



Unlock the Mysteries of the DB2 Log

TABLE OF CONTENTS

- OVERVIEW 1
 - » High-Speed Apply Engine 1
 - » Logical log 1

- RECOVERY 2
 - » Dropped Object Recovery: 2
 - » Quiet Point Analysis: 2
 - Quiet point in action 2
 - » Application Transaction Recovery 3
 - Classic point-in-time recovery 3
 - » UNDO – remove only the bad transactions 3
 - UNDO in action 4
 - » REDO - re-apply only the good transactions 4

- DATA MIGRATION 4
 - Data migration in action 5

- AUDIT 5
 - » Logical Log 5
 - Logical log in action 6

- HIGH SPEED STRUCTURE CHANGE. 6

- SUMMARY 7

OVERVIEW

DB2 records information about all data change activity on a table or index for backout and recovery purposes in the DB2 log. Log data is useful for recovery, auditing, and data migration, but it is not easy to understand the contents of the log. Getting the information you need is not easy either.

BMC Log Master for DB2 allows you to access the information in the DB2 log for advanced recovery techniques, auditing, and data migration. BMC Log Master for DB2 extracts information from the log that can be used as input into application-centric post processing, and the integrated High-Speed Apply Engine speeds the batch process of applying the information.

With BMC Log Master for DB2, you can:

- » Identify, analyze, and recover from problem transactions
- » migrate data
- » audit log information
- » automate the recovery of dropped objects
- » create logical log files
- » analyze quiet points
- » analyze Data Capture Changes (DCC) impact
- » generate data definition language (DDL) statements or objects

BMC Log Master for DB2 enables you to quickly and easily specify the criteria for a particular event. You will always specify an input, an output, and a filter.

- » BMC Log Master for DB2 can use BSDS, the DB2 log (active and/or archive logs) or a BMC Log Master for DB2 logical log as input. The logical log is an extract from the DB2 log previously built by BMC Log Master for DB2. For example, you may set up a process to extract a broad range of information from the DB2 log on a daily basis, and save it to a logical log data set. Subsequently, you may want to report further on some event from the past, and then the logical log can be your source instead of the DB2 log.
- » You can create several outputs from one pass of the DB2 log. The output can be a report, some SQL or DDL to be used for playback, the logical log, and/or a load file for populating another DB2 subsystem or another database management system (DBMS).
- » Use the WHERE clause as a filter to extract data from the DB2 log, much as you use SQL to create a result set from a DB2 table. The filter contains a predicate based on the unit of recovery, log record header, and/or detailed data information such as the table column value.

HIGH-SPEED APPLY ENGINE

The High-Speed Apply Engine supports DB2 on z/OS; DB2 for Linux, UNIX and Windows; and Oracle for UNIX and Windows as target systems. The High-Speed Apply Engine executes any ANSI SQL input and speeds execution of the input SQL through extensive parallelism and conflict resolution techniques. You can also use the High-Speed Apply Engine to speed application development by replacing DSNTTEST2, DSNTIAUL, and other dynamic SQL processing programs.

LOGICAL LOG

BMC Log Master for DB2 provides a unique form of output that can help you use and reuse DB2 log data easily - the logical log. The associated logical log control file describes the content of the logical log. The format of both of these files is published and includes easy-to-use fields designed for use by application programs. An application program or a person can easily read the logical log control file.

You can create logical logs as the only output of a BMC Log Master for DB2 job, or in addition to generated SQL and reports. Use the logical logs as input to application programs, input to report generators, or as permanent records of activity for audit needs requirements.

You can even use logical logs as input to BMC Log Master for DB2, saving the time needed to scan the DB2 log multiple times for the same records. You can generate the same reports and other forms of output from a logical log as from the DB2 log.

The logical log stores data for audit requirements, which is especially useful after the DB2 log has expired. Because the associated database schema is recorded in the control file, you can still audit data even if the schema was altered, dropped, or recreated between the time the logical log was created and when the data is needed for audit.

RECOVERY

BMC Log Master for DB2 provides powerful functionality to recover dropped objects, discover quiet points for point-in-time recovery, recover to a specific point in time, and perform transaction-level recoveries (undo and redo transactions).

Unfortunately, transactions sometimes update a database erroneously. It can be difficult to identify, analyze, and correct problem transactions. You might need to preserve transactions that were not in error but affected the same rows as the problem transactions. These types of problems can require complex analysis. BMC Log Master for DB2 handles problem transactions as follows:

- » Backout integrity checking compares changes of interest to any subsequent updates and shows problems and potential solutions in concise reports.
- » The SQL generator offers features to undo or redo specific transactions. The WHERE clauses generated are based on available index information. You can control whether updates resulting from referential integrity or trigger activity should be part of the correction process.
- » High-Speed Apply Engine executes the generated SQL statements and processes the logs for undo, redo, or migration.

DROPPED OBJECT RECOVERY:

BMC Log Master for DB2 automates the recovery of objects that were dropped from the DB2 catalog. Using information in the DB2 log, BMC Log Master for DB2 can create the following types of output:

- » data definition language (DDL) statements to re-create the dropped objects in the database
- » SYSIN syntax for BMC RECOVER PLUS for DB2 or DSNICOPY (using this syntax, BMC RECOVER PLUS for DB2 can recover the data of the dropped object)
- » SQL statements to move the recovered object into the original location in the database (used only to recover a dropped table)
- » DB2 commands to rebind application plans that were invalidated when the object was dropped
- » DB2 syntax to run a Check Data utility to take the table spaces out of check pending status

BMC Log Master for DB2 generates JCL to execute the different types of output in the correct order to accomplish the DROP RECOVER.

QUIET POINT ANALYSIS:

BMC Log Master for DB2 can analyze the log for a given time frame to determine log ranges during which no transactions were in process for a set of table spaces. This quiet point discovery can insert a QUIESCE row into SYSCOPY for the selected table spaces. You can then use this information to select a point in time for recovery.

QUIET POINT IN ACTION

A bank was running a quiesce each night that caused 300-400 ATM transactions to fail. By using BMC Log Master for DB2 to discover a quiet point, the bank eliminated the failures and was able to process the ATM transactions without problems.

APPLICATION TRANSACTION RECOVERY

When you have a database failure, you recover the entire database. But if you have a transaction problem, you may be able to recover just the failed transaction and save the time, resources, and outage that would be required for a full database recovery. You can recover from application-level problems such as the following:

- » Bad data entered
- » Wrong job run
- » Bad code or transaction moved to production
- » DBMS bug
- » Just testing
- » Program being run twice
- » Someone updating the database erroneously or maliciously

The benefits of transaction-level recovery are clear: you save the resources and time it would take to execute a complete database recovery and you save the costs and effort of writing specific recovery programs for each error. SQL-based solutions for backing out and reapplying changes allow the database to remain online and available for other processing. By eliminating only the transactions in error, you save the CPU resources that would be required for a full database recovery. BMC Log Master for DB2 provides all of these benefits - plus strong filter capabilities (including transaction information, users, database objects, update type, column level detail) and robust reporting.

BMC Log Master for DB2 enables the following types of transaction recovery:

- » UNDO - Back out only the bad transactions
- » REDO - After performing a database point-in-time recovery, re-apply only the good transactions
- » Point-in-time (PIT) - Remove all the transactions since a given point-in-time and then manually rerun or reenter the work that was valid. BMC Log Master for DB2 can discover quiesce points to use for point-in-time recoveries.

When you are deciding whether to execute a transaction recovery or a full recovery, consider your recovery time objective (how long the data can be unavailable during recovery) and your recovery point objective (how much data is lost and must be re-applied).

CLASSIC POINT-IN-TIME RECOVERY

To recover to a specific point in time, you remove all of the transactions that occurred after the point in time. You can eliminate all transactions or manually rerun the valid transactions. The native IBM DB2 utilities support this basic transaction recovery process. After the point-in-time recovery, you must rerun valid operations that occurred after the time to which you recovered.

UNDO – REMOVE ONLY THE BAD TRANSACTIONS

You can undo problem transactions by generating UNDO SQL statements to reverse the transactions in error. The database and application remain online, and processing continues as normal.

BMC Log Master for DB2 reads the DB2 log and creates the SQL needed to undo the transactions in error. It changes the SQL as follows:

- » INSERT statements are converted to DELETE statements.
- » DELETE statements are converted to INSERT statements.
- » UPDATE statements are converted to modify the data to its state prior to the original UPDATE.

The portion of the database that does not need to be recovered remains undisturbed.

UNDO IN ACTION

A rogue transaction at an auto insurance company inadvertently changed a vehicle type in the vehicle database, increasing insurance rates for all policy holders with that type of vehicle. By using UNDO, the insurance company was able to correct the error and keep the database online during the transaction recovery.

A data entry error at a healthcare company changed the sex of all policy holders to male. UNDO processing enabled the company to correct the error and maintain availability.

REDO - RE-APPLY ONLY THE GOOD TRANSACTIONS

You can perform a point-in-time database recovery and then re-apply good transactions using REDO SQL. Instead of generating UNDO SQL for the transactions you need to eliminate. . Execute a standard point-in-time recovery to eliminate all the transactions after the recovery point, and then re-apply the good transactions.

The REDO process creates SQL statements that reapply only the valid transactions from a consistent point of recovery to the current time.

Because the REDO process does not generate SQL for the problem transactions, performing a recovery and then executing the REDO SQL can restore the table space to a current state that does not include the problem transactions.

When redoing transactions in an environment where availability is crucial, you could bring the application and database online after the point-in-time recovery. You can execute the REDO SQL while the database is online, reducing application downtime.

Consider using REDO transaction recovery when the number of transactions that need to be undone is very large.

DATA MIGRATION

BMC Log Master for DB2 helps you migrate data between databases, systems, and platforms. Instead of unloading and loading databases multiple times, you can replicate/propagate just the data that was changed since the last full load or last change replication. For example, you may want to move only production changes to data warehouses, decision support, or cloned test systems. This is strict data migration; no transformation occurs. After you capture the data, the High-Speed Apply Engine applies it to the new database, system, or platform.

Data warehousing applications typically replicate huge databases on a periodic basis. Often, most of the data in the database is unchanged since the last replication. BMC Log Master for DB2 migration support discovers the changes to the source DB2 tables and captures the data in a logical log. It can then apply the data with SQL processing. If you choose the SQL method, the High-Speed Apply Engine can speed the dynamic SQL and allow for error handling.

BMC Log Master for DB2 keeps track of the log range processed by the last execution of BMC Log Master for DB2, and picks up where it left off at the next run. This is called "ongoing support". The subsequent run captures any open URIDs (or non-externalized pages), so the target file always receives complete updates. You set the interval between runs, so your target file can receive updates daily, or hourly, or as often as you choose. Some users capture as frequently as every 90 seconds, while others capture once a day.

For UPDATE SQL BMC Log Master for DB2 completes the row information without requiring data capture turned on for the tables to replicate.

After you create your target environment, BMC Log Master for DB2 can use the information from the log records of the source system to generate SQL statements for the target system. You can execute these statements against your target tables to bring them up to date. You can:

- » select only certain columns for migration/replication
- » include or exclude updates caused by referential integrity
- » include or exclude updates caused by activity defined within a trigger
- » migrate/replicate only complete transactions and automatically stage incomplete transactions for the next migration
- » create output load files for bulk migration using IBM or BMC Load utilities
- » migrate/replicate database changes to other RDBMS platforms. You can use ANSI SQL on any database that supports ANSI standards. The High-Speed Apply Engine operates on DB2 for z/OS as well as on DB2 for LUW and Oracle on Windows and UNIX.

To apply the source SQL to the target, you can use the High Speed Apply Engine. The High-Speed Apply Engine can use the logical log output. Using logical log output is much faster than generating SQL to migrate the data.

The High-Speed Apply Engine enables a multi-threaded apply process and gives you flexible control over the thread distribution. It enables robust conflict resolution and is restartable. It uses object mapping for renaming objects during a data migration process where the target object names do not match the source.

DATA MIGRATION IN ACTION

A local law enforcement agency uses BMC Log Master for DB2 when arresting and booking offenders. They migrate perpetrator data from a DB2 on z/OS database to a mug shot database on Oracle to match offenders to their photos.

AUDIT

Corporate scandals and accounting discrepancies have made good auditing capabilities a “must have.” While legislation may dictate audit requirements, DB2 DBAs know that each application is different and that a “one size fits all” compliance strategy probably will not work.

In IT, we have been dealing with audit requirements since the first application was written. Home-grown auditing facilities generally use a combination of log information, separate databases, tables, and VSAM files. We could generate audit reports on demand, but getting consistent results and maintaining the patchwork system can be problematic.

BMC Log Master for DB2 makes it easy to find the information you need, including transaction details, user data, catalog activity, and subsystem information. The online interface guides you through a simple set of steps for specifying search criteria to select specific log records. Filters enable you to narrow searches to specific users, tables, plans, or even column or value changes. You can save the information to output reports, DB2 load files, CSV files, or the logical log.

BMC Log Master for DB2 provides quick, easy, and flexible auditing capabilities – whether you need a one-time audit or an ongoing compliance strategy. You can generate reports by selecting from a list of standard report formats with a choice of presentation by user, job name, or plan name. The Catalog Activity report shows which user ID or correlation ID changed the structure of your DB2 objects (for example, changes to DB2 security with GRANT or REVOKE status). You can customize reports by using report templates. If you use your own application programs to extract information from BMC Log Master for DB2 reports, you can use templates to optimize the report data for your program.

LOGICAL LOG

The DB2 log contains limited resources addressable from the BSDS, and log information is segmented, partial, compressed, and encoded – in other words, it is almost impossible to decipher. Compressed data

requires appropriate compression dictionaries that are rebuilt during LOAD and REORG utility execution. Saving information about utility activity for a long time can be prohibitive. You can avoid the problems associated with keeping DB2 logs for transaction auditing by using the BMC Log Master for DB2 logical log.

Use the logical log to save information for future use, like weekly or monthly summary reports. You can group the information by application. You can save the logical log as long as you need to; it is not dependent on DB2 versions. The logical log is self-contained and enables you to map data. The information is completed, decompressed, and decoded. Each record shows "who, what, when" information in context. You can merge logical logs and use them as input to other processing.

LOGICAL LOG IN ACTION

A manufacturer produces a logical log 4 times each day based on DB2 catalog activity. The manufacturer rolls the logical logs into a monthly collection that enables them to do SOX reporting for all GRANT and REVOKE activity in the month as well as a separate report for Schema Changes (CREATE, ALTER, DROP) that occurred.

A telephone company tracks activity of all power users (SYSADM, DBADM) and cross references any activity to change control requests.

A financial institution tracks all plans with dynamic SQL capability (including DSNTEP2, SPUFI, and QMF) to look for ad hoc updates to specific databases.

HIGH SPEED STRUCTURE CHANGE

BMC Log Master for DB2 can improve data availability during schema changes. For example, a major table in a production database contains several million rows, and you need to change a column that contains date information from a CHAR(10) field to a DATE field. This type of change requires you to drop and recreate the table. You can shorten the length of the outage required for the change by using BMC Log Master for DB2.

1. Create a new table in a new table space with the required schema changes.
2. Unload data from the existing table.
3. Load data into the new table.
4. Use BMC Log Master for DB2 to capture changes to the existing table since the unload action.
5. Use BMC Log Master for DB2 to apply the database changes to the new table as SQL statements. During the capture and apply actions, the existing table is still available.
6. Capture and apply any changes since the start of the previous capture action.
 - a. Repeat the capture and apply process as needed until the time period is short enough to be within the time you need to stop a table space, rename two tables, perform some administrative actions, and start a table space. You can also keep this process going until you reach the best point in your processing cycle to tolerate the outage.
7. Put the original table space into read-only mode.
8. Capture and apply the last set of changes with BMC Log Master for DB2.
9. Reconcile the data to ensure that correct data exists in the new table.
10. Perform administrative actions:
 - a. Rename the existing table (keeping it for backup purposes).
 - b. Drop any views on the existing table.
 - c. Rename the new table to use the production name.
 - d. Drop and recreate any foreign keys defined on the production table (and resolve check pending status).
 - e. Recreate any views on the production table.
 - f. Rebind required application plans, packages, or collections.
11. Put the new table space into update mode.
12. Update existing maintenance procedures so that DB2 utilities use new table space and index names.
13. You can further automate the process by using BMC Database Administration for DB2.

SUMMARY

BMC Log Master for DB2 provides a comprehensive set of tools to extract DB2 log information and use it to help with transaction recovery, data migration, and auditing. The High-Speed Apply Engine high-speed apply engine shortens outages associated with data migration and schema changes. The Logical Log enables long-term reporting and trend information. And the reporting facility produces reports that even an auditor will love.

Business runs on IT. IT runs on BMC Software.

Business thrives when IT runs smarter, faster and stronger. That's why the most demanding IT organizations in the world rely on BMC Software across both distributed and mainframe environments. Recognized as the leader in Business Service Management, BMC offers a comprehensive approach and unified platform that helps IT organizations cut cost, reduce risk and drive business profit. For the four fiscal quarters ended June 30, 2010, BMC revenue was approximately \$1.92 billion.

