Large-Scale, Standards-Based Earth Observation Imagery and Web Mapping Services

Peter Baumann

rasdaman GmbH
Bunsenstr. 4
D-81675 Munich
Germany
E-Mail baumann@rasdaman.com

Abstract

Earth observation (EO) and simulation data share some core characteristics: they resemble raster data of some spatio-temporal dimensionality; the complete objects are extremely large, well into Tera- and Petabyte volumes; data generation and retrieval follow very different access patterns. EO time series additionally share that acquisition/generation happens in time slices.

The central standardization body for geo service interfaces is the Open GIS Consortium (OGC). Earlier OGC has issued the Web Map Service (WMS) Interface Specification which addresses 2-D (raster and vector) maps. This year, the Web Coverage Service (WCS) Specification has been added with specific focus on 2-D and 3-D rasters ("coverages").

In this paper we present operational applications offering WMS/WCS services: a 2-D ortho photo maintained by the Bavarian Mapping Agency and a 3-D satellite time series deployed by the German Aerospace Association. All are based on the rasdaman array middleware which extends relational DBMSs with storage and retrieval capabilities for extremely large multidimensional arrays.

1. Motivation

Online services based on extremely large volumes of aerial and satellite imagery are increasingly being established both by public authorities and by 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.

In practice, today file-based image storage completely prevails, such as servers built around compression techniques [mrsid, ecw]. As a consequence, today such data usually reside on disks and tape archives, and specialized applications alleviate search across these thousands of files. Access is through libraries instead of flexible query languages, and generally speaking data storage is driven by ingestion needs rather than by end user access patterns.

File-based implementations very much convey the same tendency that has been observed, e.g., in CAD databases more than a decade ago: arguing that the "non-standard" requirements could not be fulfilled by traditional technology, many vendors stepwise re-implement database functionality such as indexing, concurrency control, and storage managers, thereby often reinventing the wheel. Having said this, relevant knowledge about large-scale MDD management has been collected in the disciplines handling them since long, such as High-Performance Computing (HPC; see, e.g., the PANDA project [panda]). For instance, tertiary storage / tape robot support and parallel data access are areas where database researchers can benefit from HPC knowledge. It is a relevant question, therefore, how far traditional database technology can be used advantageously in this "non-standard" domain, and what it can learn from other disciplines.

Research on array data management in DBMSs usually focuses on particular system components, such as storage of multidimensional data [Sar94, panda], or query language [Mar97]. Further interesting work is reported in [Lib96].

Proceedings of the 29th VLDB Conference,
Berlin, Germany, 2003

Rasdaman, on the other hand, is a fully implemented and operational generic array DBMSs [Bau94, Bau99, ras]. The system is deployed in 12 nations (including US and Russia), with most installations being run by public mapping agencies. This makes it a valuable vehicle to study all aspects of multidimensional data management in a holistic way, thereby augmenting focused research done elsewhere.

On the side of database vendors, Oracle is working on a GeoImage extension for 2-D imagery only [oracle]; however, while its announcements describe a subset of rasdaman's functionality, the extension is not yet available.

2. Rasdaman Overview

The overall goal of rasdaman is to provide classic database services in a domain-independent way on MDD structures. Based on a formal algebraic framework [Bau99], rasdaman offers a query language which extends SQL-92 with declarative MDD operators, and an ODMG conformant programming interface. The query language has been designed highly optimizable, with a streamlined underlying storage manager. The latter combines MDD tiling with spatial indexing and compression whereby an administration interface allows to change default strategies for application-driven database tuning [Fur99]. Array sets resulting from queries are delivered in the client's main memory format or in some a data exchange format as selected by the application.

2.1 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"). They 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 lower and upper bound for each dimension. An unbounded colour image map can be defined by

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

2.2 Query Processing

The query language expressiveness enables a wide range of signal processing, imaging, and statistical operations up to, e.g., the Fourier Transform [Bun93]. The expressive power has been limited to non-recursive operations, thereby guaranteeing termination of any well-formed query.

Queries are parsed, optimised, and executed in the rasdaman server. The parser receives the query string and generates the operation tree. A number of algebraic optimisations is applied to a query tree prior to its execution. Of the 150 heuristic rewriting rules, 110 are actually optimising while the other 40 serve to transform the query into canonical form. Examples for such rules are "pull out disjunctions while aggregating cell values of an MDD using logical or" and "push down geometric operations to the expressions' leaves". The latter rule ensures that just the minimal amount of data necessary to compute the result of the query branch is read from the storage manager. Further, the query tree is searched for common MDD subexpressions. Beyond conventional subexpression matching, the spatial domains are checked for overlapping regions which have to be loaded and computed only once. The choice of physical algorithms, finally, is driven by indexing and tiling information. For instance, if an operation does not prescribe any particular tile inspection sequence, iteration order will be chosen corresponding to storage order. The tile-based execution strategy pipelines the execution process on tile level whenever possible to reduce memory requirements for intermediate results.

Associativity and commutativity of most cell operations opens up ample space for parallelization. Up to 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 [Hah02]. First performance results are promising, showing speed-up / #CPU ratios of 95.5%.

2.3 Physical Array Storage

Rasdaman storage is based on the partitioning of an MDD object into *tiles*, i.e., sub-arrays [Fur99]. Aside from regular grids, any user or system generated partitioning is possible (Fig. 1). A geo index (currently: R+-tree) is employed to quickly determine the tiles affected by a query. Optionally tiles are compressed using one of

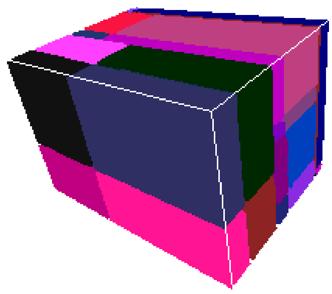

Fig. 1: Visualization of a 3-D tiling pattern using rView.

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. 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. For extremely large amounts of data, tertiary storage management with spatial clustering is available [Rei02].

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. In principle, the adaptor layer of rasdaman could be changed to make rasdaman a relational cartridge/extender etc. using object-relational technology. For commercial reasons, however, resources hitherto have been put into different projects.

3. Demonstration

We will demonstrate rasdaman using different tools. The query frontend, rView, allows to interactively submit n-D queries and display result sets containing 1-D to 3-D data. The specialized geographic Web interface, rasgeo, relying

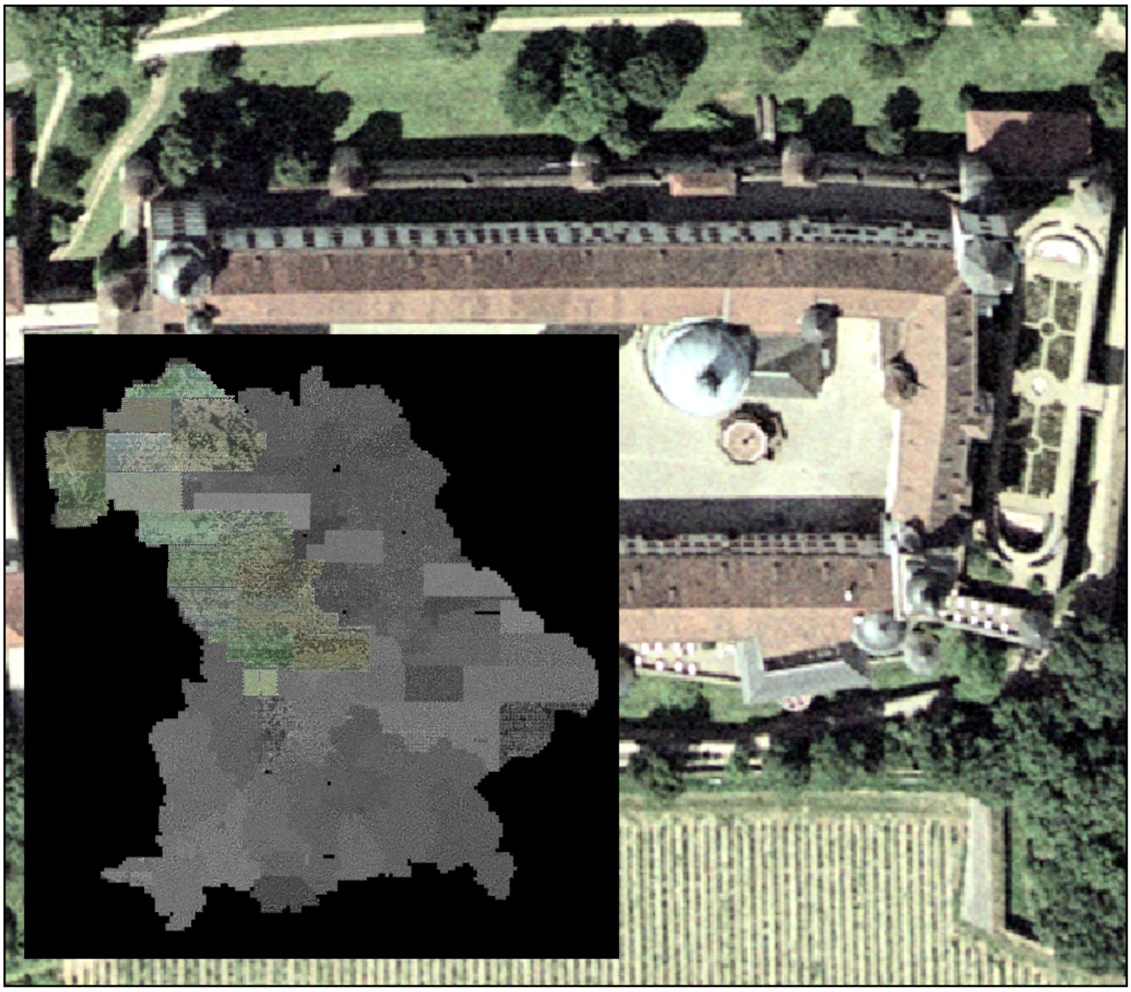

Fig. 2: Overview of (current) color and grayscale aerial image of Bavaria, and hi-res zoom. (courtesy Bavarian Mapping Agency)

on the Open GIS Consortium (OGC) Web Map Server (WMS) standard for geographic services.

Assuming that Internet access can be provided, access to large-scale data sets will be demonstrated:

- The aerial images of the Bavarian Mapping Agency (Fig. 2). This database consists of two images, color and grayscale, of the whole of Bavaria which makes up 20% of Germany. At submission time, in excess of 45,000 images have been imported. Bulk export of 600 GB through the OGC WMS interface has proven feasibility of this standard also for high-volume requests.
- DLR (German Aerospace Center [dlr]) operates a rasdaman server with satellite imagery of Europe and the Mediterranean. About 10,000 images have been put into a 3-D data cubes (Fig. 3). This service is one of the very few implementations of OGC's new Web Coverage Service (WCS) standard which is focused on 2-D/3-D raster maps.

Additional on-site demonstration will rely on 2-D maps (aerial images, thematic maps, and Digital Elevation Models), 3-D image time series, and 4-D climate data brought on-site.

In particular we present and discuss mapping of WMS requests to raster queries, and their implications for design and optimization. We thereby want to contribute to proliferating knowledge about OGC's geo service standards and to stimulate discussion across disciplines.

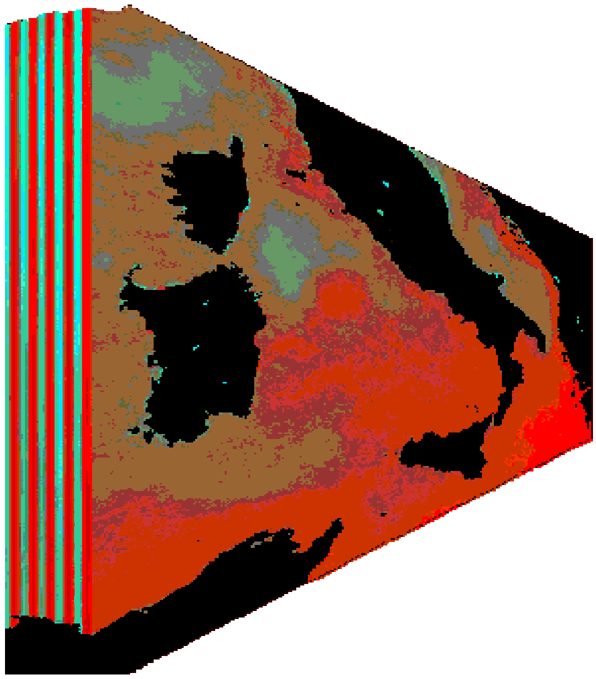

Fig. 3: perspective view on a cutout from 3-D satellite image cube covering Europe and the Mediterranean. (courtesy DLR)

References

[Bau99] 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

[Bau98] P. Baumann, A. Dehmel, P. Furtado, R. Ritsch, N. Widmann: *The Multidimensional Database System Rasdaman.* Proc. ACM SIGMOD'98, Seattle, USA 1998, pp. 575 - 577

[Bun93] P. Buneman: *The Discrete Fourier Transform as a Database Query.* Technical Report MS-CIS-93-37/L&C 60, University of Pennsylvania, 1993

[dlr] www.dfd.dlr.de

[ecw] http://www.ermapper.com

[Fur99] P. Furtado, P. Baumann: *Storage of Multidimensional Arrays Based on Arbitrary Tiling.* Proc. ICDE'99, Sidney - Australia 1999

[Hah02] K. Hahn et al : *Parallel Query Support for Multidimensional Data: Inter-object Parallelism.* Proc. DEXA, Aix-en-Provence, France, 2002

[Lib96] L. Libkin, R. Machlin, L., Wong : *A Query Language for Multidimensional Arrays: Design, Implementation, and Optimization Techniques,* Proc. ACM SIGMOD, Montreal, Canada, 1996, pp. 228 – 239

[Mar97] A. P. Marathe, K. Salem: *A Language for Manipulating Arrays.* Proc. of VLDB'97 Conference, Athens, Greece, 1997

[mrsid] http://www.algovision-luratech.com

[oracle] http://www.oracle.com

[panda] http://drl.cs.uiuc.edu/panda/

[ras] http://www.rasdaman.com/

[Rei02] B. Reiner et al: *Hierarchical Storage Support and Management for Large-Scale Multidimensional Array Database Management Systems.* Proc. DEXA, Aix en Provence, France, 2002.

[Sar94] S. Sarawagi, M. Stonebraker: Efficient Organization of Large Multidimensional Arrays. *Tenth Int. Conf on Data Engineering*, pp. 328-336, Houston, Feb. 1994.