

IBM solidDB Animation

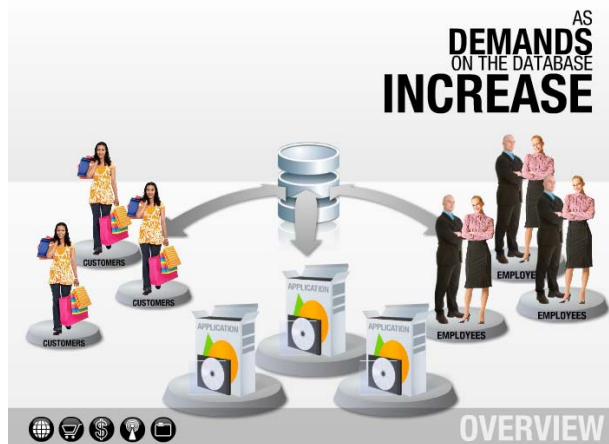
Overview

Information runs today's businesses. Customers, employees, and real time applications rely on the contents of the enterprise database.

As demands on the database continue to increase, administrators, developers, and architects search for ways to maintain performance levels and response times.

There is a better way. IBM solidDB.

INFORMATION
RUNS TODAY'S BUSINESSES



IBM solidDB



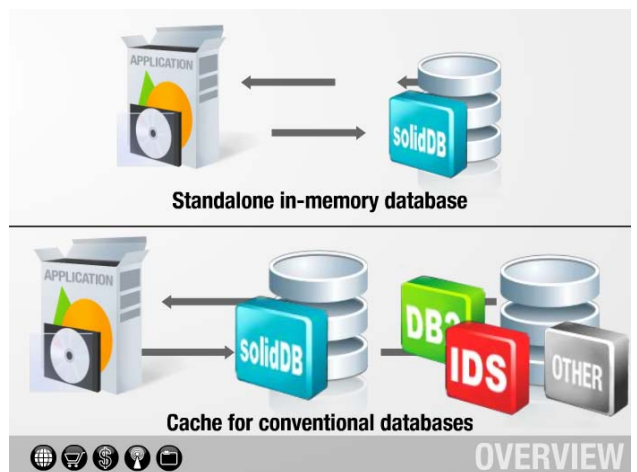
As relational in-memory database technology, solidDB performs up to ten times faster than conventional databases.



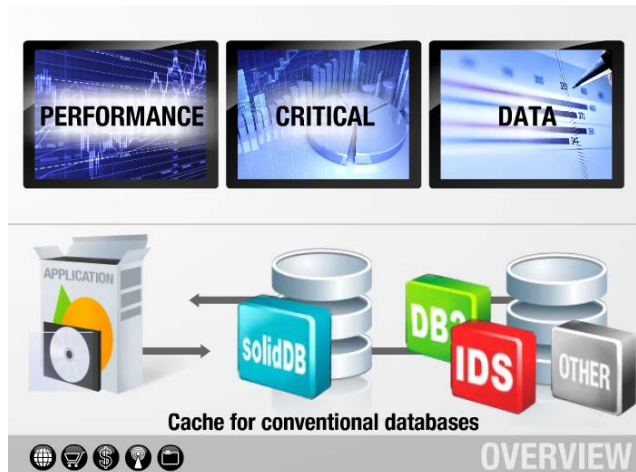
By keeping data in main memory, rather than on disk, solidDB meets the performance demands of real-time applications, allowing massive throughput: tens of thousands of transactions per second with response times measured in microseconds.



solidDB can be deployed as a stand alone in-memory database, or as a cache for conventional databases such as DB2, IDS, and other industry-leading databases.



When used as a database cache, solidDB greatly accelerates access to performance-critical data – a valuable benefit for applications servicing high numbers of concurrent users, or unpredictable or peak workloads.



Let's look at a few examples.



Use Case 1 – eCommerce

Visit any of today's e-commerce websites and you'll see the result of several backend applications working in concert.



Some are managing inventory levels, while others generate dynamic content such as personalized recommendations, shopping carts, and wish lists.



Successful sites are handling a staggering number of concurrent users...and that's on a normal day.



When a long-awaited book is finally released, the site must handle an even greater number of users. The various applications have merely fractions of a second to perform their task. Some customers are eagerly trying to complete the sale while others are adding the item to the wish list. And even more are penning a review or rating.



The site must continue to provide personalized recommendations for related movies, toys, or games, or face missing out on valuable cross-sell revenue. Timing out during a sale transaction could drive a potential customer to buy elsewhere.



With the global demand for the book, there are no off-peak times for the system; bedtime in New York means it's lunch-time in China.



IBM solidDB Cache is able to reduce latency in accessing data, providing predictable response times and high volume throughput. By caching the relational data used to provide dynamically assembled pages, the site can avoid potential bottlenecks coming from the database.



All together, web retailers can provide customers with a more rewarding experience *and* better monetize web visitor traffic.



Use Case 2 – Financial Services

Financial markets can turn on a dime, and traders look to technology for help monitoring the multitude of transactions happening each second.



An investment bank has an application to provide real time alerts about market events. This application needs a way to analyze a mountain of data with every tick of the clock. Market feeds and the bank's current trade positions are fed into the IBM WebSphere Enterprise Service Bus, which routes all the information to the analysis application.



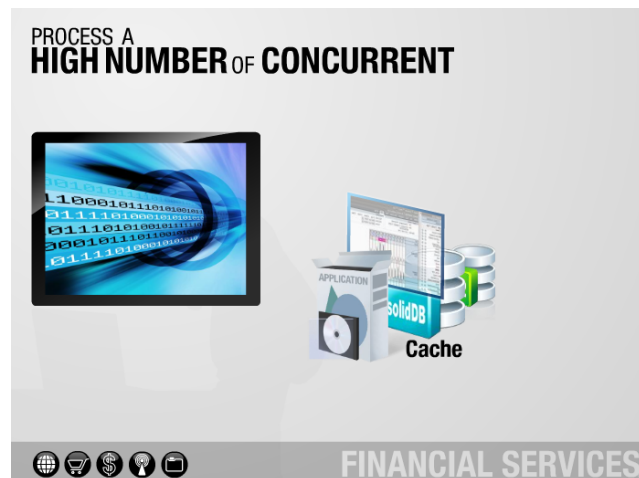
To stay on top of developing trends, the last 30 minutes of market transactions are stored in the IBM solidDB Cache.



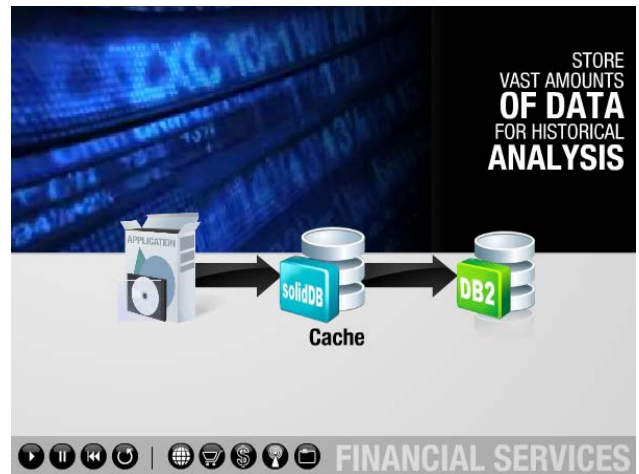
Each transaction is evaluated against thousands of rules, and traders can be notified instantly when they should take action on a profit opportunity or reduce risk exposure. Storing the transaction data in the solidDB Cache allows the application to process a high number of concurrent inserts, updates, and deletes.



Where a traditional database could perform the job in more time, solidDB Cache makes it possible to simultaneously execute a large number of concurrent queries. The response times must be very short, and it all needs to happen without falling behind the normal stream of operations.



solidDB Cache then propagates the transactions to a disk-based database, where the vast amount of data will be stored for historical analysis.



In situations where real time can mean the difference between gains and losses, solidDB delivers the extreme speed required to power the world's most performance-critical applications.



Use Case 3 – Communication

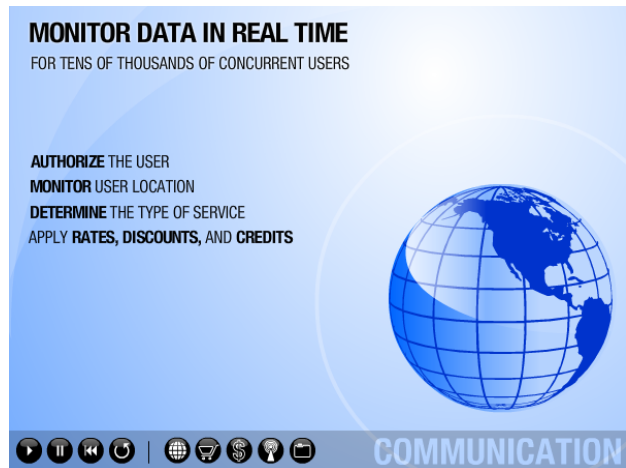
Cell phones are all around us, with a variety of options for users of all ages and needs. The monthly calling plans have also increased in variety: family plans, international plans, text message plans, and pay-as-you-go or prepaid plans.



To offer these services, carriers are tasked with several challenges. First, they must authorize the user before service can be accessed, then monitor where the user is to apply the correct usage or roaming rates; determine the type of service, such as text messaging, web surfing, or content downloading; and apply appropriate rates, discounts, and credits for billing purposes.



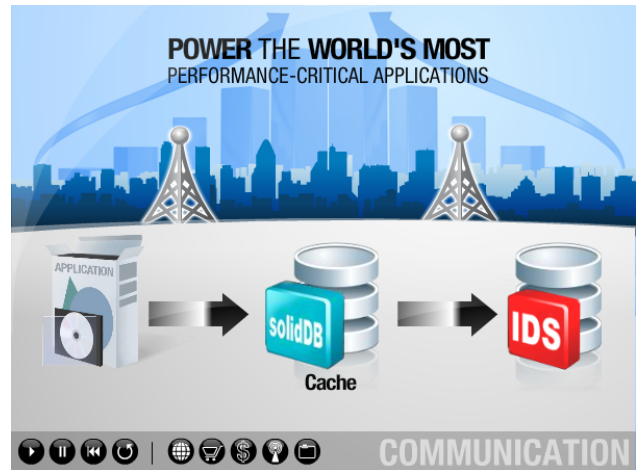
That's only some of the data to monitor, in real time, for tens of thousands of concurrent users.



Having an effective, real time infrastructure is critical for allowing communications operators to deliver competitive services, as well as to recognize revenues.

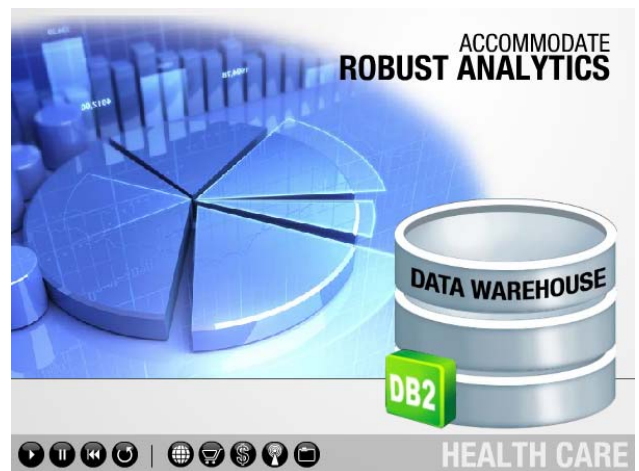


solidDB delivers the extreme speed required to power the world's most performance-critical applications.



Use Case 4 – Health Care

Industry trends in health care require health benefit providers to consolidate and integrate member information and claim data. Many plan sponsors are expanding their data warehouses to uniquely identify individuals and members, and integrate external and partner data to accommodate robust analytics.



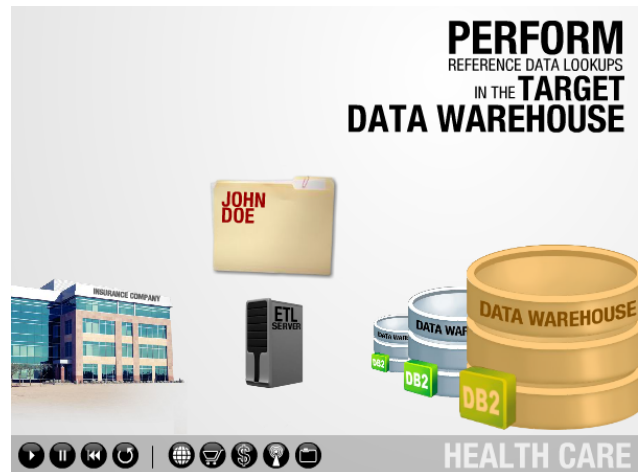
Not only are data volumes on the rise, but performing more frequent updates to data warehouses is becoming an increasingly key requirement for timely decision support.



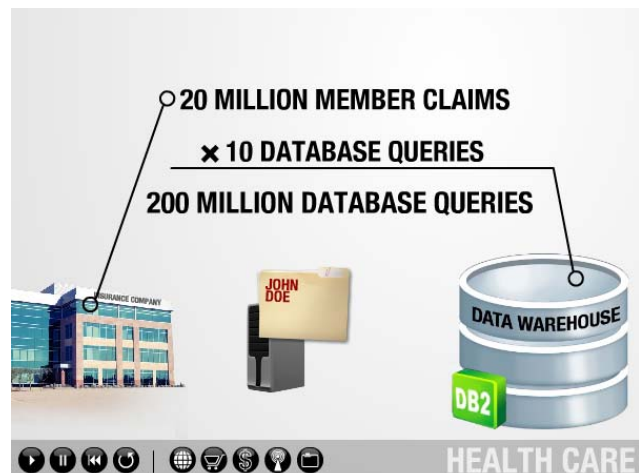
At the end of a specified time period, whether it's one day or one week, the member and claim data must be loaded to the data warehouse through a process known as ETL: extraction, transformation, and loading.



During the transformation stage, member and claim data has to be de-duplicated and cleansed. To properly transform millions of records, it is necessary to perform reference data lookups in the target data warehouse.



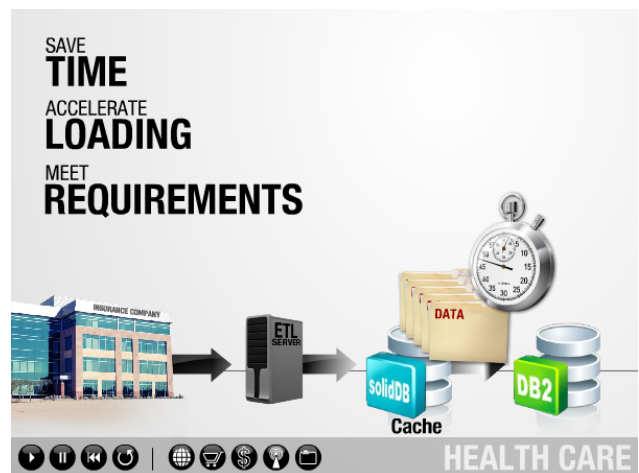
For example, if the claims data that must be added to the data warehouse has 20 million records, and each record needs ten reference lookups in the target data warehouse, the system must perform two hundred million lookups.



Organizations can minimize transformation time with solidDB Cache. By preloading the needed lookup data from the data warehouse into solidDB Cache, the time needed to perform each lookup operation can be reduced from milliseconds to microseconds – a time savings that adds up over hundreds of millions of transactions.



As a result, loading of incremental claim data will be accelerated by an order of magnitude, enabling data warehouse updates to meet operational time window requirements.



The extreme speed of solidDB accelerates the ETL process and enables users to load data warehouses rapidly, allowing faster decision making with the most recent data.

