

Berkeley DB and Solid Data

October 2002

A white paper by Sleepycat and Solid Data Systems

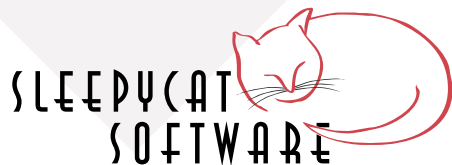


Table of Contents

Introduction to Berkeley DB	1
Performance Tuning in Database Applications	2
Introduction to Solid Data	3
Integration with Solid Data	4
Performance Results	5
Conclusions	6

Introduction to Berkeley DB

Berkeley DB, the most widely deployed embedded database system in the world, is used in systems ranging from large-scale email and directory server systems to high performance network switch and routers that handle core data transmission on the Internet. Across this range of applications, speed, reliability and availability are critical. Developers choose Berkeley DB because it provides enterprise-grade transaction management and recovery services in a compact package suitable for embedded deployment.

Berkeley DB provides superior data management for these systems because it is designed for use inside applications, rather than as a stand alone database management system. Berkeley DB links directly into the application's address space, rather than operating as a separate server. As a result, the application can store and fetch data directly, rather than sending a message to a remote server every time a record is required. This drastic reduction in the run-time cost of database access speeds up applications dramatically. Berkeley DB's easy-to-use programmatic interfaces make it simple for software developers to save and search for data in its native format, without translation to some relational database format. Because there is no query language parser, planner, optimizer, and executor that operate at runtime, and because no data translation is required, access is simpler and faster than for conventional relational database systems.

Demanding applications require not just speed, but also guaranteed reliability so that data is never lost, even if the application or system fails. Berkeley DB uses the same techniques for data integrity and failure recovery as the most popular enterprise relational systems on the market, and is able to provide the same reliability guarantees. These services, including two-phase locking and write-ahead logging, allow Berkeley DB to survive system failures without losing data. Recovery processing at system restart time ensures that all committed database updates are restored and available, and any interrupted work is rolled back so that it does not appear in the database.

Applications that require high availability, so that they continue to run even if one or more parts of the system fail altogether, can use Berkeley DB's replication service. Replication keeps multiple independent systems synchronized. In the event that one or more of the systems fails, the others are able to take over for them. The application can continue to run without interruption.

Like most database systems, Berkeley DB uses magnetic disk for storage of both actual database records and the log that it uses to guarantee that data cannot be lost. During normal processing, records are read from and written to the disk, and every time a transaction is completed, Berkeley DB writes log information to disk. Berkeley DB uses the file system interfaces provided by modern operating systems to access the disk.

Performance Tuning in Database Applications

Like most enterprise-grade database systems, Berkeley DB performs extremely well when properly tuned for the application it supports. Performance tuning for database systems requires an understanding of the data stored, common search and update workloads, and the characteristics of the hardware platform on which the system runs.

Database systems generally store information on a magnetic disk or other persistent storage device. As the application uses data, records are copied from disk into the computer's main memory. This transfer is a common source of performance problems in real-world applications. While main memory accesses can be very fast, reading information from magnetic disk involves moving the actuator arm that carries the disk head, waiting for the records of interest to rotate underneath the read head, and then copying the data from through the disk controller and into the operating system's memory. The entire operation can take tens of milliseconds using common magnetic disk systems, and that overhead translates directly into slower application performance.

In addition, enterprise-grade database engines like Berkeley DB use logging techniques to guarantee that no data is ever lost, even if the system loses power or suffers a hardware or system software failure. Whenever the application makes a change to a record in the database, Berkeley DB writes a log entry to a special location. That log entry allows the database engine to redo or back out the change later, in the event of a failure. Log processing is generally tuned to minimize its performance impact, but writing log entries to disk during normal processing can be another source of performance problems in demanding applications.

One of the best ways to improve the performance of an application that uses the database is to reduce the cost of I/O operations. Reducing the number of disk accesses and making every disk access faster makes the database engine, and therefore the application, run much faster.

Introduction to Solid Data

Solid Data manufactures a line of high-performance, highly-available solid state storage systems widely used in the telecom and financial services markets. The Solid Data product family provides applications with a familiar file system interface. Unlike conventional magnetic disk, however, Solid Data's solid state storage system uses normal memory to store file data. Every solid state storage system includes a battery-backed power supply and intelligent monitoring system that watches for power failure.

If a Solid Data device loses power, the battery-backed power supply allows it to continue running without shutting down. If the external power failure lasts long enough, the internal battery may run down. In that case, the monitoring systems write all the information in magnetic memory to an emergency backup disk inside the chassis. On restart, the memory image can be reloaded from the disk.

During normal processing, Solid Data's solid state storage products use only memory for data storage. No information needs to be written to disk. This provides I/O intensive applications, including those that use database systems, with a dramatic performance improvement - typically as much as 1,000%. The most common sources of latency - I/O costs associated with data storage on magnetic disk and log management - disappear.

Integration with Solid Data

Solid Data and Sleepycat Software have integrated Berkeley DB with the Solid Data solid state storage products to provide an embedded data management solution to the most demanding applications.

Integration of the two products was straightforward. Berkeley DB uses the operating system's file system interfaces to store and fetch data, and for log management. Solid Data provides standard file system interfaces to its solid state storage product family. The result is that an application can use Berkeley DB with a Solid Data solid state storage system by simply locating an application's Berkeley DB directory in the Solid Data file system.

Sleepycat ran exhaustive tests against the combined system. The Solid Data solid state storage system performed flawlessly with Berkeley DB.

Performance Results

In order to characterize the performance of Berkeley DB using the Solid Data solid state system for storage, Sleepycat Software ran two different benchmarks reflecting two different application workloads.

Both benchmarks ran on a single-processor Sun Enterprise 450 using a SCSI interface to the storage device. The disk tests ran against a conventional magnetic disk. The solid state tests ran against a Solid Data model 800 system. In all the tests, both the Berkeley DB log region and the database files were located on the same storage device.

The first benchmark measured the performance of write-intensive applications using Berkeley DB for record storage. The benchmark used multiple threads of control, each of which fetched a ten-byte record from the database, updated a part of the record, and wrote the changed record back to the database. Each read/modify/write operation ran as a single transaction. The benchmark did no other processing of the records. As a result, it reflects the performance that a write-intensive application should expect.

Using fifty concurrent threads, each continuously updating single records in the database, the disk-based version of Berkeley DB was able to deliver 849 transactions per second. When the Solid Data system was used instead, performance jumped to 3,266 transactions per second - nearly 300% performance improvement. The twin benefits of an embedded database engine and a high-performance solid state storage system as a backing store delivered outstanding throughput using a relatively inexpensive computer system.

The second benchmark mixed read and write operations in a single workload. Eighty percent of the database accesses were single-record reads. Twenty percent were single-record writes. This time, each record was 200 bytes long. Using magnetic disk and 100 concurrent threads of control, Berkeley DB delivered exactly 100 transactions per second under this workload. After switching to the Solid Data system, Berkeley DB delivered 1,139 transactions per second - over 1,000% performance improvement.

Conclusions

Berkeley DB is a high-performance embedded database system capable of delivering very high transaction throughput. When used with a conventional magnetic disk storage device on inexpensive hardware, benchmark applications achieved between 100 and 850 transactions per second throughput, depending on workload. When using a Solid Data solid state storage system in place of magnetic disk, throughput jumped dramatically, to between 1,139 and 3,266 transactions per second.

The combination of Berkeley DB's embedded architecture and the high-performance, low-latency Solid Data solid state storage system delivers unmatched transaction rates for applications with the most demanding performance requirements.

About Sleepycat

Sleepycat Software, Inc. (<http://www.sleepycat.com>), a private company with offices in California and Massachusetts, was founded in 1996 to commercially develop, support, and distribute Berkeley DB. Berkeley DB is the embedded database engine behind many of the largest ISPs and wired and wireless networks, as well as e-commerce sites around the world. Our solutions are used in mission-critical applications by some of the world's largest computing enterprises, including Alcatel, Amazon.com, Ask Jeeves, CMG, Cisco Systems, Motorola, RSA Security, and Sun Microsystems. Berkeley DB is open source and runs on all major operating systems, including embedded Linux, Linux, MacOS X, QNX, UNIX, VxWorks, and Windows. In January of 2002, Sleepycat won the prestigious Crossroads A-List Award for Berkeley DB.

For more information, see www.sleepycat.com.

Corporate Headquarters:
118 Tower Road
Lincoln, MA 01773, USA
www.sleepycat.com

Emily Salus
Telephone +1-510-923-9312
Email emily@sleepycat.com

About Solid Data

Solid Data Systems is the leading global provider of solid-state storage systems that multiply the throughput of business-critical, transaction-intensive applications. Solid Data's products are based on solid-state memory technology and are designed to provide near-instant access to data storage. These compact, rack-mount systems combine the speed of main memory with the persistence (non-volatility) of external disk storage - enabling enterprises to multiply the transaction performance of general-purpose systems, reduce complexity, and scale their infrastructures cost effectively. Customers include global telecommunications equipment integrators and financial service providers.

Visit Solid Data on the web at www.soliddata.com.

Corporate Headquarters:
3542 Bassett Street
Santa Clara, CA 95054
USA
Tel: +1.800.287.0373 +1.408.845.5700
Fax: +1.408.727.5496

Gary Taggart
Telephone +1-408-845-5743
E mail gtaggart@soliddata.com



The Solid Data logo is a registered trademark in the United States and Japan. All other brands, or products are the trademarks or registered trademarks of their respective owners. Solid Data disclaims any proprietary interest in