IBM Data Management Solutions
September 2001

IBM Informix Dynamic Server: High availability for mission-critical applications

Abstract
IBM offers solutions with a variety of high-availability features for critical applications that require access to all types of data, 24 hours a day, 7 days a week. This level of continuous database availability is needed during maintenance and administration, as well as in the event of network failure. Given this, the high-availability features of IBM Informix® Dynamic Server™ make it the ideal environment for all types of data processing. These features include online utilities for backup and recovery, reorganization of tables, enterprise replication, cluster and data failover, software mirroring and more.

Table of Contents

Highlights

IT systems downtime can delay critical business decisions and impact the corporate bottom line.

Around-the-clock availability is imperative for both online transaction processing systems and decision support applications.

Market overview

Around-the-clock database availability is more critical than ever before in today's data-driven business environment. Mission-critical database applications, such as supply chain, order processing and distribution management, require nearly 100 percent availability. Industries such as telecommunications, international banking and travel depend on IT systems to support virtually zero downtime. After all, downtime can delay information delivery for critical business decisions, which can result in lost sales opportunities and tarnished reputations.

Technology overview

High-availability processing solutions are used for all types of information processing, including online transaction processing (OLTP) and decision support systems (DSS) processing.

Online transaction processing

Database availability is of critical importance for OLTP systems that receive continuous transaction data. A business cannot function normally if the database is unavailable because of a hardware or software failure, which can have devastating consequences. Most airline reservation systems, for example, are available 24 hours a day, 7 days a week, through a toll-free number for making or confirming reservations. Today, Web-based OLTP systems enable passengers to bypass reservation agents and directly access an airline's Web site to check seating availability, select seats and process reservation requests. If the airline's OLTP system is unavailable, dissatisfied customers might turn to a competitor's Web site for similar services, resulting in a revenue loss for the airline.

Highlights

Predicting transaction and user loads for Web-based OLTP systems is harder than for traditional OLTP systems.

As businesses move from traditional OLTP to Web-based OLTP environments, system and database availability become even more critical. In traditional OLTP environments, the numbers of users and transactions are relatively predictable, allowing administration and maintenance downtime to be scheduled for slow periods. And because users—typically customer service agents—are employees who have been notified of the downtime, they could generally tolerate the delay.

With Web-based OLTP systems, however, both the number of users and transaction loads are less predictable, making administrative and maintenance operation downtime difficult to schedule. Because Web-based applications are accessible to huge numbers of users who expect the systems to be available around the clock, unexpected downtime on a Web-based OLTP system impacts significantly more users than a traditional OLTP system.

The ability to respond to market changes quickly means that decision support applications must be increasingly integrated with transaction systems.

Decision support processing

DSS applications provide data that enable business leaders to make informed business decisions, such as comparing sales figures from one week to the next and projecting revenue figures based on sales assumptions. Any inability to execute queries can delay such vital data analysis.

Second-generation Web sites dynamically process information and personalize customer experiences, which in turn require DSS applications to integrate more closely with traditional OLTP systems. And this integration must be enabled without impacting availability and performance of the OLTP system.

Highlights

System outages that occur while populating data warehouses or data marts impact not only IT productivity but also end-users' ability to access information in a timely manner.

The IBM suite of high-availability solutions offer a wide range of utilities for clustered and SMP architectures.

Database availability is equally important for populating data warehouses and data marts. Because of the frequency with which data is loaded into and unloaded from data warehouses and data marts, administrators often have a limited window for performing such operations. Thus, unanticipated downtime during load operations delays a user's ability to produce timely results.

The IBM suite of high-availability solutions includes online administrative utilities for backup and recovery, reorganization of tables, enterprise replication, cluster and data failover capabilities, software mirroring and more.

High-availability features for clustered architectures

Clustered architectures are necessary to achieve the highest levels of availability. To meet this requirement, IBM Informix Dynamic Server (IBM IDS) offers an additional set of continuous availability features that include cluster manager software, high-availability data replication, automatic client connection failover with high-availability data replication (HDR) and enterprise replication.

- *Cluster manager software solution*. To enhance availability in multinode environments, many operating system vendors provide cluster manager solutions to connect groups of servers, or nodes. These cluster manager solutions deliver a high level of data and application availability by providing failure detection, communication of failure to other systems and applications, system-level recovery and restarting of cluster-aware applications on the cluster's surviving node.

Cluster manager software allows each node in a cluster to run application software independent of other nodes (*Figure 1*). Disks among the nodes are either shared or are easily accessible by other nodes in the cluster system so that, if one node fails, the cluster manager software automatically switches the workload from the failed node to a surviving node.

Highlights

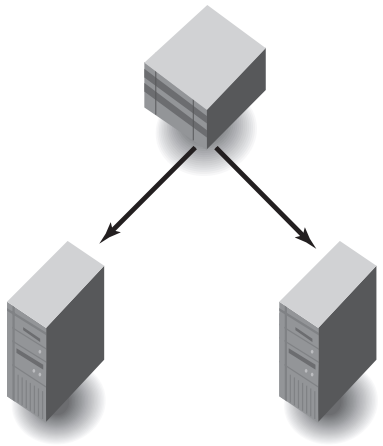


Figure 1: IBM IDS with cluster management

Cluster manager software allows each node in a cluster to run application software independent of other nodes, improving database availability between nodes in the event of a failure in one node.

By using third-party cluster manager solutions for node failure detection and notification, IBM IDS helps ensure continuous database processing. When a node failure is detected, the cluster failover facility automatically switches a database server from the failed node to a second node. This allows users of the failed node to continue database processing on the surviving node, significantly improving database availability between two SMP nodes.

IBM IDS performs fast recovery, restoring the database to physical and logical consistency.

The cluster failover process promptly creates a second instance of the database on a surviving node and performs automatic switchover, transferring ownership of the disks from the failed node to the surviving node. IBM IDS then performs a fast recovery operation, which restores the database to a physical and logical consistency. During this recovery process, the database is restored to the state of the last checkpoint. All of the transactions that had been omitted since the last checkpoint are rolled forward, and all uncommitted transactions are rolled back. After the database on the surviving node has completed its recovery, the cluster software automatically restarts the applications that had been running on the failed node.

Highlights

The failover cluster facility in IBM IDS can be implemented in both active/passive and active/active configurations. Active/passive is a configuration in which one node runs the database server and the second node acts as a hot standby for the first node. Active/active describes a configuration where both nodes run an instance of the database server. In the event of a node failure, the surviving node acquires the workload of the failed node.

With high-availability data replication, if a primary server fails, the secondary server rolls over any uncommitted transactions and takes over as the primary server.

- *High-availability data replication*. IBM IDS provides HDR, which uses two active instances of the database. The instances can be on the same system or on two different systems. When two different systems are used, they can be located anywhere because HDR replicates the primary instance to the secondary instance over a network. This replication is performed by copying IBM IDS database log records from the primary system to the secondary system as they are written.

With high-availability data replication the switchover between a primary and secondary server is transparent to the client.

The second server is active in a read-only mode and operates in fast-recovery mode. The secondary instance receives the log records from the primary instance and immediately applies the log record locally. This method ensures that at any given time the secondary system is only a few seconds behind the primary system. If the primary node fails, the secondary node rolls back any uncommitted transactions and then becomes the primary server. As the secondary server is initialized and always current relative to the primary server, the secondary server can become the primary server in a matter of seconds.

HDR can be configured to run in two different modes—synchronous and asynchronous. In synchronous mode, transactions do not commit on the primary server until the secondary server has received each log record comprising the transaction. In asynchronous mode, each transaction is allowed to commit immediately. Synchronous mode is required for applications that require absolute transactional consistency. Asynchronous mode is used for applications that require higher performance.

- *Automatic client connection failover with HDR.* When HDR is used in conjunction with an external cluster manager, clients are transparently reconnected to the current HDR primary server. If a failure occurs, the cluster manager executes two key functions. First, it executes scripts that convert the current secondary server to the current primary server and manages the restart of the failed server. Second, it swaps IP addresses of the two servers. Such "swapping" is critical because it ensures that all clients reconnect only with the current primary server. Because IP address reconfiguration is used, the change in servers is completely transparent to the client.

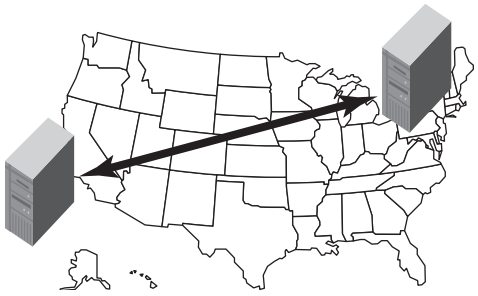


Figure 2: Geographic clusters with enterprise replication.

Highlights

Data replication can be effectively used to enhance database availability.

- *Enterprise replication*. Data replication is increasingly being used to enhance database availability. Replication can minimize, and sometimes even eliminate, both planned and unplanned downtime. To ensure database availability, enterprise replication in IBM IDS replicates the entire database or a portion of the database to a secondary server (*Figure 2*). This option is useful for creating a hot standby server to take over processing in case the primary server fails.

Enterprise replication is ideal for providing geographical availability. It supports active/active replication, enabling both sites to perform useful work in normal operation. If one site fails for any reason, such as a power failure or an Internet connection breakdown, work can be switched to the other site with minimal interruption. Active/active configurations can also survive Internet failures. Both systems can remain operational and the databases are reconciled when connectivity is restored.

IBM IDS supports a full peer-to-peer replication model with update-anywhere capability.

The enterprise replication capability of IBM IDS also supports a full peer-to-peer replication model with update-anywhere capability. Enterprise replication protects against primary system failures by replicating data asynchronously to one or multiple secondary sites. Any updates at the primary site, including changes to the global catalog, are automatically propagated to the secondary site, ensuring that all sites have consistent data replication. Transmission of updates can be immediate or time-driven, in which case the database administrator (DBA) can specify the time intervals for the updates.

Updates can also be event-driven, for example, triggered by a transaction commit or as specified by the user. IBM IDS enterprise replication employs a reliable message-delivery mechanism, which stores data locally and propagates it to the remote server as a separate transaction. If a server or network fails, the surviving server can continue to service users, thus ensuring a high degree of fault tolerance. After the failed server or network is restored to operation, all changes to the source database are propagated to the database on the affected server.

Highlights

SMP hardware vendors are building a high degree of fault resilience into their solutions with such technologies as error-correcting memory.

Online maintenance and administration is essential for high-availability SMP architectures.

Continuous availability features for SMP architectures

To respond to the increased demand for high availability, SMP hardware vendors are building systems with greater fault resilience, incorporating components with higher mean time between failures (MTBF), stabilizing the system's operating environment and employing technologies such as error-correcting memory and N+1 power supplies and cooling fans. Storage subsystems are greatly improved, and technologies such as redundant arrays of inexpensive disks (RAID) and online replaceable disks and tapes can significantly increase data availability.

The database management system is another critical factor. To enhance database availability and stability on stand-alone SMP systems, IBM IDS provides a variety of continuous-availability features, such as online maintenance and administration, fault resilience and enhanced problem diagnostics.

- *Online maintenance and administration*. To minimize database maintenance and administration downtime, IBM offers a suite of utilities that performs tasks online, such as database tuning and reorganization, and backup and recovery. For tasks that must be performed offline, data partitioning enhances availability, allowing a portion of the table to be taken offline for administration while the rest of the table remains available for user processing.

- *Dynamic tuning*. Regular database tuning is critical for ensuring efficient allocation of resources for fast response time. Because database tuning often requires the database to be taken offline, administrators usually postpone these tasks until response time has deteriorated to an intolerable level.

Highlights

IBM IDS performs a wide range of database tuning tasks online, including dynamically adding more server processes during peak usage to maximize availability and performance.

IBM IDS is capable of performing database tuning tasks online. These tasks can be accomplished transparently, without any impact to users or applications on the system. Database server processes can be allocated and retracted to adjust to the processing load. For example, an online retail Web site that experiences predictable surges in orders during the noon hour can dynamically add more server processes between 11 a.m. and 2 p.m. to provide faster order processing time during the anticipated peak period.

Another example of online tuning is shared-memory allocation that can be dynamically adjusted on an on-demand basis and can reconfigure memory usage online. After the memory is freed by the database, it can be reclaimed for operating system usage. Additionally, monitoring and fine tuning system parameters, such as CPU and memory utilization, asynchronous I/O queuing, available disk space and partitioning scheme, can be performed online.

- *Online table reorganization*. The table schema, or table reorganization, can be altered using the *alter table* command to add and delete a column; add, drop and modify data constraints placed on a column and change extent size. For example, a new column can be added in the CUSTOMER table to reflect the date of the last order to determine whether the customer is a current client. Or rarely used columns within a large table c an be deleted to reduce disk space usage.

DBAs can alter table schema without rendering the table unavailable for normal use.

To increase database availability, IBM IDS allows DBAs to alter table schema without rendering the table unavailable for normal use. Furthermore, alteration of a table occurs in place, so the changes are made as rows are updated without requiring creation of a second copy of the table. This improves performance and increases table availability, with minimal space requirements.

Highlights

- *Data partitioning*. Data partitioning enables large tables and indexes to be intelligently divided into smaller partitions and distributed across multiple disks. In addition to increasing performance and scalability, data partitioning also improves database availability. Data partitioning enables all maintenance operations, including load, index builds, backup and recovery, to occur one partition at a time—leaving the remaining portions of the table accessible for user transactions.

IBM IDS supports a wide range of partition schemes that can be monitored and tuned online.

An example of data partitioning is a customer order table that is partitioned by individual states. If the disk containing customer information fails unexpectedly, other partitions of the table remain unaffected while the partition of one state, say California, is restored to another disk. This allows users to continue processing orders for the remaining 49 states. IBM IDS supports a wide range of partition schemes that can be monitored and tuned online when necessary. These include Simple Round-robin, in which every record goes to the next partition in the sequence, and Expression, in which each partition receives a set of records based on its key values.

- *Data skip*. Data partitioning can ensure high availability through data skip, which allows users to bypass portions of the database in the event of a disk failure. This option is especially useful during execution of a complex query, when an unexpected disk failure can force the entire query to abort. Rather than canceling the entire query, which may already have taken hours to execute, the data skip option can be used to bypass the failed partition, allowing the query to complete its execution.

Data skip can be used to bypass a failed partition, allowing a query to complete its execution.

In the example of the customer order table, customer orders are partitioned by states. Suppose the disk containing the California partition fails during the execution of the following decision support query:

```
Select sum (total_dollar)
where (date_year = 1997) and
(product_code=123)
from ORDER
```

The data-skip option can be used to skip the failed disk, allowing the query to continue summing orders for other states. After the data on the failed disk has been restored, a separate query can be issued to sum the California orders. The results can then be added to the initial query.

- *Alter fragment*. After a table and its associated indexes have been partitioned, they can be altered using the *alter fragment* command. Partitioned tables and indexes can be modified by combining tables that contain identical table structures into a single fragmented table, also called an *attached table*, or by detaching a table fragment from a fragmentation strategy and placing it into a new table, called a *detached table*. Attached and detached tables are often used when limited disk space necessitates moving outdated data from disk onto other forms of storage media.

Using the *alter fragment* command to attach and detach table fragments sometimes requires indexes to be rebuilt, which can compromise table availability. To improve availability, the *alter fragment* command searches for reusable indexes before creating new indexes on the altered table. If portions of the existing indexes are reusable, the command instructs the server to build indexes on the table fragment only where indexes are unusable. By checking for reusable indexes, the index build process is minimized and sometimes eliminated entirely. This results in faster response time during execution of the *alter fragment* command and, consequently, higher data availability within the altered table.

This feature is especially useful when a customer maintains a rolling window of data, for example, one table containing 12 months of data—each month within its own fragment. A second table would contain only data from the current month. At the end of each month, the table containing the current month's data can be attached as a new fragment to the 12-month table and the oldest month's data fragment can be detached. This functionality provides customers with a clean, fast import of data with minimal impact to the base table. In this example, the oldest month is June 1996. Orders are stored in dbspace db0696. The alter fragment command detaches dbspace db0696 from the ORDER table and places it into the old_ORDER table:

```
alter fragment on table ORDER
detach
db0696 old_ORDER
```

After old_ORDER is created, it can be copied onto another form of storage and deleted from the disk. To store orders received for the current month, a new table, new_ORDER, is created. Because ORDER and new_ORDER use the same table structure, the alter fragment command can be used to attach the two tables into a single fragmented table. The command for attaching new_ORDER to ORDER table is shown below:

```
alter fragment on table ORDER
attach
new_ORDER
```

Highlights

When db0696 is detached from the ORDER table, if the index associated with the ORDER table uses the same fragment strategy as the table (monthly), the command simply instructs the server to drop the index fragment for db0696 and update the system catalogs. This process is much faster than rebuilding the entire index for the ORDER table. Similarly, when new_ORDER is attached to ORDER, the command recognizes that the existing index on the ORDER table can be reused. Thus, it instructs the server to build indexes on only the new_ORDER portion of the table and update the system catalogs accordingly.

While online backup is critical to maintain database availability, online restore is essential to bring the database back online quickly in the event of a failure.

- *Online backup and recovery*. Database backup is an important administrative task that must be frequently performed to avoid data loss. For 24x7 operations, not only is an online backup solution critical for maintaining database availability, but online recovery capability is equally important for bringing the database online in the shortest amount of time in the event of an unexpected failure.

ON-Bar backup and restore utility, which is used with IBM IDS Version 7.21 and higher, offers various features to let administrators perform backup and restore functions without bringing down the database. ON-Bar supports online backup, which lets administrators back up the entire database while the database continues to run. ON-Bar also supports online restore, which lets administrators recover non-critical database objects while the database server is online. With online restore, users have continual access to the database while sections of the database are being recovered.

Highlights

ON-Bar also supports dbspace-level backup and restore, allowing backup and restore operations to be performed one dbspace at a time, so other portions of the database can remain available. In the customer order table example, the administrator could perform a backup one state at a time, without affecting the dbspaces containing other states.

Similarly, a dbspace-level restore can significantly improve availability by enabling the administrator to restore a database to the lowest level of granularity. For example, if the database is partitioned across six disks and disk number four unexpectedly fails, the administrator need not restore the entire database—only the dbspaces stored on disk four. The remaining disks remain available to end users for processing.

Incremental backup can significantly reduce the time for performing backup operations.

Dbspace-level restore can also enhance availability of a full system restore. If a server crash corrupts the database and requires the entire database to be restored from the backup media, the administrator must recover all of the critical dbspaces, such as the root dbspace, offline. After all critical dbspaces have been restored, the database can be brought back online to recover the remaining tables. If the tables have been partitioned across multiple storage devices, they can be restored in parallel to speed the recovery process. After a dbspace has been restored to physical and logical consistency, the data in that partition can immediately be made available to users for transaction processing while the remaining dbspaces are being recovered.

DBAs also have the option to use incremental backup. Using the date/time stamp located on every database page, incremental backup copies only those pages that have been modified, rather than backing up an entire partition or table. With large tables that have relatively few updates, incremental backup can significantly reduce the time for performing backup operations—ensuring data protection while providing the highest level of database availability.

Highlights

- *External backup*. The ON-Bar backup and restore utility provides an effective means for creating an internal backup of the database that can be used to ensure database availability in the event of failure. To further assure availability, customers usually create an external system backup for disaster recovery. External backup uses proprietary hardware and software technologies to create simultaneous copies of data to host independent, local and remote sites. In this way, external backup is faster and can be restored in a similar setup environment.

To ensure maximum database availability, IBM IDS supports both internal and external backup.

To enable external backup, IBM IDS provides an administration command to force a checkpoint, which flushes the buffers to the disk and blocks the server from accepting any implicit or explicit transactions. After the external backup has been performed, another command is issued to undo the blocking, and normal server operations can be resumed. Users can then perform an internal backup using the ON-Bar backup and restore utility.

- *Restartable restore*. Sometimes an I/O error occurs on the tape, or other types of error occur within the servers during a physical or logical restore. When this happens, the entire restore process must be restarted from the beginning. To decrease the time required to perform a restore following an error, restartable restore allows the restoration to be restarted close to where the original restore failed. Depending on the quantity of data to be restored and where the data error occurred, this feature can significantly improve server availability.

Restartable restore reduces the time required to perform a restore following an error, by restarting near the original restore failure point.

If a user performs level-0, level-1 and level-2 backups for dbspace1, dbspace2 and dbspace3, and a restore of the three dbspaces is attempted and fails during the level 1 restore of dbspace2, the restarted restore performs the level 1 and level 2 restores for dbspace2, and level 0, 1, and 2 restores for dbspace3. The restore is skipped for dbspace1, as well as the level 0 restore

for dbspace2, because they had been successful during the original restore. During the logical restore portion, this feature enables the server to replay logs, starting with the log that had the most recent checkpoint before the error occurred.

If a failure occurs during a physical restore, restartable restore restarts the restore at the last non-complete dbspace. If a failure occurs during a logical restore, restartable restore restarts from the last checkpoint. Restartable restore is supported in both cold (offline) and warm (online) recovery of a physical restore. However, restartable restore for logical restore is supported only during cold recovery.

- *Oncheck utility*. IBM provides a complete suite of utilities to ensure full data integrity and optimal data consistency. The Oncheck utility performs checks to search disk structures for inconsistencies, repairs index structures that contain inconsistencies and displays information about the disk structure. To increase table availability and improve concurrency, Oncheck eliminates the need to lock the table while checking indexes. This allows users to continue to access the database while checks are being performed. By eliminating the need to place locks on a table that is being checked, Oncheck significantly enhances concurrency while ensuring an optimal level of consistency.

Oncheck no longer requires that physical and logical logs be checked during reserved page checks, because the reserved page check needs to be quick so that a server that is down can be brought back online quickly. For this reason, Oncheck lets the user decide whether or not the logs should be checked during the operation.

Highlights

IBM IDS offers features designed to work around faults, ensuring high fault resilience.

IBM IDS supports both hardware and software mirroring when supported by the operating system, system software and underlying hardware.

Fault resilience

IBM IDS offers a host of features that are designed to work around any faults that could cause a database to shut down. These features include database and log mirroring, fast recovery, enterprise replication and cluster failover.

- *Database and log mirroring*. Database and log mirroring provide database administrators a means to recover data in the event of a media failure, without having to take the database server offline. This method is ideal for protecting critical data that requires high reliability. Examples of data that should be mirrored include root dbspace and logical and physical log files. If the media that store any of these data fail, the database is immediately taken offline.

By supporting database and log mirroring, IBM IDS gives administrators the option of mirroring only the portion of the database that requires high availability. For example, if disk number one contains two tables, CUSTOMER and RECEIVABLES, the administrator may consider CUSTOMER to be a critical table and RECEIVABLES to be less important. With database mirroring, the administrator can choose to mirror only the CUSTOMER table.

IBM IDS supports hardware and software mirroring when provided by the operating system, system software and underlying hardware. Unlike database mirroring, where mirroring is achieved at the database level, hardware mirroring is achieved at the disk level. Consequently, the entire disk is mirrored, eliminating the flexibility to select which portion of the database to mirror. Therefore, in the customer order table example, hardware mirroring forces the administrator to mirror both tables on disk number one, which can result in a tremendous waste of disk space.

Highlights

With IBM IDS, unexpected shutdowns can be quickly overcome without any data loss or corruption.

Server failures can be prevented by isolating them at the session level by using exception handling routines in IBM IDS.

Fast recovery. Unexpected shutdowns can occur despite the best preventative measures. Fast recovery is an IBM IDS utility that brings the system online quickly, without data loss, to maintain full data integrity.

When invoked during a system recovery from an abnormal shutdown, fast recovery applies the transaction logs to the data files to restore the database to a state of physical and logical consistency. During this recovery process, the database is restored to its state at the last checkpoint. All committed transactions since the last checkpoint are then rolled forward, and all uncommitted transactions are rolled back.

Exception handling. A failure within a session often causes an entire server to shut down with an assertion failure. Such server failures can be prevented by isolating the errors at the session level so they do not affect the remaining server processing.

IBM IDS provides a set of routines to handle assertion failures and warnings within the server. These routines minimize server downtime by effectively pinpointing and diagnosing problem areas and returning appropriate error messages. For unavoidable server failures, these exception-handling routines provide better diagnostic information to assist in finding and fixing problems.

Enhanced problem diagnostics

In the event of a server failure, IBM IDS offers several enhancements to assist technical support personnel with problem diagnostics, analysis and resolution. These enhancements help pinpoint problem areas more quickly, allowing users to bring the server back online as quickly as possible.

Smarter diagnostics consist of enhancements in six areas—event alarms, fault isolation, shared memory dumps, stack tracing, additional utility options and thread blocking routines. These features enable quick resolution of reported problems.

Conclusion

To respond to the increasing demand for higher database availability, IBM IDS offers a wide range of features to provide around-the-clock database processing. These features minimize planned downtime by allowing administrators to perform database maintenance operations online and reduce the impact of unplanned downtime by working around any faults that may occur. Combined with high-availability features provided by hardware vendors, IBM IDS helps ensure a continuous database processing environment ideal for mission- and business-critical processing.

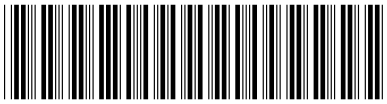
For more information

Please contact your IBM marketing representative or an IBM Business Partner, or call 1-800 IBM CALL within the U.S. Also, visit our Web site at **ibm.com**/software/data/informix/ids

IBM Corporation
Silicon Valley Laboratory
555 Bailey Avenue
San Jose, CA 95141
U.S.A.

Printed in the United States of America
09-01

Printed in the United States on recycled paper containing 10% recovered post-consumer fiber.



GC27-1495-00