Universität Augsburg

Efficient and Flexible Multimedia Delivery with Universal Database Systems

Matthias Wagner, Stefan Holland, Werner Kießling

Report 1999-02 August 1999

Institut für Informatik
D-86135 Augsburg

Institut für Informatik
Universität Augsburg
D–86135 Augsburg, Germany
http://www.Informatik.Uni-Augsburg.DE

Efficient and Flexible Multimedia Delivery with Universal Database Systems

Matthias Wagner, Stefan Holland, Werner Kießling

Universität Augsburg

{wagner, holland, kiessling}@informatik.uni-augsburg.de

Abstract

Today most commercial database management software is extended to store and manage multimedia objects beside standard alphanumeric data. In this paper we study how such universal database systems can be used for implementing an adaptive multimedia digital library. We present a framework for efficient and flexible delivery of multimedia content based on the idea that multimedia objects are often stored redundantly to support broadest system access for diverse clients from heterogenous environments. There is a large number of alternatives for universal database systems to store and deliver multimedia data, since storage formats are not independent from each other but heavily interrelated by conversion tools. Partitioning the data formats into those that are physically stored in the database and those that are converted into a delivery format on-demand poses a nontrivial optimization problem. The framework presented here tackles this problem and has been implemented and evaluated with the commercial database management system DB2 Universal Database from IBM. Beside the core implementation, a graphical database administration tool has been developed to leverage the automated or semi-automated application of the optimization framework. First experiments with the implemented tools have proved substantial savings in the total costs.

1 Introduction

Modern universal database systems like the DB2 Universal Database [Cha96, CCN+99], Informix Universal Server [Bro99], Oracle 8i [KBN99] and research prototypes [SP97, SM96] claim to easily take common databases beyond ordinary numbers and text. Their capability to integrate multimedia data and traditional data in a single query widens the application areas of database systems including web sites for dynamic multimedia content delivery, interactive kiosk-systems, digital multimedia libraries, etc. Hence, multimedia database systems are hot topics in database research and system development [ÖRS97, ÖRS95, Sub98, SSU96, AK97].

Designing a multimedia system is a critical task [Sub98, AT96]. The choice of multimedia database technology for digital libraries is typically influenced by the intended users and uses of the system. Groups of users of digital libraries are often segmented by issues (e.g., heraldry, contemporary paintings, architecture), by function or role (curators, art historians, registrars, conservators, students, internet users) or by system purpose (facsimile or mobile delivery, browsing, research, analysis). Both, current users and uses, and anticipated future users and uses, should be considered when setting up a library. The number of these users and the types of access profiles must be anticipated (at start-up time, at full-scale implementation and later on), the location of users must be considered (all in one place, dispersed within a building, widely dispersed on a campus or throughout the world) and the technological capabilities of user workstations (operating systems, internal memory, storage, display quality, networking capability and speed) must be taken into account. Care must be taken to develop a strategy that minimizes limitations, that does not foreclose future options and that offers a likely upgrade path. Since, adapting a database system manually to all the given constraints is in practice not feasible, multimedia database systems should support appropriate mechanisms for self-tuning and self-organization [Spi98, CN98a].

In particular developments in Graphical User Interface software for navigating the internet and the world wide web, such as Netscape Navigator, Internet Explorer or specialized Java-GUIs have made it possible to offer wide access to multimedia databases. Providing access to multimedia archives using these tools, however, requires multimedia-storage formats to match the capability of internet browsers. Especially in the field of large image archives, where system architectures and network topologies become significant concerns, care must be taken in identifying the file formats to be stored and delivered [FSZ95, AWA+96]. Compromises in one area often affect performance in another. Therefore, trade-offs between storage cost and capacity on one hand and access speed on the other must be examined carefully when setting up a networked image database.

However, the image data stored and delivered by universal database systems is often redundant, because storage formats are typically interrelated by conversion tools. They may only differ in aspects such as compression, color depth and resolution. In this paper we propose an extension to the storage and delivery of images in todays multimedia databases. Our approach enables the dynamic optimization of multimedia document delivery and allows the efficient and flexible application of universal database systems.

This is done by giving the option either to physically represent a multimedia object or to compute it from others. Given a set of possible conversions between multimedia objects there is generally more than one way of partitioning the formats into physically stored and computed ones. By applying a cost function the database server determines its optimal choice considering specialized aspects. This optimization is an automated complex task which can be performed periodically.

While the use of database systems in large business applications is quite common, their use in the humanities is still exceptional. The work presented here is applied in the HERON system, a digital library for art historical image material [KEUB+98, KUB+99]. One particular focus in the work of the HERON group is on the complex problem of delivering multimedia documents, in particular images, in a large variety of formats and quality levels over networked environments — a problem that is also apparent in other projects from the humanities [SP99].

The rest of this paper is organized as follows: We motivate the need for multimedia delivery by sample applications in section 2. In section 3 we describe multimedia formats and their interrelations in general and point out how multimedia data is managed by the DB2 Universal Database system. Moreover, we present a formal model for the representation of multimedia formats and conversion constraint as the basis for the further discussion. Section 4 displays the optimization algorithm and examines reasonable cost functions based on the core formal model. Section 5 gives an insight to the implementation and evaluates the whole framework against the sample applications. Finally section 6 concludes with a summary of the contributions of this paper and sketches possible extensions.

2 Sample applications

In this section we motivate our approach for the dissemination of multimedia objects, in particular images, in heterogenous or networked environments like the World Wide Web, by two different applications. At first, we present the storage layer of the HERON system, an enhanced database application for art historical image collections. As a second example application we propose an integrated web and fax service for the distribution of stock charts.

Note that the two applications introduced here, while quite different in their technical nature and application area, share an important common property of modern digital library services: The capability of delivering multimedia objects in a variety of formats and at a diversity of quality-levels.

2.1 Electronic dissemination of art historic images

The delivery of art historic multimedia material over communication networks will make cultural heritage available to a wider audience than ever before. Namely innovations in digital imaging technology have the potential to change the nature of teaching and research. For these transformations to take place, however, a large amount of digital information about works of art must exist, and it must be available online in standard forms and formats.

The University of Augsburg has launched the interdisciplinary project HERON (Heraldry Online), committed to creative use of information science and technology to advance the study of arts and humanities [KEUB+98, KUB+99]. This project investigates the impact of multimedia applications of the humanities, in particular heraldry, on future database technology and explores ways of applying computer technology to improve scholarly access to art historical information. Areas of particular concern include

- the retrieval of images based on their visual content,
- the integration of image databases with other information sources and museum projects
- and the exchange of visual information among computerized systems.

Databases for the arts and humanities pose outstanding demands to multimedia data management, since they must support various types of users from extremely heterogeneous environments at different stages of work: Small, low-resolution browse or thumbnail images may be shown quickly to enable the viewer to identify a work of art, or review large collections; a "medium-resolution" full-screen image may accomplish a full catalog record; and a "high-resolution" image may be available for detailed study only on special request. Images in a particular format may be sufficient for classroom use by undergraduate students, but contain too little information for a conservator exploring the technical construction of a work, e.g. a 256-color GIF image may be adequate for recognition purposes, but too inaccurate to support comparative analysis of an artist's palette.

A sample user interaction with the HERON image database is shown in figure 1. This scenario is a compilation of the typical stages of work an art historian passes during her/his interaction with the heraldic image database. At an abstract level these stages of work typically comprise the following activities:

Searching&Browsing: In a first step queries are posed against the heraldic database. These queries can be based on the visual content of images or on conventional features like keyword and full text, respectively. The advanced visual retrieval capabilities or fuzzy full text search typically return a ranked set of data records, including images, as a query result. Therefore, further analysis and refinement is often necessary to reduce the size of the initial set retrieved. During this stage images are normally displayed as thumbnails in browseable formats like GIF, JPEG or PNG.

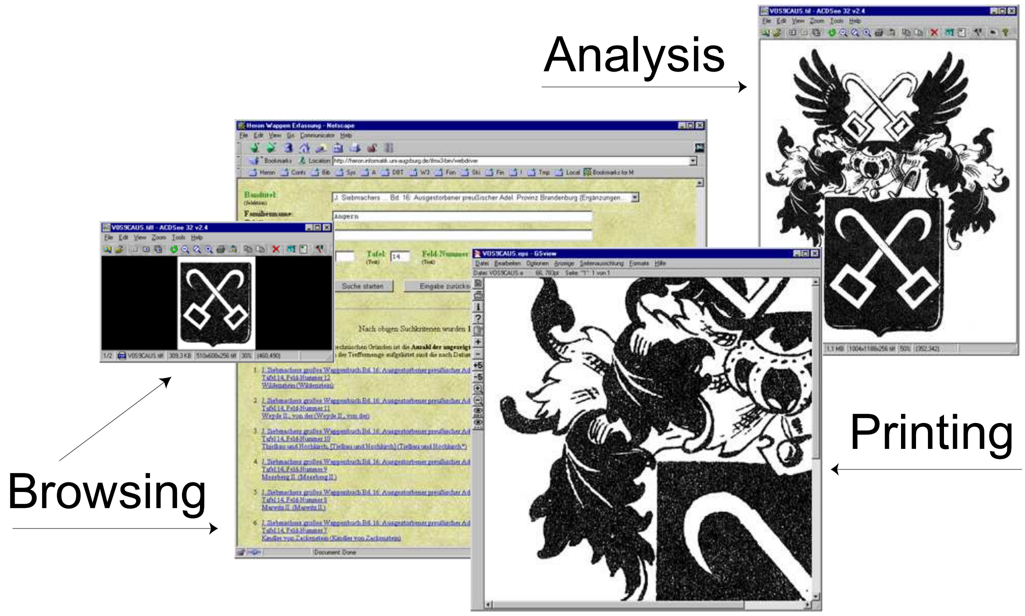


Figure 1: Typical user interaction with the HERON system.

Zooming&Analysis: During the examination of result sets, images of special interest might be required for further considerations at higher resolutions. Because not all image-file formats can be read by all viewers, a large variety of image formats at various levels of quality must be potentially deliverable by the HERON system to support broadest access and use. Some typical image formats in this stage of work are TIFF, GIF or PNG at diverse resolutions.

Storage&Printing: If search and analysis end with a final set of relevant images, these image files are transmitted to the client for user-site storage and later reference or for printing in high quality. In general only lossless formats at very high-level quality or typical print formats are appropriate for this task. Adequate formats include TIFF, PDF or Postscript.

The cost-based evaluation of our approach towards a flexible multimedia delivery in section 5 will be based on the concrete scenario of varying access profiles at the HERON image server depicted in figure 2. This diagram shows a scenario which models the typical user behaviour mentioned above: Searching and browsing, zooming and analysis, storage and printing. In the illustration x- and y-axis represent time and access profiles, respectively. The time axis is labeled with abstract times $t1$, $t2$, $t3$ etc. Note that we omit the decorations of all even points in time ($t2$, $t4$ etc.) to keep the illustration compact. Columns associated with the y-axis represent the amount of image requests at a given time. For example, at $t3$ 450 images are accessed, 100 of which are requested as GIF-75, 150 as GIF-50 and 200 in GIF-25, i.e. GIF images in resolutions of 75%, 50% and 25%.

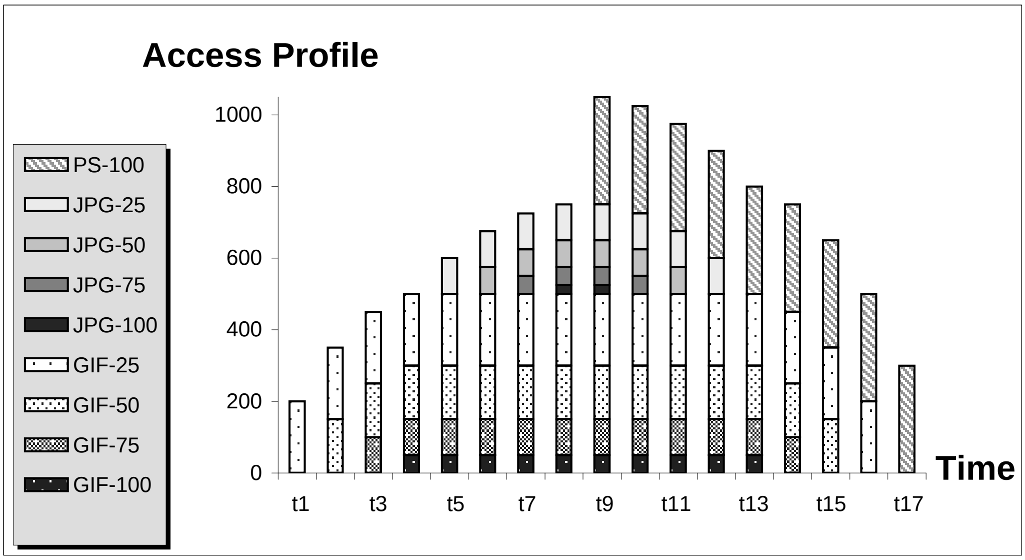


Figure 2: Sample access profile variation for the HERON image database.

2.2 Internet/fax distribution of stock charts

The internet and other modern communication channels are accelerating a personal finance revolution. For the first time, private investors have most of the tools and information they need to handle their own personal finance and investing decisions, and need not necessarily rely on professionals to make decisions.

Banks and financial institutions transform the web into a powerful tool that helps to take control of the personal financial future. However, especially the banking business is forced to provide news also through classical information channels. Since the combination of a fax service with an internet site does seem common [NQ99, AzW99], we study a combined internet/fax service for stock charts as a second test scenario. The service integrates the delivery of charts in typical web-formats as well as in standard IPS facsimile-formats at different compression levels [Tec99] and the Postscript format. In contrast to the scenario considered above this example does not yet represent a real-world application but is nevertheless potentially applicable in the financial industry where database services are considered mission critical [Sha97].

The sample scenario in figure 3 is interpreted analogously to the above example. In section 5 these sample scenarios will be the platform for the evaluation of the presented optimization framework.

3 Towards multimedia delivery with DB2 UDB

Today commercial database systems claim to be open, scalable, easily manageable and fully extensible, providing essential flexibility for virtually any application area. Whether

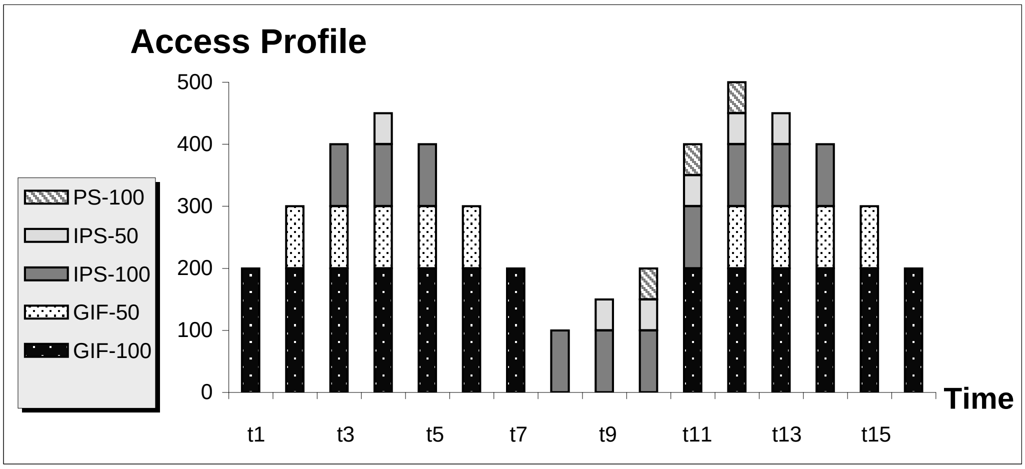


Figure 3: Sample access profile variation for internet/fax distribution.

for data warehousing, web content delivery, or multimedia digital libraries every major vendor claims to offer the perfect database platform.

IBM is one of the global players in the emerging market of multimedia database software. Besides serving traditional database tasks their product DB2 Universal Database (UDB) is sold as a tool for building, deploying and monitoring web database applications and content-driven web sites as well as multimedia libraries. Like a standard database system UDB stores and manages numeric and character data. In addition the system leverage multimedia applications by providing the functionality to store and retrieve multimedia data like image, video, audio and text in large, complex objects. Extra features like content-based image retrieval are bundled in so called DB2 Relational Extenders, which define distinct data types and specific functions for multimedia objects. At the application level these extra data types and functions are available through SQL.

Intended as a platform for multimedia digital libraries, DB2 UDB comprises a separate Audio Extender, Video Extender, Text Extender and Image Extender. The DB2 Image Extender [IBM99] provides database functionality for the storage of images as well as the capability to query images by their visual content [FSN$^+$95]. The latter feature, while used very extensively within the HERON project, is not explored any further here but is examined elsewhere [UBK99]. Instead we put the focus on UDB's possibilities for storing, manipulating and delivering images in a large variety of formats at arbitrary levels of quality.

This section further motivates the general problem of format interrelation in multimedia database systems. We have implemented a dynamic optimization of multimedia document delivery on top of UDB's Relational Extenders, providing the option either to physically represent a multimedia object or to convert it from other ones. In the sequel, we present the conversion functionality of the DB2 Image Extender and show how to restrict it to a reasonable set of conversion constraints.

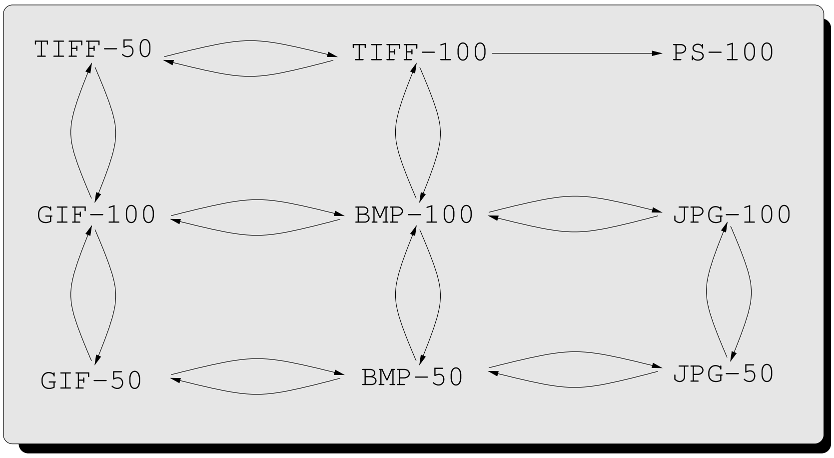


Figure 4: Subset of format interrelations of DB2 UDB.

3.1 Delivery along reasonable conversion paths

Images are often stored at much higher quality than they would be needed for display on a monitor because most printing devices are capable of a much higher resolution than screen displays. Image quality is primarily the cumulative result of the scanning resolution and is often expressed in terms of scaling. In this sense we describe formats at certain quality levels relative to the original image scan-size, e.g. TIFF-100 denotes the format of an image at its original scan size (100% scaling), whereas TIFF-25 is the format of images that are scaled down by the factor of four.

The DB2 Image Extender provides fifteen popular image formats for the storage and delivery of image objects. This includes for instance the widely used Compuserve GIF format, all TIFF formats, the OS/2 and Microsoft Windows BMP format, the JPEG format or PostScript (PS) at level I and II. In addition, image formats can be scaled to virtually any resolution. Figure 4 depicts only a small subset the formats that can be stored and delivered by UDB and their interrelations by the conversion functionality of the Relational Extender.

In digital libraries committed to high quality service not all possible options for image conversion delivered with tools like DB2 Image Extender are equally useful. In particular, care must be taken to avoid lossy format conversions. We consider the following types of image conversions as potentially lossy:

1. The conversion of an image in a low resolution to a high resolution, e.g. converting an image of format GIF-25 to GIF-100.

2. The conversion of an image in a lossy format to an image in a lossless format when the target format could be equally derived from an alternative lossless source format. E.g. in the conversion options from figure 4 it is not a good idea to convert

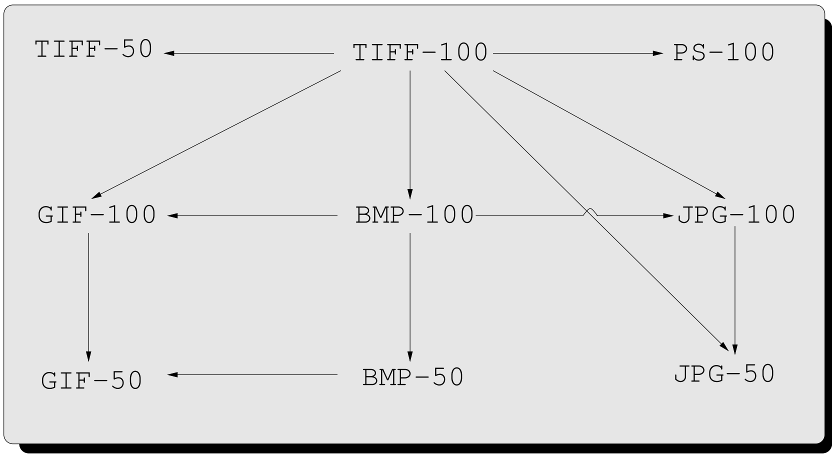


Figure 5: Subset of UDB's reasonable format interrelations.

GIF-100 from JPEG-100; instead TIFF-100 can be used as the primary source format.

We claim that a digital image library should assure high quality of service, i.e. the library should avoid lossy format conversions. In our implementation built upon UDB, this is achieved by explicitly modeling the reasonable conversion constraints. As an illustration figure 5 displays a subset of the format interrelations from figure 4 which represent reasonable conversion constraints of the DB2 Image Extender.

3.2 Cost-optimized delivery

Choosing appropriate image formats and storage-format quality for a digital library requires balancing user needs, system requirements, network infrastructure, and cost. We claim that a digital image library must potentially provide image data in a variety of formats and resolutions. Naively, one of the following approaches could implement this.

On-demand image conversion: One simple strategy for offering all images of an archive in any format and at any quality level is to store images in a minimal set of source formats which allow the conversion of all potentially deliverable target formats. In this approach an image which is not physically available in the image base would be converted into the requested target format at run-time.

However, if the initial set of images comprises high-resolution digital-image files which are very large — a 24-bit full-screen image, e.g. in the TIFF format, to be displayed on a workstation can easily occupy 6 megabytes — the explicit online conversion of derived small versions of these images, like for instance thumbnails, can be computational

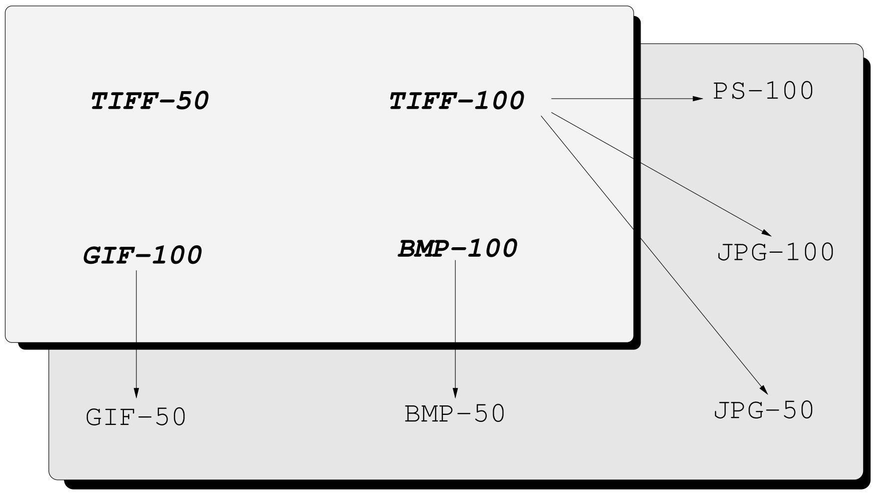


Figure 6: Possible interrelations of stored and computed formats.

expensive: We measured about 15 seconds for the conversion of such an image into a GIF thumbnail on a high-end workstation (for a description of our testbed see section 5.1).

Off-line image conversion: On the other hand, physically storing all images at all required quality levels in the database for direct and fast delivery from disk does not seem an appropriate solution either.

In most applications this approach might exceed the available mass storage. In addition, limiting the image database to a certain amount of deliverable formats would clearly foreclose future upgrades of the library for new multimedia technology.

A practical implementation supporting optimized image delivery must be capable of managing the trade-off between on-demand and off-line image conversion.

Optimized image delivery: The framework presented in this paper both reduces the data volume and allows for a flexible database design by providing a cost-optimal solution to the above trade-off. The tackled optimization problem which is ubiquitous in multimedia database systems can informally be stated as follows:

> *Given a set of interrelations between the objects of a multimedia database server, what is an optimal partition of the objects formats into those that are physically represented and those that are computed from the physically represented ones?*

To determine an optimal choice of stored image formats is a non-trivial optimization problem subject to parameters such as query profile, available disk storage, server load and network bandwidth. As an example, figure 6 displays a possible configuration of a multimedia storage server. In this setting TIFF-50, TIFF-100, GIF-100 and BMP-100 are selected as the physically represented image formats, whereas other formats like PS-100 or JPG-50 are computed on-demand.

3.3 Formal optimization framework

The implemented framework is based on the formal model for multimedia attributes (formats and resolutions) and conversion constraints presented in the sequel. The following is a refinement of the more general notations and algorithm proposed in [KKK97].

The storage format of an image together with its resolution is modeled as a multimedia attribute.

Definition 1 (Multimedia attribute)
Let M be a multimedia format, and $\mathcal{R}_M = [R_1, R_2, \ldots, R_n]$ a list of the supported resolutions for format M in ascending order. For $R \in \mathcal{R}_M$ the pair of format and quality level (M, R) denotes an attribute*.*

Note that in the example figures above multimedia attributes have already been informally denoted as *M–R*, e.g. TIFF-100.

Format interrelations between images are modelled as functional constraints. In the following let $\mathcal{A}$ denote a set of attributes.

Definition 2 (Functional constraints)
Let $A, A' \in \mathcal{A}$ be attributes, $\mathcal{F}$ a set of conversion constraints associated with $\mathcal{A}$ and $f \in \mathcal{F}$ a conversion function. Then $fc \equiv (A = f(A'))$ is called a functional constraint.

A and A' are denoted as head *and* body attribute*, respectively. We say that attribute A* depends directly *on attribute A' and denote the direct dependency as $A \stackrel{\mathcal{F}}{\longleftarrow} A'$. The transitive closure, $\stackrel{\mathcal{F}^\star}{\longleftarrow}$, of this relation is denoted as* dependency relation*. A set of functional constraints $\mathcal{F}$ is named* acyclic *if no attribute depends on itself.*

In the sequel let $\mathcal{F}$ be a set of constraints on $\mathcal{A}$. A functional base of $\mathcal{F}$ is a subset partitioning the attributes such that some are physically represented and the rest can uniquely be computed by conversion constraints.

Definition 3 (Functional base)
Let $\mathcal{F}_{comp}$ be a subset of $\mathcal{F}$. $\mathcal{F}_{comp}$ is called a functional base *iff:*

1. *$\mathcal{F}_{comp}$ is acyclic.*
2. *For every $A \in \mathcal{A}$ there is at most one $fc \in \mathcal{F}_{comp}$ such that A is the head of fc.*

$\mathcal{A}_{comp}$ is the set of head attributes of $\mathcal{F}_{comp}$, and $\mathcal{A}_{phys} := \mathcal{A} \setminus \mathcal{A}_{comp}$ are the remaining attributes. The attributes $\mathcal{A}_{phys}$ must be physically represented and the attributes $\mathcal{A}_{comp}$ are computed from attributes in $\mathcal{A}_{phys}$.

The optimization problem can now be stated formally as:

Determine a functional base optimal w.r.t. a cost function!

4 Search algorithm and cost considerations

In this section we sketch the search algorithm for an optimal partition of formats. Furthermore, we show how the search space can be narrowed to speedup the computation.

By applying a cost function the digital library server determines its optimal choice considering aspects such as storage consumption, network bandwidth, and complexity of the conversions. This optimization is a non-trivial task that cannot be done manually and must be performed periodically by the system depending on statistical data about the varying access profile at the storage server.

4.1 Optimization algorithm

Functional bases are evaluated w.r.t. a cost function. The optimal functional base is defined as the base with minimal costs.

Definition 4 (Optimal functional base)
Let cost be a cost function. A functional base $\mathcal{F}_{opt}$ is called optimal *iff*

$$cost(\mathcal{F}_{opt}) = \min\{cost(\mathcal{F}_{comp}) \mid \mathcal{F}_{comp} \textit{ is a functional base}\} \enspace .$$

Basically, the optimal partition of a given set of multimedia attributes and the associated conversion constraints can be determined brute force by evaluating all potential functional bases [KKK97]. However, the high complexity of this basic optimization algorithm seems prohibitive for large database designs: For $n = |\mathcal{A}|$ the proposed algorithm has a worst case complexity of $O((n+1)^n \cdot n^2) = O(n!)$.

To reduce the computational expense, the search can be restricted to some potential functional bases which appear to be reasonable and promising. This approach is supported by the notion of a minimal attribute and some reasonable assumptions.

Definition 5 (Minimal attribute)
Attribute $A_{min} \in \mathcal{A}$ is named minimal *iff for all $A \in \mathcal{A} \setminus \{A_{min}\} : A \stackrel{\mathcal{F}}{\leftarrow} A_{min}$.*

We further state the following assumptions on the given multimedia formats and conversion constraints.

Assumption 6
Given a set of multimedia formats $\mathcal{A}$ and a set of associated conversion constraints $\mathcal{F}$ we assume that:

1. *A linear order on the attributes associated with a single format is induced by* $\mathcal{F}$*, i.e. given* $(M, R_i), (M, R_j) \in \mathcal{A}$ *with* $R_i < R_j$ *then* $(M, R_i) \leftarrow^{\mathcal{F}} (M, R_j)$*.*

2. *For* $A, A' \in \mathcal{A}$ *such that* $A \leftarrow^{\mathcal{F}} A'$*, the dependency of* A *on* A' *is associated with exactly one functional constraint.*

3. *There exists a minimal attribute* $A_{min} \in \mathcal{A}$*.*

To the best of our knowledge these assumptions are fulfilled by most common image conversion tools and image read/write database extensions [IBM99, Vir99, Exc99]. For the DB2 Image Extender the set of possible format conversions can be naturally condensed to fulfill the restrictions. Note that the minimal attribute A_{min} is capable of deriving all formats and intended to be an attribute that must be physically stored in the database.

```
restrictedFb(A_min, 𝒜_phys, 𝒜, ℱ) : ℱ_comp
 begin
      ℱ_comp := ∅;
      foreach A ∈ (𝒜 \ 𝒜_phys) do
           (A_pred, f_pred) := predecessor(A, 𝒜_phys, ℱ);
           if f_pred ≠ null then
                insert(f_pred, ℱ_comp);
           else
                insert((A = f(A_min)), ℱ_comp);
           fi
      done
 end

predecessor(A, 𝒜_phys, ℱ) : (A_pred, f_pred)
 begin
      A_pred := f_pred := null;
      (M, R) := A;
      while R < last(ℛ_M) do
           R := next(R, ℛ_M);
           if (M, R) ∈ 𝒜_phys then
                A_pred := (M, R); f_pred := (A = f(M, R));
                break;
           fi
      done
 end
```

Figure 7: Compute a restricted functional base w.r.t A_{min}, $\mathcal{A}_{phys}$, $\mathcal{A}$, $\mathcal{F}$.

Algorithm 7 (Restricted functional base)
Let $\mathcal{A}$ be a set of multimedia attributes, $A_{min} \in \mathcal{A}$ a minimal attributes, $\mathcal{A}_{phys} \subseteq \mathcal{A}$ some physically represented attributes, and $\mathcal{F}$ the set of associated conversion constraints. Then the algorithm depicted in figure 7 determines a restricted functional base.

Given the assumptions from above the algorithm computes a functional base according to definition 3. The base is "restricted" in the sense that every format at a certain resolution which is not directly stored in the database is preferred derived from the same format physically stored in the next higher resolution, or otherwise from the minimal attribute.

The algorithm restrictedFb is conceptually split into the main part and the subroutine predecessor which, given an attribute A, searches the set $\mathcal{A}_{phys}$ for the format A in the next higher resolution. Moreover, basic functions on lists are assumed: Let l be a list, e a list element. Then $last(l)$ will determine the last element of l. $next(e, l)$ computes the direct predecessor of element e in l and $length(l)$ calculates the length of list l. E.g., let $l = [25, 50, 75, 100]$, then $last(l) = 100$, $next(50, l) = 75)$ and $length(l) = 4$.

```
optimalFb(A_min, 𝒜, ℱ, cost): (ℱ_opt, min)
 begin
     ℱ_opt := ∅; min := cost(∅);
     foreach 𝒜_phys ∈ 𝒫(𝒜 \ {A_min}) do
         ℱ_comp := restrictedFb(A_min, 𝒜_phys, 𝒜, ℱ);
         if cost(ℱ_comp) ≤ min then
             ℱ_opt := ℱ_comp; min := cost(ℱ_comp);
         fi
     done
 end
```

Figure 8: Compute an optimal functional base w.r.t. A_{min}, $\mathcal{A}$, $\mathcal{F}$, *cost*.

An optimization algorithm can now be easily defined. The algorithm exhaustively searches all possible attribute partitions, associates them with their restricted functional base and evaluates them w.r.t. a given cost function to determine the optimal base.

Algorithm 8 (Optimization algorithm)
Let $\mathcal{A}$ be a set of multimedia attributes, $A_{min} \in \mathcal{A}$ a minimal attribute, $\mathcal{F}$ the set of associated conversion constraints, and cost a cost function. Then the algorithm depicted in Figure 8 determines an optimal functional base $\mathcal{F}_{opt} \subseteq \mathcal{F}$.

Lemma 9 (Optimization complexity)
Let $n = |\mathcal{A}|$ and $m = \max\{length(\mathcal{R}_M) \mid M$ a multimedia format$\}$. Then algorithm 8 has a worst case complexity of $O(2^{n-1} \cdot n \cdot m) = O(2^n)$.

The proof of this lemma is trivial, since in optimalFb all elements from the powerset of $\mathcal{A}\setminus\{A_{min}\}$ (with size 2^{n-1}) are completed to restricted functional bases. The completion to a restricted base is done in linear time w.r.t n and m, respectively.

For practical settings of n and m (see section 5) we note substantial savings in the computation of algorithm optimalFb compared to the basic algorithm from [KKK97]:

settings for n, m	basic algorithm $O((n+1)^n \cdot n^2)$	optimalFb $O(2^{n-1} \cdot n \cdot m)$
$n = 10, m = 4$	$\approx 2,6 \cdot 10^{12}$	40 960
$n = 9, m = 4$	$\approx 8,1 \cdot 10^{10}$	9 216
$n = 8, m = 4$	$\approx 2,7 \cdot 10^{9}$	4 096
$n = 7, m = 4$	102 760 448	1792

4.2 Cost functions

Tuning database systems by trading space for time must rely on an adequate cost model [RSS96]. In the sequel we discuss reasonable cost functions for the presented optimization framework. An appropriate cost function has to consider access costs *acc_cost* and storage costs *store_cost* of the multimedia database server.

Definition 10 (Cost function)
The cost function is defined as

$$cost(\mathcal{F}_{comp}) := (1-z) \cdot store\_cost(\mathcal{F}_{comp}) + z \cdot acc\_cost(\mathcal{F}_{comp}) \ .$$

By adjusting the parameter $z \in [0,1]$ the overall behavior of the database server is influenced: For small values of z the optimization will try to minimize the storage costs even if this raises the costs for data access. On the other hand, given sufficient storage space we may decide to optimize attribute access instead, choosing a z value of almost one.

The storage costs *store_cost* are modeled as being proportional (with factor c_S) to the sum of average sizes $avg\_size(A)$ of attribute $A \in \mathcal{A}_{phys}$.

Definition 11 (Storage costs)
The storage costs are defined as

$$store\_cost(\mathcal{F}_{comp}) := c_S \cdot G(\sum_{A \in \mathcal{A}_{phys}} N_{storage} \cdot avg\_size(A)) \ .$$

The parameter $N_{storage}$ determines how the average image sizes are included in the calculation. Setting $N_{storage}$ to 1 neutralize this parameter taking only average image sizes into account, whereas letting $N_{storage}$ be the number of images stored in the database accounts for total storage costs. Furthermore, the function G can be used to model the storage hardware of the system. In this case let G be a function modeling the costs for hard disks or table spaces which typically increase by leaps and bounds. On the other hand, if such a behavior is not to be modeled G can be neutralized to ignore the hardware context by setting it to the identity mapping.

The access costs acc_cost of $\mathcal{F}_{comp}$ are modeled by the sum of the access costs for the particular attributes.

Definition 12 (Access costs)
The access costs are defined as

$$acc\_cost(\mathcal{F}_{comp}) := \sum_{A \in \mathcal{A}} N_{access}(A) \cdot acc\_cost(A) \ .$$

Again we will use a parameter, $N_{access}(A)$, as the switch to alternatively modeling the accounting for average or total costs: To model average costs $N_{access}(A)$ is set to the average access rates to the particular attribute A, whereas setting $N_{access}(A)$ to the total number images accessed in format A accounts for total costs

The access costs for a single attribute differ for physically represented and computed attributes. In the physical case we assume the costs to be proportional to the average size. For a computed attribute we have to consider the access costs of the attribute it is computed from and the computation costs themselves.

Definition 13 (Attribute access costs)
Let $A, A' \in \mathcal{A}$, $fc \in \mathcal{F}_{comp}$ with $fc \equiv (A = f(A'))$. The attribute access costs for attribute A are defined as

$$acc\_cost(A) := \begin{cases} c_A \cdot avg\_size(A) & , \quad A \in \mathcal{A}_{phys} \\ \\ c_C \cdot avg\_comp(fc) + acc\_cost(A') & , \quad A \in \mathcal{A}_{comp}, A' \in \mathcal{A}_{phys} \ . \end{cases}$$

Choosing the parameter c_A typically depends on the used mass storage devices. Furthermore, the average computation costs for a conversion $avg\_comp(fc)$ are influenced by the workload on the host.

5 Optimization experiments

Now we revisit the sample applications in section 2 and study how the implemented optimization framework reduces overall costs in balancing storage volume versus access time. In addition, we present the developed database administration tool for optimized multimedia delivery and give a short insight into the implementation as well as the testbed of the application.

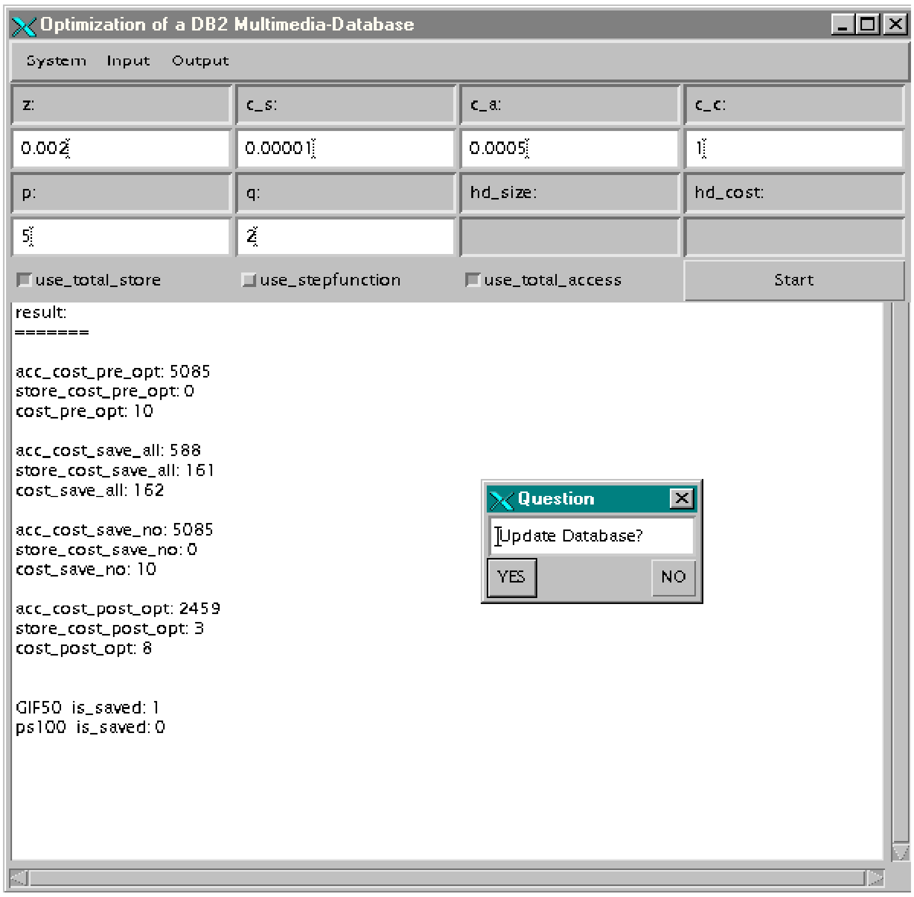


Figure 9: A database administration tool for efficient multimedia delivery.

5.1 Implementation and DBA-Tool

Intuitive, graphical database administration tools, like the Microsoft Index Tuning Wizard for SQL Server [CN98b], can ease tuning, scaling and optimization of large-scale database systems. Therefore, in addition to the core functionalities used by the optimization framework a graphical user interface has been developed. The implemented tool represents an comfortable database administration instrument which can be used automatically and semi-automatically to evaluate, simulate, test and optimize the configuration of universal database systems. Figure 9 depicts a screenshot of the working tool. In this example, a what-if analysis is performed: Given the parameters for the cost function $z = 2 \cdot 10^{-3}$, $c_S = 10^{-5}$ etc. an established database configuration is evaluated w.r.t. a given access profile. In turn, the database administrator is asked to decide whether she/he wants to add GIF-50 to the set of physically stored formats and to drop PS-100 from this set or to keep the established configuration.

In addition to the formal model from section 4 the parameters p and q are introduced: The threshold p determines which formats are taken into account, i.e. a format must be

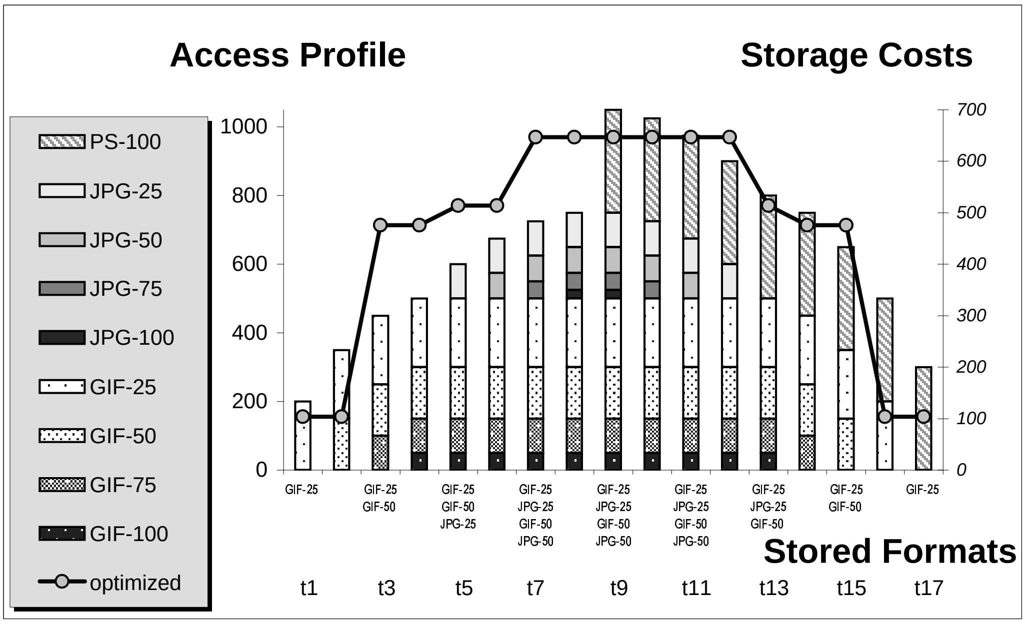


Figure 10: Storage costs at the HERON image database.

more than p-times requested to be included in the calculation, whereas q is used for the calculation of a medial profile from the last q access profiles.

Our framework has been implemented on top of the DB2 Universal Database [Hol99]. Simulation and testing was carried out on an IBM RS/6000 workstation with 768 MB RAM and approximately 30 GB of hard disk space running AIX. UDB supports Java Database Connectivity (JDBC), the Java API for client access to database management systems such as DB2. JDBC is conceptually similar to the Open Database Connectivity (ODBC) API, providing a standard method to access databases and offers a common base on which database tools can be built [Sip98]. We have used the GNU Java Development Environment (JDE) and the Java Development Kit (JDK) for the software development. Thus, ensuring the implementation to be "100% pure" Java [AG98]. At the database level the coding uses data types and user defined functions from the DB2 Relational Image Extender to store and convert images.

Since the implementation is primarily based on standard SQL, JDBC and Java, the software is platform independent and can be easily bundled and distributed as a DB2 Relational Extender as well as being ported to other universal database management systems.

5.2 Results for art historic image dissemination

At first, we will focus on the evaluation results in the context of our project. Recall the example scenario from subsection 2.1 which was motivated by the electronic delivery

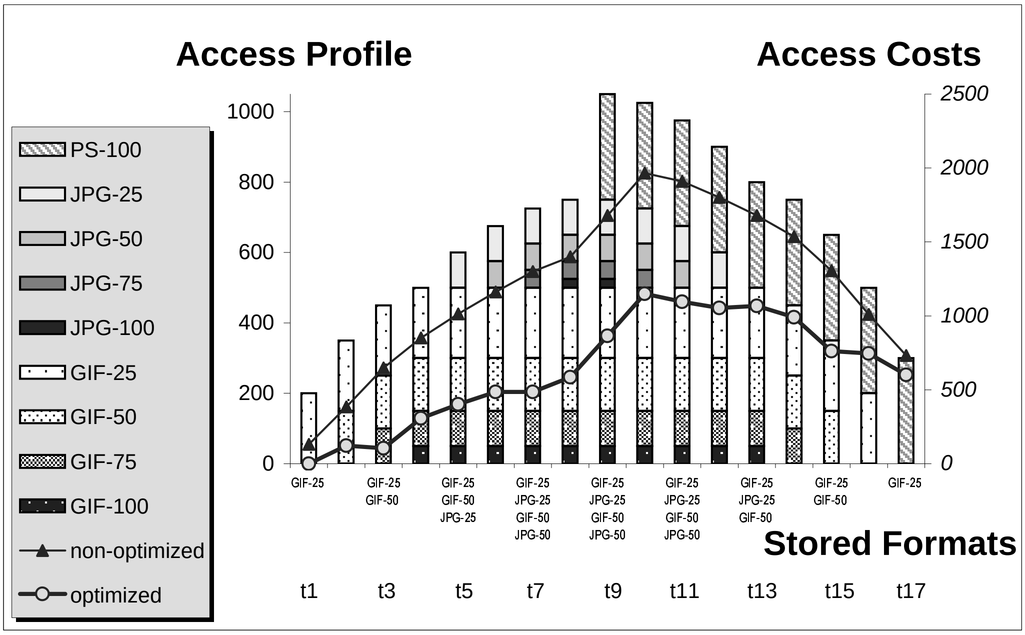


Figure 11: Access costs at the HERON image database.

of art historic image material from the HERON digital library. In the course of this scenario images are requested in the formats GIF, JPG and PS at resolutions 25, 50, 75 and 100. In addition, we let $z = 2 \cdot 10^{-3}, c_S = 10^{-5}, c_A = 5 \cdot 10^{-5}$ and $G(x) = x$. $N_{storage}$ and N_{access} account for total costs and are set to the number of images in the database and the total number of accessed images, respectively. As already mentioned, this scenario represents the varying access profiles due to typical user interaction with the heraldic image database: From searching and browsing over zooming and analysis to storage and printing. In the notions from the section above we further let A_{min} be TIFF-100, i.e. all images must be stored in the database in the TIFF format at full scale. However, since images are inserted into the heraldic collection as high quality scans, this does not represent a restriction.

The x- and the primary y-axis in the following diagrams are understood analogously to section 2. Here, in addition to the illustrations from above the x-axis is adorned with changes in the set of physically stored attributes. Again, this information is only provided explicitly for every odd point in time to keep the illustration compact. The plotted curves display changes in costs and are associated with the secondary y-axis.

Figure 10 shows how the optimization algorithm changes the physical storage layout of the database (see x-axis decoration) and influences storage costs (see curve "optimized"). Note that the constant storage cost which arises from storing all images in TIFF-100 (the minimal attribute) are ignored. As an example for a dynamic change in the set of physically stored attributes we consider the third abstract time $t3$: At this time GIF-25 and GIF-50 are added to the set of physically stored attributes (initially TIFF-100)

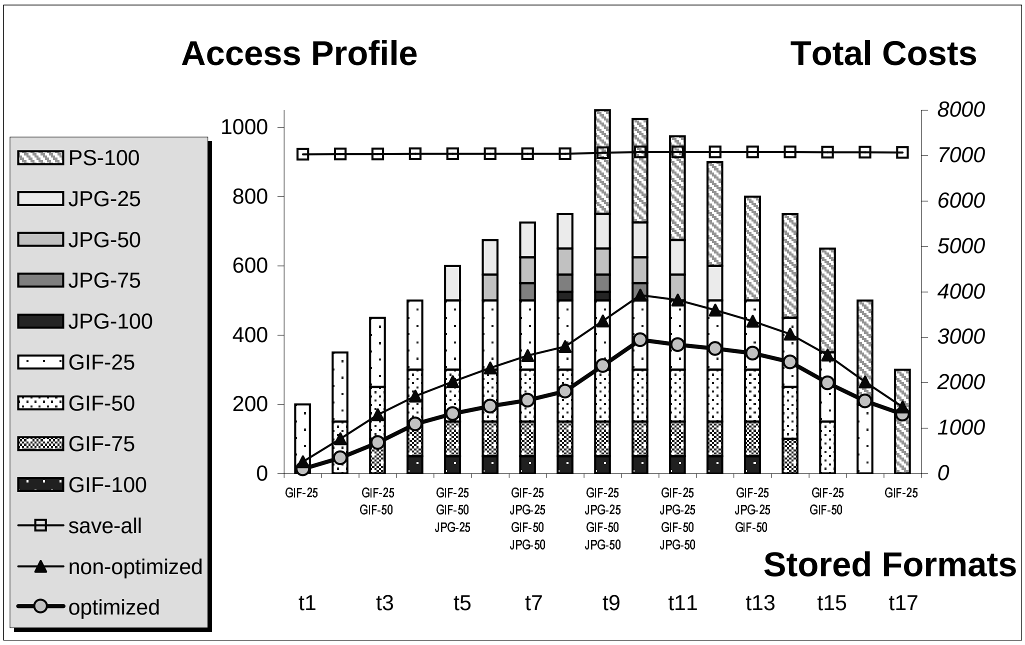


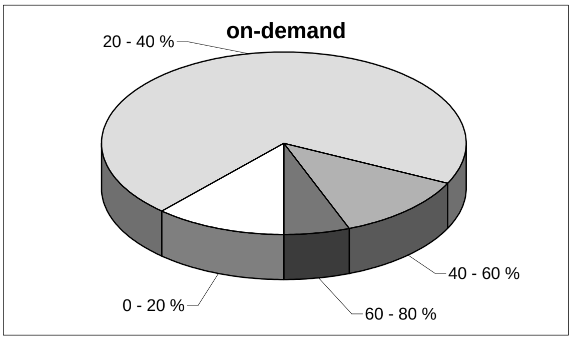
Figure 12: Total costs at the HERON image database.

raising storage costs to the abstract value of about 480.

Expenses for storage forced by the delivery optimization on the one hand are leading to access efficiency gain on the other. Figure 11 illustrates this. Whereas in figure 10 the naive configuration option for on-demand conversion is not explicitly considered — because no extra costs for storage arise — the curve "non-optimized" in figure 11 plots access costs for this setting: At time $t10$ maximal costs for deriving all the formats in the access profile from TIFF-100 are reached at an abstract value of about 2000. In contrast, the access cost (see curve "optimized") accounts for an abstract value of about 1100 which represents an efficiency gain of 45%.

Accumulated costs must be considered to evaluate overall behavior. Therefore, figure 12 displays the total overall costs for the case of on-demand image conversion (see curve "non-optimized"), the case of off-line image conversion (see curve "save-all") and the case of optimized delivery (see curve "optimized").

We complete the study for the HERON system with the charts in figure 13 which summarizes the savings of the optimized delivery: The savings from the single optimization steps are grouped into clusters, e.g. "0-20%", "20-40%" etc. The left and the right pie-chart in the illustration depict the percental size of these clusters for optimized delivery compared to on-demand and off-line conversion, respectively. For the considered sample scenario we note substantial gains.



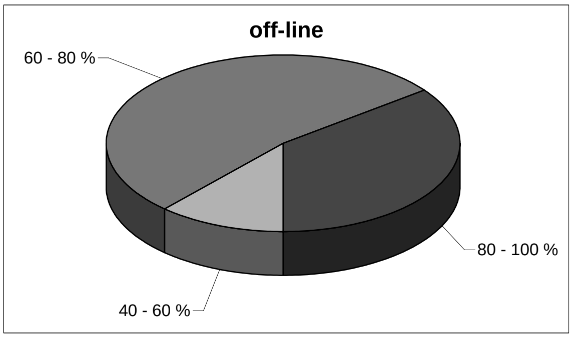


Figure 13: Optimized vs. on-demand and off-line delivery.

5.3 Results for internet/fax distribution

In a second study we compared the costs of optimized image delivery to on-demand and off-line conversion in the scenario from subsection 2.2 which was motivated by a combined World Wide Web and fax distribution of stock charts. As already introduced earlier, images in the formats GIF, IPS and Postscript are requested at resolutions 50 and 100 in the course of this example. In this scenario we assume that all the original images are stored as Microsoft Windows bitmaps, e.g. after the construction with Microsoft Excel.

Figure 14 depicts the total overall costs for the case of on-demand image conversion, the case of off-line image conversion and the case of optimized delivery. In this example we let $z = 2 \cdot 10^{-5}, c_S = 1$ and $c_A = 5 \cdot 10^{-5}$. We assume that costs for hard disks increase by leaps and bounds and model this behavior with $G(x) = \lceil x/500 \rceil \cdot 20$.

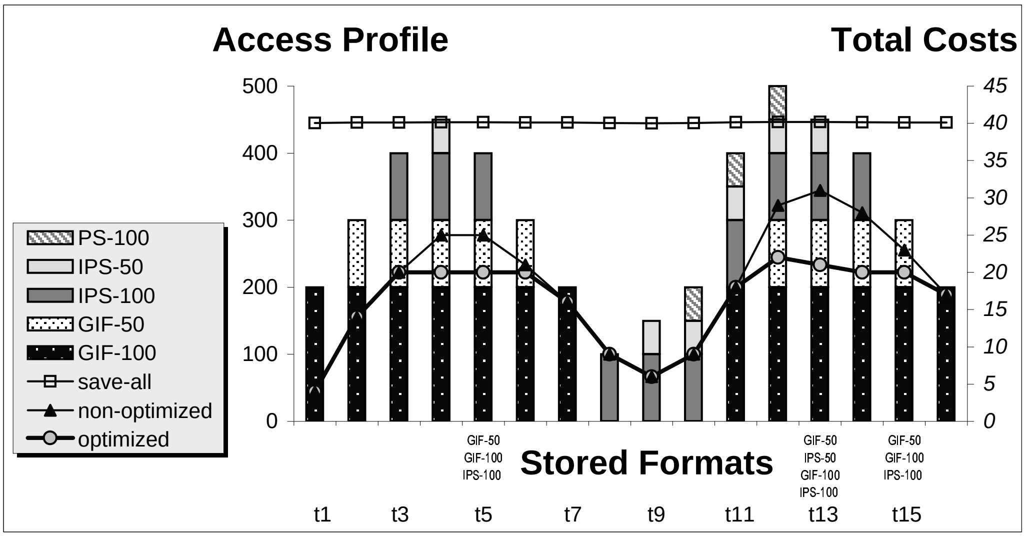


Figure 14: Total costs for the internet/fax distribution.

Again, we summarize the optimization results in charts that compare the savings of the optimization frameworks to the naive configuration options of on-demand and off-line image conversion, see figure 15.

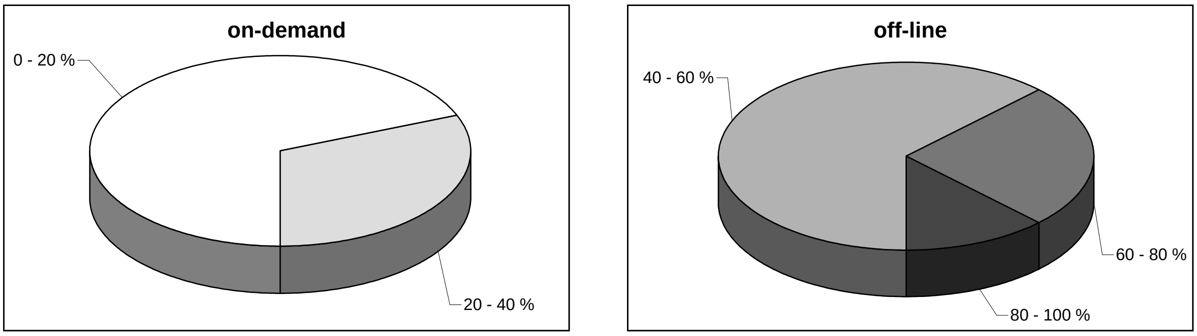


Figure 15: Optimized vs. on-demand and off-line delivery.

6 Summary and outlook

In this paper we discussed an optimization problem that is ubiquitous in multimedia digital libraries. A variety of different multimedia data formats required by the clients is accompanied by many conversion tools interrelating those formats. Thus the server can compute some formats as well as physically represent them. We formalized the conversion tools as functional constraints among formal attributes and presented an algorithm to determine an optimal division of physically represented and computed attributes. We illustrated the benefits of our approach by two scenarios arising from practical applications. First experimental results from the prototypical implementation with DB2 Universal Database are encouraging.

Multimedia digital libraries and the world wide web hold great promise for providing wider access to the world's cultural heritage [NF98]. Museum images and information in digital form can be distributed and be used for many different purposes, from enhancement of scholarship and teaching to personal exploration and enjoyment. In order to take advantage of these opportunities, however, digital-image databases must be constructed to remain relevant beyond a single short-term project, and useful to a wide audience [Kla98]. In our project the need for efficient and flexible multimedia delivery became apparent throughout the design and implementation of the HERON system, a digital library for heraldic collections.

We think that novel database technologies, like the presented work, must be employed to bring the full benefit of digital libraries to the arts and humanities. However, the established optimization framework is general enough to be applied in other domains beside this field. In virtually any application domain where large-scale universal database systems are applied for the efficient and flexible delivery of multimedia data, self-organization and self-tuning of these systems is mandatory. Research work in other areas concerned with self- and auto-administration of database systems [Spi98], e.g. automated index selection in databases [CN97, CN98a], strongly advocate this claim.

The internet has made it possible to offer a comprehensive range of services and products inexpensively over the web [Kra97, Man96]. However, other channels for improved and personalised services will be mobile networks and the mobile phone. Already there are more mobile subscribers than web clients worldwide and their rapidly increasing number will make this channel ever more important for the future success of service providers. Recent projects concerned with mobile computing have already pointed out the need for on-demand image conversion [KBA+96, FGCB98]. As an emerging universal standard the Wireless Application Protocol (WAP) [HTK22, For99, Nok99] is a technology for bringing internet content and advanced value-added services to mobile phones and other wireless devices [FGG+98]. This includes the delivery of multimedia data and images. As the platform for the new generation of "media phones" WAP is backed by major vendors including Nokia, Ericsson, Motorola, Microsoft, IBM and supports several wireless systems GSM, IS-136, CDMA, PDC etc. and because WAP and web tools are similar, it is relatively straightforward to adapt existing applications and information systems to the mobile environment. Since standards like WAP will allow developers to develop once and content providers to publish once for all users across all protocols and all carriers, the presented framework for efficient and flexible multimedia delivery can leverage providers unified access to the entire global user community.

Acknowledgements

We are grateful to Wolf-Tilo Balke, Thorsten Ehm and Thomas Birke for helpful comments and suggestions. The HERON-project is funded by the Deutsche Forschungsgemeinschaft DFG (German Research Society) within the strategic research initiative "Distributed Processing and Exchange of Digital Documents". The database software applied in the HERON system is granted within IBM's program "DB2 For Educational Purpose".

References

[AG98] Ken Arnold and James Gosling. *The Java Programming Language*. The Java Series. Addison-Wesley, Reading, MA, second edition, 1998.

[AK97] Karl Aberer and Wolfgang Klas. Multimedia and its Impact on Database Management Systems. In Peter M. G. Apers, Henk M. Blanken, and Maurice A. W. Houtsma, editors, *Multimedia Databases in Perspective*, pages 31–61. Springer, 1997.

[AT96] Prabhat K. Andleigh and K. Thakrar. *Multimedia Systems Design*. Prentice Hall, Upper Saddle River, 1996.

[AWA+96] M. Abrams, S. Williams, G. Abdulla, S. Patel, R. Ribler, and E. A. Fox. Multimedia traffic analysis using CHITRA95. In *The Third ACM International Multimedia Conference and Exhibition (MULTIMEDIA '95)*, pages 267–276, New York, November 1996. ACM Press.

[AzW99] AzWest. International Internet Fax Services. http://www.e-mail2fax.net/, 1999.

[Bro99] Paul Brown. Implementing the Spirit of SQL-99. In Delis et al. [DFG99], pages 515–518.

[CCN+99] Michael J. Carey, Donald D. Chamberlin, S. Narayanan, Bennet Vance, D. Doole, S. Rielau, R. Swagerman, and Nelson Mendonça Mattos. O-O, What's Happening to DB2? In Delis et al. [DFG99], pages 511–512.

[Cha96] D. Chamberlin. *Using the New DB2: IBM's Object-Relational Database System*. Morgan-Kaufmann, San Francisco, 1996.

[CN97] Surajit Chaudhuri and Vivek R. Narasayya. An Efficient Cost-Driven Index Selection Tool for Microsoft SQL Server. In Matthias Jarke, Michael J. Carey, Klaus R. Dittrich, Frederick H. Lochovsky, Pericles Loucopoulos, and Manfred A. Jeusfeld, editors, *VLDB'97, Proceedings of 23rd International Conference on Very Large Data Bases, August 25-29, 1997, Athens, Greece*, pages 146–155. Morgan Kaufmann, 1997.

[CN98a] Surajit Chaudhuri and Vivek R. Narasayya. AutoAdmin 'What-if' Index Analysis Utility. In Haas and Tiwary [HT98], pages 367–378.

[CN98b] Surajit Chaudhuri and Vivek R. Narasayya. Microsoft Index Tuning Wizard for SQL Server 7.0. In Haas and Tiwary [HT98], pages 553–554.

[DFG99] Alex Delis, Christos Faloutsos, and Shahram Ghandeharizadeh, editors. *SIGMOD 1999, Proceedings ACM SIGMOD International Conference on Management of Data, June 1-3, 1999, Philadephia, Pennsylvania, USA*. ACM Press, 1999.

[Exc99] Excalibur. Visual RetrievalWare. http://www.excalib.com/products/vrw/, 1999.

[FGCB98] A. Fox, S. Gribble, Y. Chawathe, and E. A. Brewer. Adapting to Network and Client Variation Using Active Proxies: Lessons and Perspectives. *IEEE Personal Communications*, August 1998. Special Issue on Adaptation.

[FGG+98] Armando Fox, Ian Goldberg, Steven D. Gribble, David C. Lee, Anthony Polito, and Eric A. Brewer. Experience With Top Gun Wingman: A Proxy-Based Graphical Web Browser for the USR PalmPilot. In *Proceedings of the IFIP International Conference on Distributed Systems Platforms and Open Distributed Processing (Middleware '98)*, Lake District, UK, September 1998.

[For99] WAP Forum. Wireless Application Protocol. http://www.wapforum.org/, 1999.

[FSN+95] Myron Flickner, Harpreet Sawhney, Wayne Niblack, Jonathan Ashley, Qian Huang, Byron Dom, Monika Gorkani, Jim Hafner, Denis Lee, Dragutin Petkovic, David Steele, and Peter Yanker. Query by image and video content: The QBIC system. *Computer*, 28(9):23–32, September 1995.

[FSZ95] B. Furth, S. W. Smoliar, and H. Zhang. *Video and Image Processing in Multimedia Systems.* Kluver Academic Publishers, Boston, 1995.

[Hol99] Stefan Holland. Java/JDBC Implementierung und Auswertung von Algorithmen zur Format-Optimierung von Multimedia-Datenbanken. Master's thesis, Universität of Augsburg, Institut für Informatik, 1999.

[HT98] Laura M. Haas and Ashutosh Tiwary, editors. *SIGMOD 1998, Proceedings ACM SIGMOD International Conference on Management of Data, June 2-4, 1998, Seattle, Washington, USA.* ACM Press, 1998.

[HTK22] Jo Herstad, Do Van Thanh, and Steinar Kristoffersen. Wireless Markup Language as a Framework for Interaction with Mobile Computing and Communication Devices. In *Proceedings of the First Workshop on Human Computer Interaction with Mobile Devices*, page 13, Glasgow, Scotland, 1998-05-21/1998-05-22.

[IBM99] IBM. DB2 Image Extender. http://www.software.ibm.com/data/db2/, 1999.

[KBA+96] Randy H. Katz, Eric A. Brewer, Elan Amir, Hari Balakrishnan, Armando Fox, Steve Gribble, Todd Hodes, Daniel Jiang, Giao Thanh Nguyen, Venkata N. Padmanabhan, and Mark Stemm. The Bay Area Research Wireless Access Network (BARWAN). In *Proceedings of the Spring COMPCON Conference*, 1996.

[KBN99] Vishu Krishnamurthy, Sandeepan Banerjee, and Anil Nori. Bringing Object-Relational Technology to Mainstream. In Delis et al. [DFG99], pages 513–514.

[KEUB+98] Werner Kießling, Katharina Erber-Urch, Wolf-Tilo Balke, Thomas Birke, and Matthias Wagner. The HERON Project — Multimedia Database Support for History and Human Sciences. In Rudolf Kruse, editor, *28. Annual Conference of the German Computer Society (GI): INFORMATIK98*, LNCS, pages 309–318, Magdeburg, Germany, September 1998. Springer Verlag Heidelberg.

[KKK97] Gerhard Köstler, Wolfgang Kowarschick, and Werner Kießling. Client-Server Optimization for Multimedia Document Exchange. In *Proceedings of the Fifth International Conference on Database Systems for Advanced Applications*, pages 135–144, Melbourne, Australia, April 1997.

[Kla98] Judith Klavans. Data Bases in Digital Libraries: Where Computer Science and Information Management Meet. In *Proceedings of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 1-3, 1998, Seattle, Washington*, pages 224–226. ACM Press, 1998.

[Kra97] Ralf Kramer. Databases on the Web: Technologies for Federation Architectures and Case Studies. In Peckham [Pec97], pages 503–506.

[KUB+99] Werner Kießling, Katharina Urch, Wolf-Tilo Balke, Thomas Birke, and Matthias Wagner. Das HERON Projekt - Ein Zwischenbericht. Technical Report 1999-04, Universität Augsburg, Institut für Informatik, August 1999.

[Man96] Udi Manber. Future Directions and Research Problems in the World Wide Web. In *Proceedings of the Fifteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 3-5, 1996, Montreal, Canada*, pages 213–215. ACM Press, 1996.

[NF98] E. J. Neuhold and R. Ferber. Scientific digital libraries in Germany: Global-Info, a federal government initiative. *Lecture Notes in Computer Science*, 1513, 1998.

[Nok99] Nokia. Nokia Wireless Data Forum. http://www.forum.nokia.com/, 1999.

[NQ99] Netscape and Qwest. Fax center. http://home.netscape.com/qwest/fax/, 1999.

[ÖRS95] Banu Özden, Rajeev Rastogi, and Avi Silberschatz. Research issues in multimedia storage servers. *ACM Computing Surveys*, 27(4):617–620, December 1995.

[ÖRS97] Banu Özden, Rajeev Rastogi, and Abraham Silberschatz. Multimedia Support for Databases. In *Proceedings of the Sixteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, May 12-14, 1997, Tucson, Arizona*, pages 1–11. ACM Press, 1997.

[Pec97] Joan Peckham, editor. *SIGMOD 1997, Proceedings ACM SIGMOD International Conference on Management of Data, May 13-15, 1997, Tucson, Arizona, USA*. ACM Press, 1997.

[RSS96] Kenneth A. Ross, Divesh Srivastava, and S. Sudarshan. Materialized View Maintenance and Integrity Constraint Checking: Trading Space for Time. In H. V. Jagadish and Inderpal Singh Mumick, editors, *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, Montreal, Quebec, Canada, June 4-6, 1996*, pages 447–458. ACM Press, 1996.

[Sha97] Dennis Shasha. Lessons from Wall Street: Case Studies in Configuration, Tuning, and Distribution. In Peckham [Pec97], pages 498–501.

[Sip98] Matthew D. Siple. *The Complete Guide to Java Database Programming with FDBC: JDBC, ODBC & SQL*. Java Masters Series. McGraw-Hill, New York, NY, USA, January 1998.

[SM96] M. Stonebraker and D. Moore. *Object-Relational DBMSs: The Next Great Wave*. Morgan Kaufmann Publ., San Francisco, 1996.

[SP97] Praveen Seshadri and Mark Paskin. PREDATOR: An OR-DBMS with Enhanced Data Types. In Peckham [Pec97], pages 568–571.

[SP99] Renate Seib and Guido Pelzer. Projekt Exilpresse Digital: Deutsche Exilzeitschriften 1933-1945. http://www.ddb.de/online/index.htm, 1999. Die Deutsche Bibliothek Frankfurt.

[Spi98] Peter M. Spiro. Ubiquitous, Self-tuning, Scalable Servers. In Haas and Tiwary [HT98], pages 512–515.

[SSU96] A. Silberschatz, M. Stonebraker, and J. Ullman. Database research: Achievements and opportunities into the 21st century. *SIGMOD Record (ACM Special Interest Group on Management of Data)*, 25(1), 1996.

[Sub98] V. S. Subrahmanian. *Principles of Multimedia Database Systems*. Morgan Kaufmann, San Francisco, 1998.

[Tec99] Brooktrout Technology. Fax Formats. http://www.brooktrout.com, 1999.

[UBK99] Katharina Urch, Wolf-Tilo Balke, and Werner Kießling. Erfahrungen in der Digitalisierung und Erschließung einer historischen Wappensammlung. Technical Report 1999-01, Universität Augsburg, Institut für Informatik, August 1999.

[Vir99] Virage. Image Read/Write (IRW) Toolkit. http://www.virage.com, 1999.