed 90 days after creation.

The intention is to create an ILM concept with three different storage classes. The files shall be migrated between the three hierarchies automati-

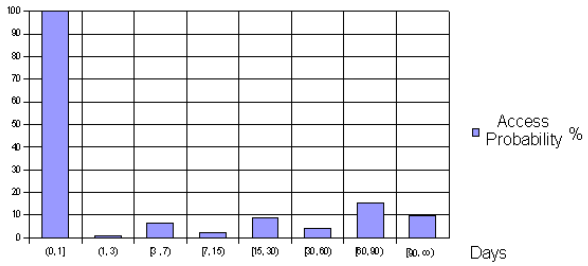

Figure 2: Access probability

cally. The task is to find an adequate and efficient set of migration rules.
To give an impression how specific rules can be evaluated we show the process for testing the rule "Move information when it is not accessed for a period of time (90 and 200 days)."

4.2 Application of the formal approach to the case study

The framework with its steps 1-4 is amended by step 0, the assessment, and by step 5 the simulation.

Specific case:
Step 0: There are 150,000 files from different office applications on a database.
$I_1, ...I_{150000}$

Step 1: The analysis tells us that 89 percent are not accessed 90 days after creation. The resulting valuation metric is "Information accessed within the last 90 days has high value. Information not accessed for 90 days has medium value. Information not accessed for 200 days has low value."

Step 2: There are three information classes:
Class 1: High value,
i.e. maximum days of not being accessed < 90 days
Class 2: Medium value,
i.e. maximum days of not being accessed < 200 days
Class 3: Low value,
i.e. maximum days of not being accessed ≥ 200 days

Step 3: There are three different storage classes with the QoS attributes high, middle and low. They are fix mapped to the information classes with high, medium and low value.

Step 4: The rule says: If a file is created it is stored on the high storage class 1 (high QoS). If it is not accessed for 90 days it is migrated to the storage class 2 (medium QoS). If it is not accessed for 200 days it is migrated to storage class 3 (low QoS). If it is accessed when stored on storage class 2 or 3, then it is migrated to storage class 1.

Step 5: In the simulation all the 150,000 files are stored on storage class 1 to begin with. A distribution function of access behaviour is assigned. The accesses are simulated on a daily basis for a lifecycle of 1,000 days. Depending on the simulated accesses the rule is executed. After the simulation the migration log of each file is analysed.
We look at the "migration jitter", ie. migrated files which were migrated downwards, then upwards, again downwards etc. A correctly established migration rule should have almost no jitter because migrating files bring about costs (e.g. bandwidth). Hence the rule "Move files not accessed for 90 days", is obvious, but it creates too much migration jitter.
A more effective rule is "Move Microsoft Powerpoint files not accessed for 90 days". It is a combination of "access" and "application". The rule could be extended for example to "user" or "file size". Doing this the rule set becomes more complex. The more complex the rule, the greater the need for simulations with a flexible framework.

5 Summary and Conclusion

In this paper we presented a formalized definition of Information Lifecycle Management (ILM). The quality of an ILM solution depends on the quality of migrating rules. Simulations improve the quality of the rules. The formal approach offers the mathematical base for implementing simulation routines.

6 Outlook

Our next step will be the simulation of actual ILM systems in order to get reliable statements for mi-

grating information. Furthermore the case study will be extended to derive a distribution function for file accesses.

Analytical models are planned to analyse the cost saving potential of ILM. The analysis will be extended by simulations.

References

[1] A. S. Tanenbaum, *Modern Operating Systems*, Prentice Hall, 2nd Edition, February 2001

[2] O. Heckmann, *A System-oriented Approach to Efficiency and Quality of Service for Internet Service Providers*, Submitted PhD Thesis, TU Darmstadt, December 2004. http://elib.tu-darmstadt.de/diss/000522/

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

[4] M. Satyanarayanan, *A Study of File sizes and Functional Lifetimes*, Proceedings of the 8th Symposium on Operating Systems Principles, Association of Computing Machinery, 1981

[5] S. Strange, *Analysis of Long-term Unix File Access Patterns for Application to Automatic File Migration Strategies*, University of Berkeley, 1992

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

[7] J. B. Schmitt, *Heterogeneous Network Quality of Service Systems*, Kluwer Academic Publishers 2001

[8] G.E. Moore, *Cramming more components onto integrated circuits*, Electronics, volume 38, number 8, 1965.

[9] A. Lyman, B. Varian, *How Much Information?* 2003, University of California, October 2003 http://www.sims.berkeley.edu/research/projects/how-much-info-2003/

[10] D. Fletcher, K. Elliott, *Benchmark and Trend Analysis*, Division of Information Technology Services, State of Utah, October 2003

[11] T. Kraemer, J. Berlino, *The Storage Report - The customer Perspectives & Industry Evolution*, McKinsey & Company, June 2001

[12] F. Moore, *Storage - New Game New Rules*, Horison Information Strategies, 2003 http://www.horison.com/horison/books/2004/

[13] H. Nguyen, *IDC's Worldwide Disk Storage Systems Quarterly Tracker*, IDC, March 2005 http://www.idc.com/getdoc.jsp?containerId=pr2005-03-03-154203

[14] R. Paquet, *Why You Need a Storage Department*, Gartner Research, June 2004