SYBASE®

**CROSS PLATFORM
DUMP AND LOAD FOR 12.5.3**

TABLE OF CONTENTS

CROSS
PLATFORM
DUMP AND LOAD

OVERVIEW

Sybase Adaptive Server Enterprise (ASE) version 12.5.3 introduces enhanced dump and load capabilities that address previous restrictions for moving databases across different platforms or restoring backups from an ASE server on one platform to an ASE server on a different platform. Previously, loading a database dump to an ASE server running under a different platform architecture was not supported, and transferring the database required the use of other utilities and commands such as bcp and sybmigrate. These tools are still useful, but in many cases are less efficient than dump and load.

It is now possible to easily reload databases across ASE platform combinations such as between Unix to Linux platforms and between 32-bit and 64-bit architectures, or between other platforms with different byte-storage architectures including backward compatibility with database dumps created with ASE 11.9 upwards. This enhanced flexibility makes it much simpler to manage databases and handle database migration in cross-platform environments. The purpose of this paper is to summarize the new cross-platform compatibility, explain how to use dump and load in a cross-platform environment, identify some restrictions, and provide some tuning and troubleshooting recommendations.

BACKGROUND

The Sybase Backup Server is used for all dump and load operations. The scope of this paper is limited to using the new cross-platform capabilities of dump and load for databases. The bcp and sybmigrate utilities are frequently used for extracting user and system data. In some cases, they have been used to work around cross-platform incompatibility issues. These new dump and load capabilities may provide a simpler and faster method of transferring large amounts of data, but overall backup and migration strategies are beyond the scope of this paper.

SUMMARY OF COMPATIBILITIES

The new capabilities support cross-platform dump and load between platforms with different byte-storage (endian) architectures. With this release, it is now fast and easy to load database dumps that were created:

- From ASE 11.9.x or later
- Under any supported operating system (Unix, Linux, Windows, MacOS)
- Under any supported platform architecture (32-bit, 64-bit, Big endian, Little endian)

While most users may be interested in specific migrations, i.e. Unix to Linux, or 32-bit to 64-bit, the cross-platform conversion can be used between any of these combinations.

USING DUMP AND LOAD IN A CROSS-PLATFORM ENVIRONMENT

There are no changes to the DUMP DATABASE function. This allows existing database dumps to be processed for the cross-platform conversion and also eliminates any need to upgrade or make changes to the source ASE server. The supported database dump file versions are 11.9.x, 12.0.x, 12.5.x (11.9 upwards).

As for traditional dump operations, the database must be transactionally quiescent before the DUMP command is executed, and the new cross-platform load feature will check the syslog records to make sure this was the case before processing the database dump. The recommended procedure is to:

1. Put the database in single user mode using **sp_dboption**
2. Run **sp_flushstats** and wait for the pages to be flushed
3. Checkpoint the database
4. Run DUMP DATABASE

The cross-platform conversion functionality has been added internally to the LOAD DATABASE function for the target ASE server, without any syntax changes to the DUMP command. The data conversion operations are performed automatically when LOAD DATABASE is invoked for the target ASE server. During the LOAD DATABASE operation, the byte storage of the database dump file is automatically detected. If the byte storage of the target ASE server differs, then the target ASE server performs the conversion to the new byte order as the database dump is processed. If byte reordering is required, there generally will be a performance impact on loading the database. Depending on the schema and data types involved, this may be 2-3 times more intensive than processing a database dump from the same byte order architecture as the target server.

LOAD DATABASE sample output :

```
1> load database pubs2 from '/linuxkernel_eng1/terada/test/dump.pubs2.sol32.1253'
Backup Server session id is: 54. Use this value when executing the 'sp_volchanged' system
stored procedure after fulfilling any volume change request from the Backup Server.
Backup Server: 6.28.1.1: Dumpfile name 'pubs2042890A14E' section number 1
mounted on disk file '/linuxkernel_eng1/terada/test/dump.pubs2.sol32.1253'
Backup Server: 4.58.1.1: Database pubs2: 1384 kilobytes LOADed.
Backup Server: 4.58.1.1: Database pubs2: 3078 kilobytes LOADed.
Backup Server: 4.58.1.1: Database pubs2: 3086 kilobytes LOADed.
Backup Server: 3.42.1.1: LOAD is complete (database pubs2).
Started cross-platform conversion for database pubs2.
```

Started cross-platform conversion for system objects.
Cross-platform conversion for database pubs2: 33 pages completed.
Completed cross-platform conversion for system objects.
Started cross-platform conversion for user objects.
Cross-platform conversion for database pubs2: 259 pages completed.
Completed cross-platform conversion for user objects.
Started cross-platform conversion for log records.
Cross-platform conversion for database pubs2: 1 page completed.
Completed cross-platform conversion for log records.
Completed cross-platform conversion for database pubs2.
Started estimating recovery log boundaries for database 'pubs2'.
Database 'pubs2', checkpoint=(1150, 19), first=(1150, 19), last=(1150, 19).
Completed estimating recovery log boundaries for database 'pubs2'.

MAKING THE ASE DATABASE READY

Although there are no syntax changes to the ONLINE DATABASE command, ASE will
automatically execute some additional operations after a cross-platform database load has
completed when ONLINE DATABASE is invoked. After a cross-platform load, ASE will:

- Truncate syslogs – using the completion of the load as a new checkpoint
- Fix up colidarray in sysstatistics – regenerating it using the new byte-order
- Check and rebuild indexes on system tables if they need to be regenerated
- Mark suspect user tables indexes (except clustered index on APL (all-pages-locked) table)

User table indexes may need to be rebuilt after the byte-order conversion to re-optimize the
search path because of changes to index row identifiers. Suspect indexes can be identified using
the **sp_indsuspect** stored procedure. Recreating these indexes for large tables can be a lengthy
process, so several options are provided. The indexes on the target server can be fixed by manually
dropping and recreating them, or by using a new stored procedure. The new stored procedure,
sp_post_xload, checks the user table indexes that have been marked suspect and rebuilds them
if necessary. It then also checks and rebuilds the following indexes:

- Non-clustered index on APL table
- Clustered index on DOL (data-only-locked) table
- Non-clustered index on DOL table

Completion of the stored procedure clears the suspect status of these indexes.

Sybase recommends manually dropping and recreating indexes for databases larger than 10Gb, and using the stored procedure to fix indexes in a single process for smaller databases.

RESTRICTIONS

- The database should be transactionally quiescent when the DUMP DATABASE command is used.
- Loading database dumps from ASE servers prior to ASE 11.9 is not supported.
- Loading a master database is not supported.
- Cross-platform loading is only available for database dumps, not transaction dumps.
- Cross-platform loading of password-protected dump files is not supported.
- Cross-platform dumping or loading using a remote backup server is not supported.
- Cross-platform loading of parsed XML objects is not supported.
- Cross-platform loading of embedded data structures stored in binary, varbinary, or image columns is not supported.
- After cross-platform load, all stored procedures and other compiled objects will be recompiled from source text at the first execution. This can be done proactively if desired using dbcc upgrade object.

TUNING AND TROUBLESHOOTING

To assist in diagnosing and correcting conditions that prevent successful completion of the cross-platform conversion, the following error codes are new:

3151 ERROR	Adaptive Server cannot load this database because the database that was dumped was not quiescent when the dump was performed. Run sp_flushstats before DUMP DATABASE and ensure that the database is not updated during the dump.
	• Returned when ASE finds that the dump file is not appropriate for cross-platform load. • There was a concurrent update during DUMP DATABASE. • There was an open transaction when DUMP DATABASE started. • Sp_flushstats was not run before DUMP DATABASE. • Checkpoint was not run before DUMP DATABASE. • The database was not put in a single user mode during DUMP DATABASE.

3161 AND 3162 ERRORS	LOAD DATABASE for database '%.*s' failed during cross-platform conversion: insufficient memory to allocate context structure. LOAD DATABASE for database '%.*s' failed during cross-platform conversion: insufficient memory to allocate translation dictionary.
	• Returned when ASE can not allocate memory for context structure or translation dictionary. • Need to increase procedure cache memory. (sp_configure 'procedure cache size') • Approximate calculation for memory needed in procedure cache 2KB(context structure) + 16KB(translation dictionary for system tables) + 146KB(work memory to build translation dictionary for system tables and user tables) + ((# of user tables) * 20 bytes) + ((# of indexes on user tables) * 8 bytes) + ((# of columns on user tables which needs translation + # of columns in index keys on user tables which needs translation) * 12 bytes) Example : 10,000 user tables + 40,000 indexes + 400,000 columns = 164KB + 200KB + 320 KB + 4.8 MB = 5.5MB

SUMMARY

With this release of cross-platform database dump and load capabilities, Sybase ASE significantly expands its capabilities to handle database transfer and restore across versions, operating systems and platform architectures. Database dump and load is generally much faster than other data transfer methods, so eliminating the previous cross-platform constraints will make it much faster and easier to manage data migration and other cross-platform operations.

It is now possible to easily reload databases across ASE platform combinations such as between Unix to Linux platforms and between 32-bit and 64-bit architectures, or between other platforms with different byte-storage architectures including backward compatibility with database dumps created with ASE 11.9 upwards. This enhanced flexibility makes it much simpler to manage databases and handle database migration in cross-platform environments.

SYBASE®