As could be expected, demands of the user community and developments in computer hardware and software have affected database technology. Although, traditionally, database practitioners worked with well-structured or semistructured databases, the development of new and reliable sensing techniques added impetus to research of unstructured databases such as those containing images.

These trends focused interest on unconventional database applications, despite their very large data volumes. For example, a 150-bed hospital generates 2 Gbytes of image data each day, and the US National Aeronautics and Space Administration’s planned Earth Observing System program could generate several Gbytes of data every day. The archiving, retrieval, and management of such large databases constitute a complex and difficult task. Recent trends in database technology suggest that database computers can best undertake the efficient management of large databases. That is, specialized hardware and software configurations aimed primarily at handling large databases and answering complex queries become the best solution.

In general, most of the data modeling effort focuses on three widely used, structured database models: relational, hierarchy, and network. Other, recently proposed object-oriented data models may be suitable for modeling databases that include structured and unstructured data. Since the theoretical underpinnings of the relational model have been well defined, the relational model became the focus of most of the commercial and research efforts in database machines. Some additional effort in the development of hardware solutions for semistructured databases like text retrieval also surfaced. The articles in this issue reflect these database machine trends. Four of them discuss issues related to relational databases, and another delves into text retrieval.

The primary motivation of the database machine approach is to increase the performance of updating operations and the query processing of databases. One approach taken to achieve this objective employs parallel architectures. The designer must choose the appropriate approach for data distribution and storage, interprocessor communications, and computational capability of each node. We can divide the parallel architectures studied in the context of database processing into three groups: shared memory, shared disk, and “shared nothing.”

The current trend in the design of database computers is toward the increasingly popular shared-nothing message-passing architectures. Designers usually prefer these architectures over shared-memory and shared-disk architectures because they permit a high degree of scalability. The MDBS database computer described by Hsiao in this issue belongs to this category.

Database engines represent another example of shared-nothing database systems. The high-performance and fault-tolerant database engines generally support a server/client architecture. A number of new database engine products have already hit the market, and more are currently under development. Teradata’s DBC/1012 uses 80386 processors (up to 1,024). Because processors can be added as needed to increase both CPU and input/output capabilities, this database engine can handle very large databases. White Cross offers a database engine based on the Inmac transputer. The hardware and software structure in this database engine permits a maximum of 100,000 MIPS of processing power.

The articles in this special issue on database computers fit into two main classes. The first class centers on hardware accelerators for database computers, and the second class presents two different database computers. The proposed hard-
ware accelerators concentrate on the efficient implementation of specific database tasks. The articles on database computers deal with the overall architecture, performance, and data distribution issues; they also discuss specific hardware accelerators incorporated in the systems.

The first three articles examine the design and evaluation of special-purpose accelerators for database computers. Lee et al. present two VLSI accelerators for formatted and unformatted databases. The database tasks performed by the accelerators include operations such as associative search, aggregation, and string search.

Faudemay and Mhiri present an associative accelerator whose speedup ratio is independent of the database size. The accelerator implements relational database operations and sorting and aggregate functions.

Our article describes the design and simulation of a bit-serial accelerator for statistical aggregation operations. A number of bit-serial processing elements connected according to the odd-even network topology make up the accelerator. The proposed unit achieves a high degree of pipelining and parallelism.

The remaining two articles describe two database computers and emphasize both architectural and performance considerations. Hsiao presents an experimental database computer with a variable number of database processors known as the Multiback-end Database Supercomputer (MDBS). Each microprocessor-based processor has a private database store, consisting of a small Winchester drive for paging and metadata and a large drive for the database.

Finally, Inoue et al. describe a relational database processor with specialized hardware for searching and sorting known as Rinda. Rinda is composed of two hardware accelerators, the CSP (Content Search Processor) and ROP (Relational Operation Accelerating Processor). The CSP searches rows of data on a disk storage and transfers the selected rows to the main memory. The ROP subsequently sorts the row transfers to the main memory.

We thank the reviewers who helped us referee the submitted papers and the IEEE Micro editorial staff. This special issue would not have appeared without their assistance.

### References


### Reader Interest Survey

Indicate your interest in this article by circling the appropriate number on the Reader Service Card.

- High 150
- Medium 151
- Low 152

Address questions concerning this issue to M. Abdelguerfi, Department of Computer Science, University of New Orleans, New Orleans, LA 70148; mahdi@noacl.cs.uno.edu.