Querying Large Geo Image Databases: A Case Study[1]

PETER BAUMANN
MAXIMILIAN HÖFNER
WALTER SCHATZ
Active Knowledge GmbH, Kirchenstr. 88, D-81675 Munich, Germany
{baumann|hoefner|schatz}@active-knowledge.com

Abstract: Online services based on extremely large volumes of aerial and satellite imagery are becoming more and more important, being established both by public authorities and private providers. Technically, the requirement is to store raster arrays of multi-Terabyte sizes and to offer flexible, efficient support for extracting rectangular cutouts and performing raster operations such as zooming, overlaying, channel extraction, and derivation of value-added information like vegetation index. Usually such image servers today are implemented file based, because standard DBMSs currently do not give adequate query support for the functionality sketched.

Providing maps and aerial imagery of Bavaria is the task of the Bavarian Land Survey Authority. The state of Bavaria covers 20% of the area of Germany; its aerial photo coverage consists of 13,500 images. As the first German state, Bavaria has set up a database server for Web-based raster image access recently, based on a combination of the array database middleware RasDaMan and Oracle. The seamless aerial image, which has been compressed losslessly, occupies about 700 GB on disk. In this contribution we present design rationales, server architecture, and first experience.

1 Motivation

Increasingly raster data are complementing vectorial data in geo applications. On the one hand, storage facilities allow to keep substantial raster archives online, on the other hand acquisition of raster data is rapidly getting easier and less expensive. Additionally, timeliness of raster imagery is significantly higher as compared to vector data, due to the easier acquisition.

This advance in technology is paired by a rapidly increasing user demand for flexible, ubiquitous access to large archives of raster images and maps of various types. A broad range of raster imagery has to be considered in geo data management: grayscale and colour aerial images, multi- and hyperspectral satellite images, topographical raster maps, RADAR and laser scan, and Digital Elevation Models (DEMs). Sometimes it is desirable to keep older versions, resulting in 3-D time series data cubes[2].

Retrieval technology for large raster repositories, however, is lagging behind. Often storage is done in a one-file-per-image manner, driven by the data acquisition process rather than by user access patterns. Raster archives today commonly are implemented in a file-based manner; databases serve only for meta data search, but not for image retrieval itself. One exception to this is TerraServer [Ba-98] where maps are partitioned, and an application server logic retrieves the small images from the database to put them together. However, this is not the kind of service one would expect from a database: TerraServer offers a limited set of access functions (e.g. zoom only in steps, but not arbitrary zoom levels) whereas a query language would allow to tailor the result to the user needs. Consequently, retrieving the data needed for processing and analysis involves a lot of time and computing resources. The following example we consider as representative for an important, although still basic class of queries: "*Overlay selected channels of a multi-band satellite image with cadastral maps; colour all areas in blue that would be flooded if water rose to level L, based on the DEM; do this for the geographic area selected; zoom the result into my browser window*". Notably, each of the map items has a distinct pixel resolution.

Ideally, one would expect to formulate raster image requests as queries which are answered by the database management systems (DBMS) by returning

[1] Work supported by the European Commission under grant no. IST-1999-11009 ESTEDI, www.estedi.org, and IST-2000-28766 EuroClim, http://euroclim.nr.no.
[2] see www.eoweb.dlr.de/data-service for an example

the desired result image. Traditional DBMSs cannot accomplish this, because multidimensional arrays or *multidimensional discrete data* (MDD) obviously form a separate, fundamentally distinct information category aside from sets, object nets, and text. MDD support, consequently, requires new techniques in all aspects of database architecture, such as modelling and query support, optimisation, and storage management.

A system offering all these classical database features on raster maps is RasDaMan[3] [Ba-99, BF-99, Ri-99, Wi-99]. It is implemented as middleware running on top of a relational database system, offering a query language streamlined to semantically adequate raster retrieval.

In this paper, we present the raster database operated by the Bavarian Land Survey Authority (Bayerisches Landesvermessungsamt, BLVA), a database for interactive navigation on unbounded continuous aerial image maps and for ad-hoc generation of various map products. This database has been built with RasDaMan and Oracle.

The remainder of this paper is organized as follows. In Section 2, we highlight the BLVA raster geo service tasks and requirements. In Section 3, a brief overview of the RasDaMan model, query language, and architecture is presented. Section 4 describes the BLVA database as it has been implemented. Section 5 reviews the state of the art, and Section 6 concludes the paper.

2 Geo Image Services in Bavaria

2.1 Tasks, Products, and Workflow

In Germany surveying and mapping is organized federally. The main task of the surveying authorities within each of the 16 German state's land surveying offices is to provide accurate and up-to date geo base data. These form the basis for more specialized private geo data services.

A main concern of the federal surveying authorities is to soon provide a digital geo data infrastructure which ensures continuous service to distributed information services and further geo databases, e.g., in public-private partnerships.

The Bavarian Land Survey Authority (BLVA) is responsible for the production and provision of geo base data such as topographical maps and aerial images of Bavaria. In the past, every five years systematic photogrammetric flights have been performed to produce grayscale aerial images for the entire of Bavaria at a scale of 1:15,000; since 2002, images are taken in colour and with a frequency of three years. Bavaria, the largest state of the Federal Republic of Germany, has about 20% of the Germany area.

These aerial images comprise the source of a popular product of the land surveying authorities, the geo-referenced (*Digital*) *Ortho Photo*, DOP.

DOPs are cut in a way that each one covers at least the area of a cadastral map (which has an approximate size of 2.4 by 2.4 km) yielding an average image size of 6,500 by 6,500 pixels with 43 MB file size. Since the introduction of the digital production process for high-resolution ortho photos with a pixel resolution of 40 cm in 1995, about 13,500 DOPs have been produced.

In current practice, the resulting images are not aligned with horizontal/vertical image boundaries, hence a black rim of null values, i.e. grayscale value 0, is added by the imaging software (see Fig. 1) to achieve an axis-parallel bounding box.

As an aside, for topographical maps the situation is still more complicated in that there is a large variety of products, pertaining to different combinations of the map layers, different colourings, and different editions depending on the usage (e.g., standard and military edition).

2.2 Core User Requirements

DOPs form the starting point for various products, ranging from pure ortho photos to aerial image maps, i.e., overlays of DOPs and the digital cadastral map.

In the past, geo data users could order ortho photos only by indicating the identifier according to the cadastral nomenclature. This has several shortcomings:

- Users have to know the nomenclature which frequently is meaningless to them and makes ordering error prone; actually, experience in several other German states shows that false orders happen not just occasionally.
- Images always cover a whole grid square. Sometimes users want only a smaller part but nevertheless have to order (and pay) the whole image, sometimes they need a larger area and get a set of files comprising the desired map.

[3] See www.rasdaman.com

Fig. 1: Ortho photo with rim of null values

- Resolution is fixed, users can not indicate an arbitrary scale factor and resolution which is feasible for many practical applications (e.g., to get a quick overview without moving hundreds of Megabytes of data).

Hence, it is desirable to have just one seamless map on which users can navigate online and interactively (based on visual inspection or on features such as cities or ZIP codes) and where they can select a zoom factor arbitrarily. As an additional benefit, such an approach requires considerably less human effort to satisfy customer requests. This is urgently needed in view of the anticipated wave of user requests – for example, in the state of Hassia where geo data access now is for free for public authorities, 7,000 regions – each involving a set of DOPs – have been requested in the first three months of 2002 alone.

In the past, these images have been stored in a CD juke box. To achieve online interactive access, it was decided to switch to a database system. Critical requirements were [GH-02]:

- Access to any area at any zoom level within a second;
- Handling of all the raster map types occurring, ranging from binary topographical maps up to RGB colour ortho photos and floating point DEMs, all with individual resolution;
- Flexibility to perform not just zoom and pan, but also overlaying, colouring, high-resolution data export, etc.; it was felt that a query language offers substantial benefits wrt. flexibility and openness for future functionality requirements;
- Image storage in a database to obtain all the classical database benefits like high-level interfaces, transaction support, query optimisation;
- A clean multi-tier component architecture with open, standard conformant interfaces.

2.3 OGC Geo Server Standards

For raster services, two Open GIS Consortium (OGC) specifications are particularly relevant, the *Web Map Server* (WMS) *Implementation Specification* and the *Web Coverage Server* (WCS) *Implementation Specification*[4]. Several vendors already support WMS; WCS, which focuses on multidimensional raster data, is still in progress; implementations exist, to the best of the authors' knowledge, only in the academic field for now[5].

OGC WMS was decided to form the central client interface for the BLVA server. For the server components, a combination of the multidimensional raster server RasDaMan and relational Oracle was chosen after a thorough selection process.

3 The RasDaMan Array Server

RasDaMan is middleware specialized on large n-dimensional arrays. Its algebra-based query language [Ba-99, BF-99], RasQL, extends standard SQL92 with declarative raster operators. Server-based query evaluation relies on algebraic optimisation [Ri-99] and a specialised array storage manager [Wi-00].

3.1 Conceptual Model

The conceptual model of RasDaMan centers around the notion of an n-D array (in the programming language sense) which can be of any dimension, spatial extent, and array cell type. Following the relational database paradigm, RasDaMan also supports sets of arrays. Hence, a RasDaMan database can be conceived as a set of tables where each table contains a single array-valued attribute, augmented with an OID system attribute.

Arrays can be built upon any valid C/C++ type, be it atomic or composed ("struct"), based on ODMG's [Ca-97] type definition language. Arrays are defined through a template `marray<b,d>` which is instantiated with the array base type `b` and the array extent (*spatial domain*) `d`, specified by the

[4] see www.opengis.org

[5] as for an example, a WCS service is accessible under http://eoweb.dlr.de:8080/wcs-ows1/

lower and upper bound for each dimension. Thus, an unbounded colour DOP can be defined by

```
typedef marray
<  struct{ char red, green, blue; }, [ *:*, *:* ]
> ColourOrthoImg;
```

3.2 Array Retrieval

Like SQL, a RasQL query returns a set of items (in this case MDD objects).

Trimming produces rectangular cut-outs, specified through the corner coordinates.

Example 1: "*A cut-out between (1000,1000) and (2000,2000) from all ortho images*":

```
SELECT OrthoColl[1000:2000,1000:2000]
FROM   OrthoColl
```

For each operation available on the cell (i.e., pixel) type, a corresponding *induced operation* is provided which simultaneously applies the base operation to all MDD cells. Both unary (e.g., record access) and binary operations (e.g., masking and overlaying) can be induced.

Example 2: "*Topographical map bit layer 3 oveölaid with the (grayscale) ortho image*":

```
SELECT Ortho overlay bit( Map, 3 ) * 255c
FROM   Map, Ortho
```

In general, MDD expressions can be used in the `SELECT` part of a query and, if the outermost expression result type is Boolean, also in the `WHERE` part. See [Ba-99] for further query constructs such as condensers (the MDD counterpart to aggregates).

The expressiveness of RasQL enables a wide range of signal processing, imaging, and statistical operations up to, e.g., the Fourier Transform. The expressive power has been limited to non-recursive operations, thereby guaranteeing termination of any well-formed query.

3.3 Physical Array Storage

RasDaMan storage is based on the partitioning of an MDD object into *tiles*, i.e., sub-arrays [FB-99]. Aside from regular grids, any user or system generated partitioning is possible (Fig. 2). A geo index (currently: R-tree) is employed to quickly determine the tiles affected by a query. Optionally tiles are compressed using one of various techniques, using lossless or lossy (wavelet) algorithms; moreover, query results can be compressed for transfer to the client. Both tiling strategy and compression comprise database tuning parameters.

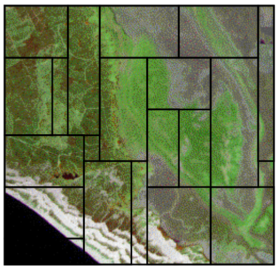

Fig. 2: arbitrary 2-D tiling

Tiles and index are stored as BLOBs in a relational database which also holds the data dictionnary needed by RasDaMan's dynamic type system.

Implementing an index on top of the relational DBMS of course incurs some performance penalty. However, despite the large body of discussion on various indexing techniques, our observation is that indexing influence on performance actually is secondary. Taking far less than one percent of overall query execution time, we consider the index not worth deep experimenting at the moment.

Data units with different growth behaviour are separated into different tablespaces so that the administrator can control growth (and eventually migration) individually. For example, RasDaMan data dictionary, index data, and raster blob data reside in different tablespaces.

Adaptors are available for Oracle, IBM DB2, and IBM Informix. A coupling with object-oriented O2 has been done earlier, showing the wide range of DBMSs with which RasDaMan can interoperate.

3.4 Query Evaluation

Queries are parsed, optimised, and executed in the RasDaMan server. The parser receives the query string and generates the operation tree. A number of optimisations is applied to a query tree prior to its execution [Ri-99]. Of the 150 heuristic rewriting rules, 110 are actually optimising while the other 40 serve to transform the query into canonical form. All rules are based on the algebra.

Execution of queries is parallelised. Right now, RasDaMan offers inter-query parallelism: A dispatcher schedules requests into a pool of server processes on a per-transaction basis. Current research work involves intra-query parallelism where the query tree transparently is distributed across available CPUs or computers in the network [Ha-02].

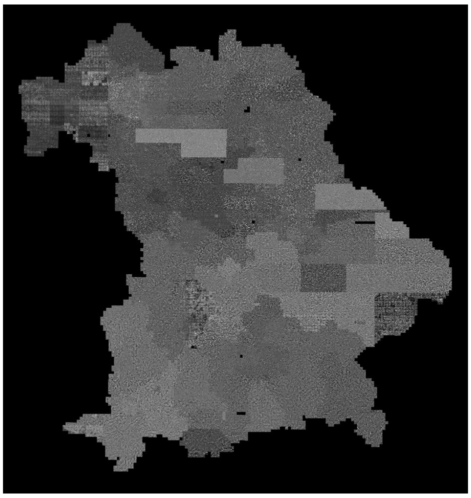

Fig. 3: overview image of Bavaria DOP map (zoom of the DOP map to browser window size)

First performance results are promising, showing speed-up / #CPU ratios of 95.5%.

4 The BLVA Database

The BLVA database contains a seamless grayscale DOP map (Fig. 3) covering the whole area of Bavaria with approximately 70,000 square kilometres at a ground resolution of 40 cm. At the time of this writing, the seamless statewide ortho photo resides in the raster server. All data contents has been verified down to pixel level.

Obviously radiometric correction is not complete in Fig. 3. Currently, up to thousand DOPs can be radiometrically balanced simultaneously. In general, radiometric correction is a photogrammetric image processing task in the preprocessing phase and, as such, beyond the scope of the raster server itself.

4.1 Design Issues and Image Import

The DOPs have been merged into the single seamless image of size 950,000 x 1,000,000 pixels. The base data set consists of 13,500 images; they have been stored, however, grouped by county, with duplicates to ensure that every county is covered completely. Including these duplicates there is a total of 16,500 images. While in principle the base set of 13,500 images is sufficient as data source, it would have been too tedious a task to sort out images not needed, hence simply all 16,500 were imported.

A couple of complications had to be overcome:

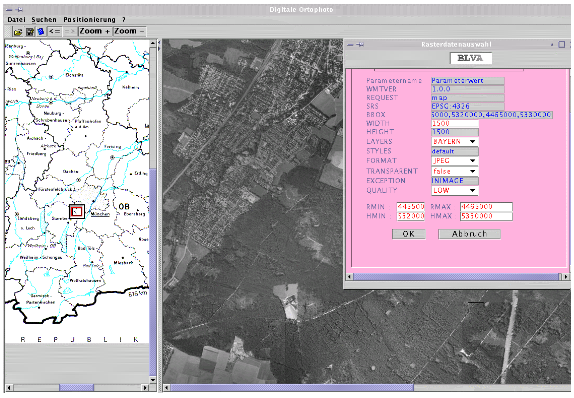

Fig. 5: OGC viewer accessing the map database

- During the flight campaign, aerial images tentatively are taken in an overlapping way. This data redundancy has to be handled.
  Decision was made to neglect redundancy due to overlaps; during import a simple "last pixel wins" strategy was adopted. The slight loss of information is more than outbalanced by the easier handling for the users who can navigate now over a single, continuous, coherent area.
- The null values in the rim of each file have to be removed during import. RasDaMan offers two ways to handle this: either by passing a polygon bounding box for each image which delineates meaningful values, or by defining a pixel value, in this case: 0, as transparent value.
  As the second alternative incurs small errors in situations where the actual intensity value within the DOP by coincidence is zero, the first strategy was adopted.

Database set-up is as follows. In a first run, an empty overall map is generated. This allows, among others, to set global parameters such as compression and tiling schemes used. Notably, compression has to be lossless – as a public authority, BLVA must be able to deliver original, unmanipulated data. This excludes techniques such as ECW [EC-02] which rely on lossy wavelet-based compression. From the various compression schemes available in RasDaMan, `zlib` was chosen. This is a public domain compression algorithm which is used, e.g., in GNU zip. The tiling scheme used was a simple one which poses an upper limit of about 250 kB on tile size.

To speed up zoom access, an image pyramid [BA-83] is maintained by the RasDaMan server. Altogether, nine layers are maintained.

4.2 Overall Architecture

The RasDaMan v5.0 middleware with the underlying relational database Oracle 8.1.7 is hosted on a DEC AlphaServer 4000 with 2 processors of 533 MHz clock, 2 GB main memory, and operating system Tru64 v4.0f. This server hosts not only the raster database, but several more services like the ILIAS data catalog.

The raster part of the Oracle database occupies in excess of 700 GB. After various tests of different physical database designs to get experience with issues such as backups of huge volumes on DLT tapes, an Oracle datafile size of 20 GB was chosen.

RasDaMan is operated with five parallel server processes for satisfying an expected clientele of about 5,000 users.

The WMS interface is implemented in Java using the Tomcat servlet engine. The use of an enterprise beans container in the application tier is planned. Due to the platform independence of the OGC WMS interface it is possible to implement a variety of different front-ends at the client-tier. In Fig. 6, an HTML/JavaScript based client displays the result of an WMS-conformant http request.

4.3 Performance Results

Benchmarking was done on a Pentium III based Linux notebook (650 MHz) with 256 MB main memory. It ran an Oracle 8.1.7 server together with the RasDaMan server and the query client, plus the X server, which actually brought the machine to swapping in some situations. Data sets used were a 36,000x36,000 ortho image and a 48,000x47,000 topographical map (1:10,000) containing nine thematic layers. The JPEG images generated were between 10 kB and 25 kB in size. Ortho image zoom factors were chosen randomly, no difference was observed in response time. Each query was run five time, times measured were averaged.

Query 1: "*From the grayscale ortho image, a cut-out scaled to window size, in JPEG format":*

```
SELECT jpeg( scale(Ortho[x0:x1,y0:y1],f) )
FROM   Ortho
```

Query 2: "*From the topographical map, the streets layer in red with background black; cut-out area (x0,y0) to (x1,y1), scaled to window size*":

```
SELECT jpeg(
          scale(bit(Tk10[x0:x1,y0:y1],streets),
                f) * {0c,0c,255c} )
FROM   Tk10
```

Query 3: "*The Digital Elevation Model (DEM), coloured in three steps: 0.0 to 5.0 in red, 5.0 to 100.0 in green, above 100.0 in blue":*

```
SELECT (Dem <   5.0) * {255c,0c,0c}
     + (Dem >   5.0 AND Dem < 100.0)
                    * {0c,255c,0c}
     + (Dem > 100.0) * {0c,0c,255c}
FROM   Dem
```

The table below summarises measurements.

Table 1: Benchmark of the example queries.

Query type	Average total elapsed time [sec]
Query 1	0.194
Query 2	0.360
Query 3	0.887

In summary, the most important function – zomoing into the image (see Fig. 6) – can be accomplished typically in less than one second on less than common hardware.

5 State of the Art

Image processing packages are widely used in remote sensing. They offer advanced image processing algorithms, but lack data management capabilities such as sub-image extraction from a several hundred GB continuous image.

For fast zoom and pan on mosaicked image file sets, many products are available, e.g., [Li-02]. Access is done through low-level API libraries instead of high-level, model-based query support with internal optimisation and without flexible image extraction functionality, such as hyperspectral chan-

Fig. 6: Zoom into the Bavaria image database

nel extraction, overlaying, and *ad hoc* thematic colouring. Most important, file-based solutions *per se* do not scale very well.

Relational DBMSs, designed to scale well indeed, traditionally store multidimensional arrays as unstructured BLOBs ("binary large objects") introduced by Lorie as "long fields" [Lo-82]. This technique cannot give any support for operations beyond line-by-line access, something clearly not feasible for large archives.

Object-relational database systems (ORDBMSs) allow to add new data types, including access operations, to the server [SD-98]; examples for such a data type is Oracle Intermedia. Arrays, however, are not a data *type*, but a data type *constructor* ("template"), parametrized with cell type and dimension – see Section 3.1. Such templates are not supported by ORDBMSs, hence a separate data type has to be defined for 2-D grayscale ortho images, 2-D hyperspectral MODIS images, 4-D climate models, etc. Furthermore, server internal components are not prepared for the kind of operations occurring in MDD applications, therefore important optimization techniques like tile-based pipelining and parallelization of array query trees are difficult to implement. Finally, implementing the whole RasDaMan system as an Oracle cartridge, DB2 extender, etc. would mean a complete loss of portability. Hence, we feel that ORDBMSs currently are not an option in array data management, at least for deployment cycles measured in industrial timescales.

Some research focuses on specific raster database aspects, such as data storage [SS-94] and data models [LM-96, MS-99]. While there are interesting results, these have not (yet) made their way into operational systems, and they are not supported by an appropriate overall architecture. RasDaMan, conversely, is a complete multidimensional array DBMS product. It is being used in 12 countries over Europe, Russia, and USA for, 2-D ortho image maps, 3-D satellite image time series, 4-D climate simulations, and others.

6 Summary and Outlook

Management of geo imagery in databases is an emerging area receiving growing attention. Technology is at a point where databases as a substitute to homegrown file-based image management are considered, albeit sometimes still with doubts among users and content providers. The BLVA case clearly demonstrates feasibility and scalability of large image databases. Actually, we believe that the 700 GB seamless ortho image of Bavaria is the largest image that has been accessed by query language up to now.

Meantime a mass export has taken place where the whole database contents was exported via the OGC WMS interface, showing the scalability of the OGC WMS interface.

Database size will approximately triple by switching from grayscale to colour images, so that the seamless colour image is expected to occupy 2 TB compressed then. Further it is planned to bring other raster types into the database, such as DEM and topographic raster maps.

Although RasDaMan is implemented as an additional tier on top of a relational DBMS, performance turned out to be clearly superior to file-based approaches; in fact, a broad range of practically relevant queries on near-Terabyte databases performs in the sub-second area even on PC servers.

Beyond performance, however, we see further advantages in the enhanced functionality provided by the query language approach, and in general by bringing standard database benefits such as multiuser synchronisation, transaction support, and concise, explicit schema modelling to the area of raster data management. Finally, storing geo images in the database together with meta and vector data not only eases administration, but also enhances data consistency considerably.

BLVA is going to establish a digital geo data infrastructure where the RasDaMan database will form a main component at the back-tier for the online retrieval of DOPs.

On the architectural side, next steps include introduction of load balancing techniques to accommodate heavily parallel access and failover hardware/software redundancy towards 7x24 service availability.

Current research work of FORWISS on RasDaMan encompasses parallel query evaluation, advanced optimisation for complex statistical queries, and tape silo support.

Acknowledgements

We want to express our gratitude to the phantastic RasDaMan team. Many thanks also to Bruno Roder from BLVA who always is a most valuable and supportive counterpart in all discussions.

References

[BA-83] P.J. Burt, E.H. Adelson: The Laplacian pyramid as a compact code. IEEE Transaction on Communications, vol. COM-31, April 1983, pp. 532-540.

[Ba-98] T. Barclay et al.: The Microsoft TerraServer. Technical Report MSR-TR-98-17, Microsoft Research, June 1998.

[Ba-99] P. Baumann: A Database Array Algebra for Spatio-Temporal Data and Beyond. Proc. Next Generation IT and Systems NGITS, Zikhron Yaakov, Israel, 1999, pp. 76 - 93.

[Ca-97] R.G.G. Cattell: The Object Database Standard: ODMG 2.0, Morgan Kaufmann Publishers, California, 1997.

[EC-02] www.gisdevelopment.net/technology/ic/techip0003.htm .

[FB-99] P. Furtado, P. Baumann: Storage of Multidimensional Arrays Based on Arbitrary Tiling. Proc. ICDE, Sydney, Australia, 1999, pp. 480-489.

[GH-02] C. Glock, W. Hettmer: Web-based management of raster data at the Bavarian Land Survey Authority (in German). In: Schilcher, Matthäus (ed.): 7. Münchner Fortbildungsseminar Geoinformationssysteme, Dept. of Geographic Information Systems, Technische Universität München, Munich, 2002.

[Ha-02] K. Hahn et al: Parallel Query Support for Multidimensional Data: Inter-object Parallelism. Proc. DEXA, Aix-en-Provence, France, 2002.

[LM-96] L. Libkin, R. Machlin, and L. Wong: A Query Language for Multidimensional Arrays: Design, Implementation, and Optimization Techniques. Proc. ACM SIGMOD, Montreal, Canada, 1996, pp. 228 - 239.

[Lo-82] R.A. Lorie: Issues in Databases for Design Transactions. in: Encarnaçao, J.; Krause, F.L. (eds.): File Structures and Databases for CAD, North-Holland Publishing, 1982.

[MS-99] A.P. Marathe, K. Salem: Query Processing Techniques for Arrays. Proc. ACM SIGMOD '99, Philadelphia, USA, 1999, pp. 323-334.

[Re-02] Bernd Reiner et al: Hierarchical Storage Support and Management for Large-Scale Multidimensional Array Database Management Systems. Proc. DEXA, Aix en Provence, France, 2002.

[Ri-99] R. Ritsch: Optimization and Evaluation of Array Queries in Database Management Systems. PhD Thesis, Technische Universität München, 1999.

[SM-94] S. Sarawagi, M. Stonebraker: Efficient Organization of Large Multidimensional Arrays. Proc. ICDE'94, Houston, USA, 1994, pp. 328-336.

[SM-98] M. Stonebraker, D. Moore, P. Brown: Object-Relational DBMSs: Tracking the Next Great Wave (2nd edition). Morgan Kaufmann, September 1998.

[Wi-00] N. Widmann: Efficient Operation Execution on Multidimensional Array Data. PhD Thesis, Technische Universität München, 2000.