

DataBlitz: A High Performance Main-Memory Storage Manager

J. Baulier P. Bohannon S. Gogate S. Joshi C. Gupta A. Khivesera
H. Korth P. Mcilroy J. Miller P. P. S. Narayan M. Nemeth R. Rastogi
A. Silberschatz S. Sudarshan

Bell Laboratories
700 Mountain Avenue
Murray Hill, NJ 07974

DataBlitz¹ is a main-memory storage management toolkit that supports the development of *high-performance* and *fault-resilient* applications requiring concurrent access to shared data. By combining transactions, data organization, and fault-resilience with direct access to data, DataBlitz bridges the gap between the speed of data structures in shared memory and the safety and convenience of a traditional database system.

The architecture of Datablitz differs significantly from that of commercially available database management systems as it is optimized for high performance in the case when sufficient main-memory is available to hold the entire database. Our benchmark results indicate that DataBlitz can deliver as high as 50 times the performance of commercial disk-based systems.

Most commercial database systems have a client-server architecture, assume data is primarily disk-resident, and have a buffer manager to cache the disk-resident data. Consequently, the performance of applications on these systems suffer since each access incurs network latency and buffer manager overhead. Also, most commercial systems expose to applications only high-level interfaces (e.g., SQL) which provide additional functionality at the cost of increased processing overhead.

In contrast, Datablitz eliminates inter-process communication and the buffer manager by mapping the entire database directly into the address space of the application's process. By locking the entire database

in main-memory, disk accesses for data location can be completely eliminated. Thus, applications have direct access to data resulting in high speed. Finally, Datablitz provides a multi-layered API that includes low-level APIs to its storage manager and index components, as well as a high-level relational API. As a result, simple applications involving updates, lookups and scans can be built to deliver high performance using the low-level interfaces without incurring the overhead associated with the high-level components.

Applications like real-time billing, and switching and call routing in telecommunications are suited to DataBlitz's features. The characteristic these applications have in common is the need for fast random access to valuable data that is main-memory resident (at current DRAM prices, machines with 128MB-2GB of main-memory are fairly common).

Some of the key features provided by DataBlitz include:

- Direct access to data mapped into the application's address space.
- Full transaction semantics (ACID properties), if desired.
- Protection from process, system and media failures.
- Fuzzy checkpoints that minimally interfere with transaction processing.
- Codeword and memory protection to prevent data corruption due to stray application pointers.
- Hot standby support for higher availability.
- Fine-grained item-level locking.
- Fast, user-level semaphores.
- Highly concurrent hash and ttree index structures.
- Multi-layered Application Programming Interface that includes a high-level relational API and low-level collection and storage management APIs.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the VLDB copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Very Large Data Base Endowment. To copy otherwise, or to republish, requires a fee and/or special permission from the Endowment.

**Proceedings of the 24th VLDB Conference
New York, USA, 1998**

¹DataBlitz is the product version of the *Dali* research project.