Altibase 7.3.0.0.1 Release Notes

(2023.08)

1. System Requirements

Minimum Hardware Requirements

- 1GB RAM (Recommended: 2GB)
- 1 CPU (Recommended: 2 CPUs)
- 4 GB free hard disk space (Recommended: 12 GB)

Operating Systems and Platforms

Altibase 7.3.0.0.1 can be run on the operating systems and platforms listed in the table below.

	Altibase Server	Altibase Client	Software requirements
AIX on IBM Power Systems			
AIX 6.1	•	•	
Linux x86-64			
Red Hat Enterprise Linux 6 Red Hat Enterprise Linux 7 Red Hat Enterprise Linux 8	•	•	- GNU glibc 2.12 ~ 2.33
Linux on Power			
Red Hat Enterprise Linux 6.5 and higher	•	•	- GNU glibc 2.12 ~ 2.33
Linux on Power (Little Endian)			
Red Hat Enterprise Linux 7.3 and higher	•	•	- GNU glibc 2.17 ~ 2.33
HP-UX Itanium (IA-64)			
HP-UX 11.31	•	•	

> Altibase 7.3 server and client both support 64-bit only.
>
> Altibase 7.3 is compatible with minor versions of Red Hat Enterprise Linux 6, 7 and 8.
>
> Java 버전: Altibase 7.3 is compatiable with JDK 1.8 and higher.

2. Release Nots

2.1 New Features

2.1.1 AKU(Altibase Kubernetes Utility)

AKU(Altibase Kubernetes Utility) is a utility that helps with synchronizing data or resetting synchronization information when Pods start and terminate in a Statefulset in Kubernetes.

2.1.2 AltiShapeLoader 1.0

altiShapeLoader is developed for importing and exporting shapefiles[1] and is based on the GeoTools, which is a Java-based open source framework.

2.1.3 Partial Support for JDBC 4.2

Altibase 7.3 provides partial support for JDBC API Specification 4.2. For more detailed information about the JDBC 4.2 API supported by the Altibase 7.3 JDBC driver, please refer to **JDBC User's Manual** - JDBC 4.2 API References. Additionally, for information on the changes and compatibility issues, please consult Changes and Compatibility Issues with Altibase 7.3 JDBC Driver section in this manual.

2.1.4 OpensSSL 3.0.8 Support

To enhance security, Altibase 7.3 supports the latest version of OpenSSL 3.0.8, and no longer provide support for OpenSSL versions 1.0.x. Altibase now extends its protocol support to include TLS 1.3, in addition to TLS 1.0 and 1.2. If users want to specify particular cipher algorithms for TLS 1.3, please set them in the server property, SSL_CIPHER_SUITES. For more detailed information, please refer to **Altibase SSL TLS User's Guide** - server properties to connect over ssl.

Additionally, Altibase supports the FIPS(Federal Information Processing Standards) module. To use the FIPS module with SSL, you must set the SSL_LOAD_CONFIG property to 1. Refer to Altibase SSL TLS User's Guide - Step4. FIPS module with SSL.

2.1.5 Funtionality Improvement - SQL Extension

VARRAY TYPE

Within stored procedures, the VARRAY type is now supported as a new user-defined type. The VARRAY type is capable of storing a sequence of data with the same data type. For more details, please refer to Stored Procedures Manual - VARRAY.

Anonymous Block

An anonymous block is a form of a stored procedure composed of a body block without a header, declared with a structure like DECLARE...BEGIN...END;. Anonymous blocks do not create PSM (Persistent Stored Module) objects, are not stored in the database, and do not return a value in the RETURN clause. Unlike stored procedures, anonymous blocks allow the use of INPUT, OUTPUT, and INOUT bind variables.

Internal Mode in C/C++ External Procedure

In Internal mode within External Procedures, it directly loads dynamic libraries and invokes external procedures from Altibase Server without the need for an agent process, resulting in improved efficiency compared to external mode. For more details, refer to **External Procedures Manual**.

Multiple Delete, Update

Provides support for multiple delete, multiple update statements. Refer to **SQL Reference Manual** - multiple_delete , multiple_update.

Regular Expression for Korean searching

Provides PCRE2 compatibility mode for support Regular Expressions used in Korean searching. Refer to **SQL Reference Manual**.

Fetch Across Rollback

To address the 'Fetch out of sequence' error issue, we now provide the 'fetch across rollback' feature.

Improved Queue Functionality with Delete Statement Control

Queue functionality has been enhanced to allow the use of DELETE statements. Additionally, new 'DELETE ON' and 'DELETE OFF' clauses have been introduced to control the execution of DELETE statements within queues. Refer to **SQL Reference Manual** and V$QUEUE_DELETE_OFF performance view for more information.

Sequence Restart Statement

Supports the restart_clause with ALTER SEQUENCE statement. Refer to **SQL Reference Manual** - restart_clause.

2.1.6 Funtionality Improvement - Spatial SQL

SRID(Spatial Reference Identifier) interface

SRID (Spatial Reference Identifier) is an identifier used to distinguish spatial objects, represented by a 4-byte integer, and can be applied to GEOMETRY columns. You can specify the SRID when creating a table and also alter the SRID using the ALTER TABLE statement.

The support for SRID introduces new ways to represent GEOMETRY data types:

- Extended Well-Known Text (EWKT) format: It includes SRID (Spatial Reference Identifier) information in the WKT format to represent spatial objects.
- Extended Well-Known Binary (EWKB) format: It includes SRID (Spatial Reference Identifier) information in the WKB format to represent spatial objects.

Spatial Functions

New functions have been introduced as follows.

- ASEWKT
- ASEWKB
- GEOMFROMEWKT

- GEOMFROMEWKB
- SETSRID
- SRID
- ST_Collect
- ST_IsCollection
- ST_LinestringFromWKB
- ST_MakeEnvelope
- ST_MakeLine
- ST_MakePoint
- ST_Point
- ST_PolygonFromText
- ST_Transform

Spatial object Creation Functions

New functions have been introduced as follows.

- ACSGetGeometrySRID

2.1.7 Funtionality Improvement - Replication

DDM Synchronization

DDL Synchronization is now possible through replication. To use this feature, you must set the REPLICATION_DDL_SYNC property to 1 on each node. Also, the REPLICATION_DDL_ENABLE property of each node must be set to 1 and the REPLICATION_DDL_ENABLE_LEVEL property must be set to the same.

To use DDL Synchronization, the following constraints must be verified:

- Replication must be operational on the nodes where DDL Synchronization is performed.
- The table names on both local and remote nodes for DDL Synchronization must match.
- The table partition names on both local and remote nodes for DDL Synchronization must match.
- The username for replication target users must be the same for DDL Synchronization.
- Only one node at a time can perform DDL Synchronization.
- The values of the REPLICATION_DDL_ENABLE and REPLICATION_DDL_ENABLE_LEVEL properties must be identical for each node.
- The Altibase Server version (5 digits) must be the same.
- DDL Synchronization is not permitted when using the Propagation option.

RECEIVE_ONLY option

A new option, RECEIVE_ONLY, has been introduced to prevent the transmission of transaction logs for changing data to other nodes. Creating a replication with the RECEIVE_ONLY option means it does not read transaction logs, ensuring that it does not affect the other node, even in the case of issues such as network failures.

2.1.8 Application Development Interface Extension and Improvements

InfiniBand Support

Supports Infiniband, which is based on Remote Direct Memory Access (RDMA) communication for high-speed data communication.

New JDBC Driver

- **Auto-loading of JDBC driver class**

  Automatic driver loading using the META-INF/services/java.sql.Driver file , eliminating the need for explicit Class.forName() class loading.

- **Wrapper Pattern Support**

  Supports the JDBC 4.0 standard interface for obtaining references to implementation objects from proxies. This allows obtaining JDBC objects from proxy objects created in Connection Pool.

  ```
  try (Connection sWrappedCon = dbPool.getConnection()) {
      if (sWrappedCon.isWrapperFor(AltibaseConnection.class)) {
          AltibaseConnection connection =
  sWrappedCon.unwrap(AltibaseConnection.class);
          ...
          ...
  }
  ```

- **National Character Set Support**

  Supports for the standard multilingual processing interface in the JDBC 4.0 specification.

- **Aborting Connections**

  Supports for the Connection.abort() interface for asynchronously terminating the physical connection to the database.

- **Standard Socket Network Timeout API Support**

  Support for the standard interface Connection.setNetworkTimeout() to set the socket response wait time from the database server.

- **Connection Management Enhancements**

  Support for Connection.isValid() to perform validation on Connection objects without a Validation Query

- **Large Update Counts Support**

  Support for executeLargeUpdate() and executeLargeBatch() for updating large numbers of records

- **Set Client Information Support**

  Support for configuring client application attributes (name) using Connection.setClientInfo()

- **java.sql.SQLType interface Support**
- **Automatic JDBC resource release using the Try-with-resources statement**

```
try (Statement stmt = con.createStatement()) {
      ResultSet rs = stmt.executeQuery(query);
      while (rs.next()) {
        String coffeeName = rs.getString("aaa");
        int supplierID = rs.getInt("bbb");
      }
    }
}
```

- **Support for using an enhanced for-each loop with SQLException**

```
catch(SQLException ex) {
     for(Throwable e : ex ) {
        LOG.error("Error occurred: " + e);
     }
}
```

2.1.9 Functionality Improvement - Stored Procedures

DBMS_STANDARD package

Offers a function for checking trigger events.

DBMS_METADATA package

A function for export object creation DDL statements and GRANT statements from database dictionary.

DBMS_SQL_PLAN_CACHE package

Provides a stored procedure that functions to keep or delete a specific Execution Plan in the SQL Plan Cache.

print_enable/print_disable procedure in DBMS_OUTPUT pakcage

Added the 'print_enable' and 'print_disable' procedures to enable or disable the 'println' function within PSM. These procedures are executed on a per-session basis.

sleep2 procedure in the DBMS_LOCK package

Added system stored procedure 'sleep2' to support microsecond sleep.

SYS_SPATIAL package

Provides the function to register and delete Spatial Reference System metadata in the SPATIAL_REF_SYS table.

2.1.10 Functional Improvement - Utilities

Added Platforms for altimon: AIX 7 and Power Linux LE(Little endian)

Supports altimon on AIX 7 and Power Linux LE(Little endian).

Added the AltiComp Commit Count Configuration Feature

New property, COUNT_TO_COMMIT has been added to enable the configuration of commit counts. Refer to Utilities Manual.

2.1.11 JDBC Adapter, oraAdpater Improvements

Support for LOB Data Types

A new property, ADAPTER_LOB_TYPE_SUPPORT, has been introduced to enable LOB data type support. To enable Lob data type support feature, set the value of ADAPTER_LOB_TYPE_SUPPORT property to 1 and then restart the adapter.

Offline Option

Offline option is a feature used with adapters (JDBC Adapter, oraAdapter) to handle the event of a failure in the Altibase server when applying changes from Altibase to the target database.Refer to **Adapter for JDBC User's Manual** - Offline Option or **Adapter for Oracle User's Manual** - Offline Option.

2.1.12 Performance Improvement

Improvement in TABLE LOCK Bottlenecks

The LOCK_MGR_TYPE property, previously used for specifying the TABLE LOCK manager type, has been removed. In its place, a new TABLE LOCK mode called 'light mutex mode' has been introduced to enhance performance and address TABLE LOCK bottlenecks.

Improvement Tablespace Manager Mutex Bottlenecks

Improvement the TABLESPACE MANAGER MUTEX by removing unnecessary LOCK.

Improved Disk Temporary Table Performance

Improved the performance of SQLs that use disk temporary tables, resulting in enhanced performance and reduced memory usage.

Enhanced Transaction Log Recording Performance

Changed the log compression algorithm to the faster LZ4, improving transaction log recording performance.

OLTP Scalability

- Enhanced Transaction Performance on Linux x86-64 with 24+ CPU Cores
- Improved Logging Structure for Memory Database Deletion (DELETE) Transactions
- Enhanced In-Place MVCC Operation for Disk Database Alterations
- Elimination of Unnecessary Transaction Logging during INSERT/UPDATE Transactions

- Improved Memory Allocation/Deallocation Bottlenecks during Transaction Log File Compression
- Enhanced Commit and Garbage Collection Thread Bottlenecks
- Improved Transaction Performance on Memory Databases
    - Eliminated bottlenecks in functions that trigger disk reads
    - Group Commit Log

Index Performance Improvement

- Reduced the time and memory usage for POINTER BASE index creation during server startup.
- Reduced the time and memory usage for VALUE BASE index creation during server startup.

Database Startup

Improved the management of threads used for index building during server startup.

Improve Transaction Performance for Volatile and Non-Volatile Memory Table

Improved volatile/non-volatile memory table transaction performance by simplifying the memory table object identifier tracking steps.

DEQUEUE in Parallel

eliminated bottlenecks that occur during parallel DEQUEUE operations.

Reduction in Prepare Time for Common Subexpression Elimination

CSE (Common Subexpression Elimination) is an optimization feature that identifies and removes redundant conditional expressions in query conditions. Improved the CSE execution algorithm and enhance the performance of related queries.

Simple Query Optimization on Memory Partitioned Tables

Previously, simple query optimization was only supported for memory tables. But now it extendes support to memory partitioned tables and enhanced DML performance of memory partitioned tables.

SERIAL FILTER EXEUTE

Improved the performance of performing row filters by serializing the Filter operator and optimizing the function call structure. The SERIAL_FILTER hint and SERIAL_EXECUTE_MODE properties have been added to enable this feature. You can see FILTER SERIAL EXECUTE in the execution plan. Serial Filter Enable can be confirmed by checking for 'FILTER SERIAL EXECUTE' in execution plan.

Performance Enhancement for Scalar Subqueries

Optimized the execution of scalar subqueries to improve their performance.

Performance Enhancement in the FOR Loop Clause of PSM

Performance Enhancement in Replication Sender

- Added functionality to decompress only the logs required for replication in compressed logs.
- Changed the xLog compression algorithm from LZO to LZ4.

Performance Enhancement for Migration

Improved performance of data insertion for large data migrations. A new option, -lightmod, has been introduced of iloader. Refer to **iLoader User's Maunal**.

JDBC fetch Performance

To enhance JDBC fetch performance, ResultSet object are now reused. When multiple ResultSet objects are created from the same PreparedStatement object, the first ResultSet object is reused. If you do not want to reuse ResultSet objects, you can change the value of the reuse_resultset property in the JDBC connection properties to false.

2.1.13 High Availability

Enhanced DDL PVO Stability

Improved reliability by strengthening exception handling in the DDL PVO phase.

Improved Protocol Validation

Imporved to prevent abnormal server termination and behavior caused by the transmission of invalid packets (malformed packets). Packet validity is now checked during protocol processing, and in cases of abnormalities, the client's connection is terminated, and diagnostic logs are generated. To enable this feature, the default value of CM_MSGLOG_FLAG has been set to 3, and the default value of SERVER_MSGLOG_FLAG has been changed to 15.

Enhanced Transaction Stability - Multiple Rollback Segment

The maximum number of concurrent transactions for disk table that can be executed simultaneously has been expanded from the previous 512 to 16,384

Improved Undo Tablespace Reusability

To enhance the stability of Undo tablespace reuse, unnecessary associations between the undo tablespace and disk indexes have been removed, thereby eliminating potential bug-related risks. Additionally default and maximum values of related properties have been changed to improve disk page space efficiency.

- INDEX_INITTRANS maximum value has been increased from 30 to 50.
- Default and maximum values for INDEX_MAXTRANS have been changed from 30 to 50.

2.1.14 DBeaver Package

Provide a package for DBeaver on Windows.

2.2 Changes

The following describes the features that DBAs and developers need to be aware of, which include additions, modifications, and removals.

2.2.1 Database Version

Version by Database Component

Altibase server/client Version	Database Binary Version	Meta Version	Communication Protocol Version	Replication Protocol Version
7.1.0.8.8	6.5.1	8.11.1	7.1.7	7.4.7
7.3.0.0.1	7.3.0	9.3.1	7.1.8	7.4.9

2.2.2 Compatibility

Database Binary Version

The database binary version indciates the compatibility of the database image file and log file.

The database binary version has been updated due to enhancements in the logging structure of log files. As a result, Altibase 7.3 and earlier versions are not compatible, requiring migration efforts when upgrading to Altibase 7.3.

Meta Version

Given that the major version of Meta has changed, it is necessary to reconfigure the metadata when upgrading from the earlier version to Altibase 7.3.

Communication Protocol Version

The patch version of the communication protocol version has been changed. The same major and minor versions ensure client backward compatibility.

> Client backward compatibility ensures that user applications developed using a older version of the Altibase library work well when connected to a newer version of Altibase.

Replication Protocol Version

Major and minor version of replication protocol version has not been changed, but the patch version has been updated. Thus, LAZY mode replication is compatible between older and newer versions, but EAGER mode replication and other additional features that require the same replication protocol version are not compatible.

> **Altibase Replication Backward Compatibility**
>
> Only LAZY mode replication guarantees backward compatibility.
>
> While EAGER mode replication, DDL synchronization and Offline Replication do not ensure backward compatibility since they require the same replication protocol version.

2.2.3 Others

aexport

To run Altibase 7.3 aexport, you must install the DBMS_METADATA package. Otherwise, the following error message will be displayed.

```
[ERR-91144 : DBMS_METADATA package does not exist.]
```

Changes and Compatibility Issues with Altibase 7.3 JDBC Driver

The Altibase 7.3 JDBC driver guarantees backward compatibility, but for some interfaces, the behavior has changed according to JDBC API Specification 4.2.

SQLFeatureNotSupportedException

The exception handling class for the following interfaces has been changed from SQLException to SQLFeatureNotSupportedException. SQLFeatureNotSupportedException is a subclass of SQLException, so existing user programs will continue to work without modification.

- Altibase.jdbc.driver.AltibaseConnection
  - setTypeMap(Map)
- Altibase.jdbc.driver.AltibaseStatement
  - setCursorName(String)
- Altibase.jdbc.driver.AltibasePreparedStatement
  - setArray(int, Array)
  - setRef(int, Ref)
  - setURL(int, URL)
  - setUnicodeStream(int, InputStream, int)
- Altibase.jdbc.driver.Blob
  - position(Blob, long)
  - position(byte[], long)
- Altibase.jdbc.driver.Clob
  - position(Clob, long)
  - position(String, long)
- Altibase.jdbc.driver.CallableStatement
  - getArray(int)
  - getObject(int, Map)
  - getRef(int)
  - getURL(int)
- Altibase.jdbc.driver.AltibaseDatabaseMetaData
  - getColumnPrivileges(String, String, String, String)
  - getUDTs(String, String, String, int[])
- Altibase.jdbc.driver.AltibaseResultSet
  - getCursorName()

- getArray(int)
- getObject(int, Map)
- getRef(int)
- getURL(int)
- getUnicodeStream(int)
- updateArray(int, Array)
- updateRef(int, Ref)

DatabaseMetaData

SPECIFIC_NAME column has been added to the results of the getProcedures(), getProcedureColumns(), getFunctions(), and getFunctionColumns() interfaces.
In Altibase 7.3 JDBC, SPECIFIC_NAME is implemented as follows.

```
ProcName(FuncName) + '_' + ouid
```

JDBC Connection Properties

- reuse_resultset
  - In Altibase 7.3, the default value is 'true', which means that ResultSet objects are reused.
  - In Altibase 7.1, the default value is 'false', which means that ResultSet objects are not reused.
- lob_null_select
  - When a LOB column's value is NULL, the JDBC connection property 'lob_null_select' has been introduced to control the behavior of getBlob() and getClob() functions.
  - In Altibase 7.3, the default value is 'off', which means getBlob() and getClob() functions return 'NULL'.
  - In Altibase 7.1, the default value is 'on', which means thease functions return LOB objects.
- getprocedures_return_functions
  - Configuration for including function information in the results of DatabaseMetaData.getProcedures() and getProcedureColumns() is provided. While the JDBC API Specification 4.2 standard excludes function information, Altibase 7.3 JDBC Driver includes it for client backward compatibility. To exclude function information in accordance with the JDBC 4.2 standard, set this property value to false.

CLIENT_TYPE

The CLIENT_TYPE of an Altibase 7.3 JDBC session is NEW_JDBC42.

Changes in SQL Results and Execution Plans

- Improved SQL Performance When Using the ORDER BY Clause in Subquery Inline Views

  This change has an impact on the execution plan of SQLs that utilize it, with the SORT plan node being eliminated within the SUBQUERY FILTER.
- Optimization of Nested LEFT OUTER JOIN Operations

This change can result in alterations to the execution plan and potential differences in the SQL execution results for SQLs affected by it.

- Modifications and Additions to Subquery Unnesting Functionality

  This change can lead to alterations in the execution plans of SQLs influenced by it.

Replication Restrictions

Replication Restrictions between Altibase 7.1 and Altibase 7.3

Because the replication protocol version has been changed, DDL synchronization are not supported between Altibase 7.1 and Altibase 7.3.

Because the database binary version has been changed, offline replication are not supported between Altibase 7.1 and Altibase 7.3.

Replication Restrictions between Altibase 6.5.1 and Altibase 7.3

Replication from Altibase 7.3 to Altibase 6.5.1 may fail when the target table contains data with SRID values.

2.2.4 Properties

The following properties have been added, changed, and deleted in Altibase 7.3.0.01. For more information on each property, please refer to the *General Reference*.

New Properties

- DISK_INDEX_BUILD_SORT_AREA_SIZE
- DBLINK_GLOBAL_TRANSACTION_LEVEL
- IB_CONCHKSPIN
- IB_ENABLE
- IB_LATENCY
- IB_LISTENER_DISABLE
- IB_MAX_LISTEN
- IB_PORT_NO
- INIT_TOTAL_WA_SIZE
- IPCDA_SEM_KEY
- IPCDA_SHM_KEY
- IPC_SHM_KEY
- IPC_SEM_KEY
- JOB_MSGLOG_COUNT
- JOB_MSGLOG_FILE
- JOB_MSGLOG_FLAG
- JOB_MSGLOG_SIZE
- LISTAGG_PRECISION

- MATHEMATICS_TEMP_MEMORY_MAXIMUM
- NETWORK_ERROR_LOG_FILE
- PSM_MAX_DDL_REFERENCE_DEPTH
- REGEXP_MODE
- REPLICATION_DDL_SYNC
- REPLICATION_DDL_SYNC_TIMEOUT
- REPLICATION_GAP_UNIT
- REPLICATION_IB_LATENCY
- REPLICATION_IB_PORT_NO
- REPLICATION_META_ITEM_COUNT_DIFF_ENABLE
- REPLICATION_RECEIVER_APPLIER_YIELD_COUNT
- REPLICATION_SENDER_IP
- SERIAL_EXECUTE_MODE
- SERVICE_THREAD_RECV_TIMEOUT
- SSL_CIPHER_SUITES
- SSL_LOAD_CONFIG
- ST_MSGLOG_COUNT
- ST_MSGLOG_FILE
- ST_MSGLOG_FLAG
- ST_MSGLOG_SIZE
- VARRAY_MEMORY_MAXIMUM

Modified Properties

- ARCHIVE_FULL_ACTION

  The property has been changed from Read-Only to Read-Write. While there is no change in the default value, a new setting value 2 has been added.

- CM_MSGLOG_FLAG

  Default value changed to 3.

- EXECUTE_STMT_MEMORY_MAXIMUM

  Default value changed from 1073741824 to 2147483648.

- HASH_AREA_SIZE

  Minimum changed from 512K to 3M.

- INDEX_INITRANS

  The maximum value was changed from 30 to 50.

- INDEX_MAXTRANS

  The default and maximum values were changed from 30 to 50.

- LOB_CACHE_THRESHOLD

  The maximum value was changed from 8192 to 524288.

- MEMORY_INDEX_BUILD_RUN_SIZE

Default value changed from 32768 to 131072.

- MM_MSGLOG_FILE

  The default value was changed to 1.

- PSM_CHAR_DEFAULT_PRECISION

  The default value was changed from 32767 to 32000.

- PSM_NCHAR_UTF16_DEFAULT_PRECISION

  Default value changed from 16383 to 16000.

- PSM_NCHAR_UTF8_DEFAULT_PRECISION

  The default value was changed from 10921 to 10666.

- PSM_NVARCHAR_UTF16_DEFAULT_PRECISION

  Default value changed from 16383 to 16000.

- PSM_NVARCHAR_UTF8_DEFAULT_PRECISION

  The default value was changed from 10921 to 10666.

- PSM_VARCHAR_DEFAULT_PRECISION

  The default value was changed from 32767 to 32000.

- REPLICATION_EAGER_PARALLEL_FACTOR

  The minimum value was changed from 1 to 2.

- SERVER_MSGLOG_FLAG

  Default value changed from 7 to 15.

- TOTAL_WA_SIZE

  The minimum value was changed to 0.

- TRANSACTION_SEGMENT_COUNT

  The maximum value was changed from 512 to 16384.

Removed Properties

- GLOBAL_TRANSACTION_LEVEL
- LOCK_MGR_TYPE
- LOCK_MGR_SPIN_COUNT
- LOCK_MGR_MIN_SLEEP
- LOCK_MGR_MAX_SLEEP
- LOCK_MGR_DETECTDEADLOCK_INTERVAL
- TEMP_MAX_PAGE_COUNT
- TRANSACTION_START_MODE

2.2.5 Meta Tables

New Meta Tables

- SYS_GEOMETRIES_
- SYS_GEOMETRY_COLUMNS_
- SYS_REPL_RECEIVER_

- SYS_REPL_TABLE_OID_IN_USE_
- USER_SRS_

Modified Meta Tables

- SYS_REPLICATIONS_

  New columns have been introduced below.

  - REMOTE_LAST_DDL_XSN
- SYS_REPL_HOSTS_

  New columns have been introduced below.

  - CONN_TYPE
  - IB_LATENCY
- SYS_REPL_OLD_COLUMNS_

  New columns have been introduced below.

  - MT_SRID
- SYS_REPL_OLD_ITEMS_

  New columns have been introduced below.

  - REMOTE_USER_NAME
  - REMOTE_TABLE_NAME
  - REMOTE_PARTITION_NAME
  - PARTITION_COUNT
  - PARTITION_METHOD
  - PARTITION_ORDER
  - PARTITION_MIN_VALUE
  - PARTITION_MAX_VALUE
  - INVALID_MAX_SN

Removed Meta Tables

The following meta tables have been removed.

- STO_COLUMNS_
- STO_DATUMS_
- STO_ELLIPSOIDS_
- STO_GEOCCS_
- STO_GEOGCS_
- STO_PRIMEMS_
- STO_PROJCS_
- STO_PROJECTIONS_
- STO_SRS_
- STO_USER_COLUMNS_

2.2.6 Performance Views

The following performance views have been added.

For more informaiton on each performance view, please refer to the **General Reference-2.The Data Dictionary**.

New Performance Views

- V$LIBRARY
- V$PROCINFO
- V$QUEUE_DELETE_OFF
- V$REPL_REMOTE_META_CHECKS
- V$REPL_REMOTE_META_COLUMNS
- V$REPL_REMOTE_META_INDEX_COLUMNS
- V$REPL_REMOTE_META_INDICES
- V$REPL_REMOTE_META_ITEMS
- V$REPL_REMOTE_META_REPLICATIONS

Removed Performance Views

- V$ST_ANGULAR_UNIT
- V$ST_AREA_UNIT
- V$ST_LINEAR_UNIT

2.3 Packages

OS	CPU	File Names
AIX	PowerPC	altibase- server-7.3.0.0.1-AIX-POWERPC-64bit-release.run
		altibase- client-7.3.0.0.1-AIX-POWERPC-64bit-release.run
HP-UX	IA64	altibase- server-7.3.0.0.1-HPUX-IA64-64bit-release.run
		altibase- client-7.3.0.0.1-HPUX-IA64-64bit-release.run
LINUX	x86-64	altibase-server-7.3.0.0.1-LINUX-X86-64bit-release.run
		altibase-client-7.3.0.0.1-LINUX-X86-64bit-release.run
LINUX	PowerPC	altibase-server-7.3.0.0.1-LINUX-POWERPC-64bit-release.run
		altibase-client-7.3.0.0.1-LINUX-POWERPC-64bit-release.run

OS	CPU	File Names
LINUX	PowerPCLE (Little Endian)	altibase-server-7.3.0.0.1-LINUX-POWERPCLE-64bit-release.run
		altibase-client-7.3.0.0.1-LINUX-POWERPCLE-64bit-release.run

2.4 Download

Package

http://support.altibase.com

Manual

https://github.com/ALTIBASE/Documents/blob/master/Manuals/Altibase_7.3/eng/README.md

Installation

https://github.com/ALTIBASE/Documents/blob/master/Manuals/Altibase_7.3/eng/Installation%20Guide.md

---

[1] Shapefile: A file format developed by ESRI, a software developer specializing in Geographic Information Systems (GIS). In the GIS field, the Shapefile is considered a standard file format and consists of the following three essential files.

- shp : contains spatial data information representing points, lines, and shapes in a vector format.
- shx : Index file, which contains the location of the shape information stored in the shp file.
- dbf : dBASE table file containing attribute information about the shape information in the shp file.

References : Geoprocessing considerations for shapefile output