

Altibase® Tools & Utilities

Adapter for Oracle User's Manual

Release 7.1 (September 18, 2017)



Altibase® Tools & Utilities Adapter for Oracle User's Manual
Release 7.1
Copyright © 2001~2017 Altibase Corp. All rights reserved.

This manual contains proprietary information of Altibase Corporation; it is provided under a license agreement containing restrictions on use and disclosure and is also protected by copyright patent and other intellectual property law. Reverse engineering of the software is prohibited.

All trademarks, registered or otherwise, are the property of their respective owners.

Altibase Corp.
10F, Daerung PostTower II,
306, Digital-ro, Guro-gu, Seoul 08378, Korea
Telephone: +82-2-2082-1000 Fax: 82-2-2082-1099
Homepage: <http://www.altibase.com>

Contents

Preface	5
About This Manual	6
Audience	6
Software Environment.....	6
Organization.....	6
Documentation Conventions.....	6
Related Documents.....	9
Online Manual.....	9
Altibase Welcomes Your Opinions.....	9
1. Introduction	11
1.1 About the Adapter for Oracle.....	12
1.1.1 Structure and Concepts.....	12
1.1.2 Terminology.....	13
2. Installation and Configuration	15
2.1 Pre-installation Tasks.....	16
2.1.1 OS	16
2.1.2 Database Versions.....	16
2.1.3 Database Character Sets	16
2.1.4 National Character Set.....	16
2.1.5 Interface	16
2.2 Installation	17
2.2.1 Installing oraAdapter	17
2.3 Post-installation Tasks.....	20
2.3.1 Setting Environment Variables	20
2.3.2 Checking the Installation Directory.....	20
2.4 Configuration	21
2.4.1 ORA_ADAPTER_HOME.....	21
2.4.2 Library_Path	21
2.4.3 NLS_LANG.....	21
2.5 Properties.....	23

2.5.1 Property Constraints	23
2.5.2 Properties for ALA.....	23
2.5.3 Properties for Checking Constraints.....	25
2.5.4 Properties for OCI	25
2.5.5 DML-Related Properties	27
3. Managing Adapter for Oracle	29
3.1 Properties for Checking Constraints	30
3.1.1 oraAdapter Constraints.....	30
3.1.2 Prequisites.....	30
3.1.3 Data Constraints.....	30
3.1.4 Connection Constraints	30
3.1.5 Allowable DDL Statements.....	31
3.1.6 Allowable DDL Statements on Target Replication Tables	31
3.2 Startup and Shutdown.....	32
3.2.1 Startup	32
3.2.2 Shutdown	33
3.3 Data Types	34
3.3.1 Examples.....	34
3.4 Adapter for Oracle Utility.....	36
3.4.1 oaUtility Start.....	36
3.4.2 oaUtility Stop	36
3.4.3 oaUtility Status.....	36
3.4.4 oaUtility Check.....	37
3.5 Command-Line Options.....	38
3.5.1 Syntax.....	38
3.5.2 Description.....	38
3.5.3 Example	38
3.6 FAQ.....	39
3.6.1 What do I have to do after I change an environment variable or a property?	39
3.6.2 What if the data are not successfully applied to the Oracle DB?	39

Preface

About This Manual

This manual explains the Adapter for Oracle utility, which applies data that has been changed in Altibase to an Oracle database.

Audience

This manual has been prepared for the following Altibase users:

- Database administrators
- Application developers
- Technical support workers

It is recommended that those reading this manual possess the following background knowledge:

- Basic knowledge in the use of computers, operating systems, and operating system utilities
- Experience in using relational databases and an understanding of database concepts
- Computer programming experience
- Experience in database server, operating system or network administration

Software Environment

This manual has been written assuming that Altibase 7 will be used as the database server.

Organization

This manual has been organized as follows:

- [Chapter1: Introduction](#)

This chapter describes the concept of the Adapter for Oracle and how data that has been changed in Altibase is written to an Oracle database.

- [Chapter2: Installation and Configuration](#)

This chapter explains how to install and configure the Adapter for Oracle.

- [Chapter3: Managing Adapter for Oracle](#)

This chapter explains how to manage the Adapter for Oracle.

Documentation Conventions

This section describes the conventions used in this manual. Understanding these conventions

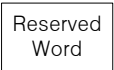




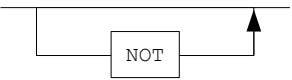
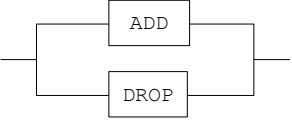
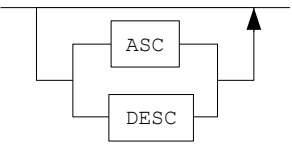
will make it easier to find information in this manual and other manuals in the series.

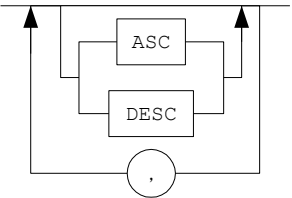
There are two sets of conventions:

- Syntax diagram conventions
- Sample code conventions

Syntax Diagram Conventions

This manual describes command syntax using diagrams composed of the following elements:

Elements	Meaning
	Indicates the start of a command. If a syntactic element starts with an arrow, it is not a complete command.
	Indicates that the command continues to the next line. If a syntactic element ends with this symbol, it is not a complete command.
	Indicates that the command continues from the previous line. If a syntactic element starts with this symbol, it is not a complete command.
	Indicates the end of a statement.
	Indicates a mandatory element.
	Indicates an optional element.
	Indicates a mandatory element comprised of options. One, and only one, option must be specified.
	Indicates an optional element comprised of options.

Elements	Meaning
	<p>Indicates an optional element in which multiple elements may be specified. A comma must precede all but the first element.</p>

Sample Code Conventions

The code examples explain SQL statements, stored procedures, iSQL statements, and other command line syntax.

The following table describes the printing conventions used in the code examples.

Rules	Meaning	Example
[]	Indicates an optional item.	VARCHAR [(size)] [[FIXED] VARIABLE]
{ }	Indicates a mandatory field for which one or more items must be selected.	{ ENABLE DISABLE COMPILE }
	A delimiter between optional or mandatory arguments.	{ ENABLE DISABLE COMPILE } [ENABLE DISABLE COMPILE]
.	Indicates that the previous argument is repeated, or that sample code has been omitted.	iSQL> select e_lastname from employees; E_LASTNAME ----- Moon Davenport Kobain 20 rows selected.
Other Symbols	Symbols other than those shown above are part of the actual code.	EXEC :p1 := 1; acc NUMBER(11,2);
Italics	Statement elements in italics indicate variables and special values specified by the user.	SELECT * FROM table_name; CONNECT <i>userID/password</i> ;

Rules	Meaning	Example
Lower case words	Indicate program elements set by the user, such as table names, column names, file names, etc.	SELECT e_lastname FROM employees;
Upper case words	Keywords and all elements provided by the system appear in upper case.	DESC SYSTEM_.SYS_INDEX_;

Related Documents

For more detailed information, please refer to the following documents:

- Administrator's Manual
- Error Message Reference
- Log Analyzer User's Manual
- Replication Manual
- Utilities Manual

Online Manual

Online versions of our manuals (PDF or HTML) are available from Altibase's Customer Support site (<http://altibase.com/support-center/>).

Altibase Welcomes Your Opinions

Please feel free to send us your comments and suggestions regarding this manual. Your comments and suggestions are important to us, and may be used to improve future versions of the manual. When you send your feedback, please make sure to include the following information:

- The name and version of the manual you are using
- Your comments and suggestions regarding the manual
- Your full name, address, and phone number

In addition to suggestions, this address may also be used to report any errors or omissions discovered in the manual, which we will address promptly.

If you need immediate assistance with technical issues, please contact Altibase's Customer Support site (<http://altibase.com/support-center/>).

We always appreciate your comments and suggestions.

1. Introduction

This chapter describes the concept of the Adapter for Oracle and how data that has been changed in Altibase is written to an Oracle database.

- [About the Adapter for Oracle](#)

1.1 About the Adapter for Oracle

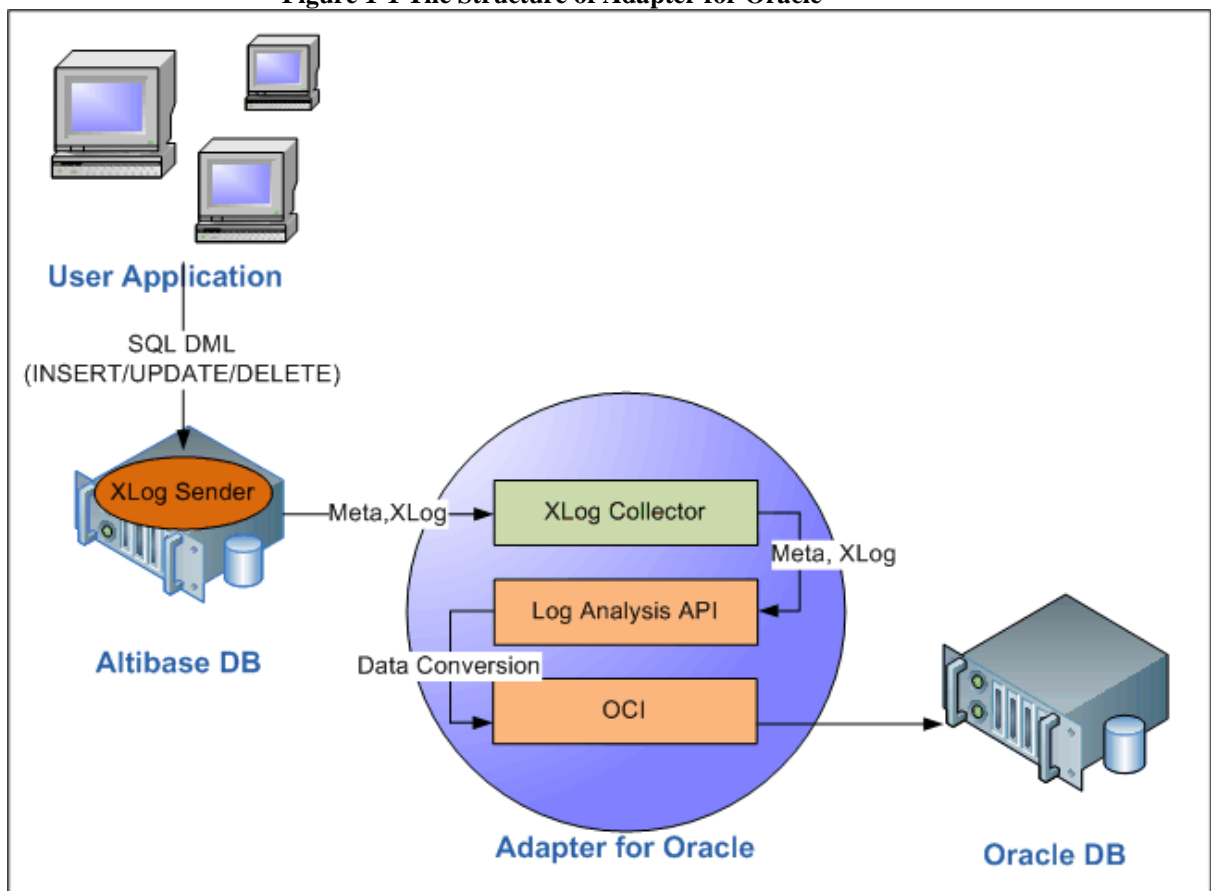
Altibase's Adapter for Oracle (referred to as "oraAdapter") is a utility that enables data that has been modified in Altibase to be applied in an Oracle database.

1.1.1 Structure and Concepts

To copy data that has been modified in Altibase to an Oracle DB, the user first needs to install Altibase, oraAdapter, and Oracle as shown in the following picture. The Altibase Log Analysis API (referred to as "ALA"), and the Oracle Call Interface (referred to as "OCI") are integrated in oraAdapter. The ALA utility receives data that has been modified in Altibase, whereas OCI enables the data to be written to an Oracle database. For more detailed information about ALA, please refer to the *Log Analyzer User's Manual*.

The following picture illustrates how oraAdapter is used to apply data from Altibase to an Oracle database.

Figure 1-1 The Structure of Adapter for Oracle



1. When the user creates or updates data, an XLog Sender that exists within Altibase

creates XLogs and meta information, and sends them to the XLog Collector. Note that meta information is exchanged only when handshaking takes place.

2. The XLog Collector, which exists within oraAdapter, uses ALA to provide XLogs and meta information to the user. If the call to ALA fails, a trace log will be written to the trc directory.
3. oraAdapter uses ALA to convert the acquired data so that the data can be used with an Oracle database.
4. The converted data is sent to the Oracle database using OCI.

1.1.2 Terminology

1.1.2.1 Handshaking

Handshaking is the task of checking the protocol version and meta data before the XLog Sender sends XLogs to the XLog Collector.

1.1.2.2 Log Analysis API

The Log Analysis API provides XLogs and meta data that are used to interpret the XLogs.

1.1.2.3 OCI

The Oracle Call Interface (OCI) is a set of C-language APIs that provide an interface with an Oracle database.

1.1.2.4 XLog

An XLog is a logical log that is converted from a physical log. It stores the history of transactions involving DML (INSERT/UPDATE/DELETE) statements.

1.1.2.5 XLog Collector

The XLog Collector is the module that receives meta data and XLogs from the XLog Sender. The XLog Collector contains meta data, an XLog queue, a transaction table, and an XLog pool.

1.1.2.6 XLog Pool

The XLog Pool is memory that has been allocated for the storage of XLogs.

1.1.2.7 XLog Sender

The XLog Sender is the module that analyzes active logs to create XLogs and then passes

them on to the XLog Collector.

The XLog Sender actively performs handshaking and XLog transmission.

2. Installation and Configuration

This chapter explains how to install and configure the Adapter for Oracle.

- [Pre-installation Tasks](#)
- [Installation](#)
- [Post-installation Tasks](#)
- [Configuration](#)
- [Properties](#)

2.1 Pre-installation Tasks

The following system requirements must be met to install and run Adapter for Oracle (oraAdapter). It is also necessary to make some environment settings for the system to ensure that oraAdapter runs properly.

For detailed information about system requirements, please contact Altibase's Customer Support site (<http://altibase.com/support-center/>).

2.1.1 OS

- AIX 5.3
- AIX 6.1
- LINUX

2.1.2 Database Versions

Altibase: Version 5.5.1 or above

Oracle Database: Version 10g or higher (must be compatible with OCI). Refer to the Library_Path chapter for more details.

2.1.3 Database Character Sets

This is the default character set for saving data. Setting Altibase and Oracle to use the same database character set can reduce the cost for data conversion; the user is recommended to do so.

2.1.4 National Character Set

Data in a language that is not supported by the database character set can be saved using the NCHAR and NVARCHAR types. Just as with the database character set, setting Altibase and Oracle to use the same national character set can reduce the cost for data conversion; the user is recommended to do so.

2.1.5 Interface

Before installing oraAdapter, it is necessary to confirm that the Oracle Call Interface (referred to as "OCI") has been installed. For more information about installing the OCI, please refer to the relevant Oracle documentation.

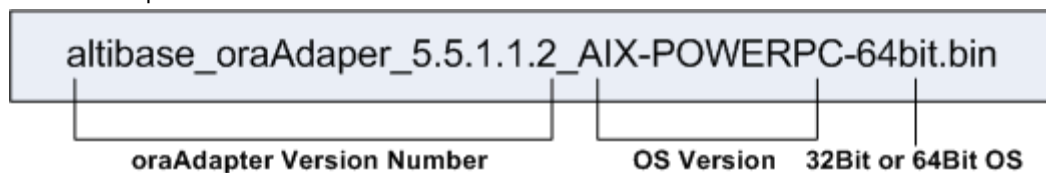
2.2 Installation

The oraAdapter installer can be executed in GUI mode if suitable display settings have been made. The description of oraAdapter installation in this manual will assume that the installation is being conducted in GUI mode. It is also possible to install oraAdapter in text mode if the appropriate display settings have not been made.

2.2.1 Installing oraAdapter

1. The first task is to obtain the version of the oraAdapter installer that is correct for the system on which it is to be run. The picture shows the naming convention for the oraAdapter installer.

The oraAdapter version must be the same as the version of Altibase with which it will be run.



2. When the installer is executed, the dialog appears. Verify that the version of oraAdapter to be installed is the correct version and click "Next".
3. The next dialog is for choosing the directory in which oraAdapter is to be installed. By default, a directory called "oraAdapter" will be created within the home directory of the user performing the installation.
4. To use oraAdapter, the Altibase Log Analyzer property settings must be made as follows. For more detailed information on properties, please refer to the section entitled "Properties" later in this chapter.
 - ALA_LOGGING_ACTIVE: This setting determines whether ALA will output trace logs. The default value is 1, which means that trace logs will be output.
 - ALA_RECEIVER_PORT: This is the number of the port on which oraAdapter listens. It can be set within the range from 1024 to 65535.
 - ALA_REPLICATION_NAME: This is the name of a replication object that exists in Altibase.
 - ALA_SENDER_IP: This is the IP address of the server on which Altibase is installed. It is set to 127.0.0.1 by default, assuming that Altibase and oraAdapter are operating on the same server.
 - ALA_SOCKET_TYPE: This is used to set the communication protocol that is used by ALA. oraAdapter supports the use of TCP/IP and the UNIX Domain Socket protocols.
 - ALA_XLOG_POOL_SIZE: This is used to set the maximum size of the XLog pool. The default is 10,000 XLogs.
5. In the next dialog, appropriate Altibase property settings must be made. For more detailed information on properties, please refer to the section entitled "Properties for

Checking Constraints" later in this chapter.

- ALTIBASE_IP: This is the IP address of the server on which Altibase is installed. It is set to 127.0.0.1 by default, assuming that Altibase and oraAdapter are operating on the same server.
 - ALTIBASE_PASSWORD: This is the password corresponding to the user account with which Altibase will be accessed.
 - ALTIBASE_PORT: This is the number of the port at which Altibase listens. It can be set within the range from 1024 to 65535.
 - ALTIBASE_USER : This is the name of the user account with which to access Altibase.
6. To use oraAdapter, appropriate Oracle DB property settings must be made. For more detailed information on properties, please refer to the section entitled "[Properties for OCI](#)" later in this chapter.
 - ORACLE_PASSWORD: This is the password corresponding to the user account with which to access the Oracle DB.
 - ORACLE_SERVER_ALIAS : This is used to specify an alias for accessing an Oracle DB that is already set in an Oracle Client. If this is not specified, access will be made to the Oracle DB that is set as the default host.
 - ORACLE_SKIP_DELETE : If you choose "YES", DELETE statements executed on Altibase will not be implemented on the Oracle DB.
 - ORACLE_SKIP_INSERT : If you choose "YES", INSERT statements executed on Altibase will not be implemented on the Oracle DB.
 - ORACLE_SKIP_UPDATE: If you choose "YES", UPDATE statements executed on Altibase will not be implemented on the Oracle DB.
 - ORACLE_USER: This is the name of a user account with which to access the Oracle DB.
 7. In the next two dialogs, appropriate Oracle DB property settings must be made. For more detailed information on these properties, please refer to the section entitled "[Properties for OCI](#)" later in this chapter.
 - ORACLE_ARRAY_DML_MAX_SIZE: "Array DML" means grouping multiple DML statements. This property sets the maximum number of DML statements that are grouped in this way. To disable Array DML, set this property to 1.
 - ORACLE_ASYNCHRONOUS_COMMIT: If you choose "YES", "asynchronous commit" will be used on the Oracle DB.
 - ORACLE_GROUP_COMMIT: If you choose "YES", "group commit" will be used on the Oracle DB.
 - ORACLE_UPDATE_STATEMENT_CACHE_SIZE: This property is used to set the size of the cache in which prepared UPDATE statements are stored. If this property is set to 0, oraAdapter does not cache UPDATE statements.
 8. Once all of the property settings pertaining to the use of oraAdapter have been made, they can be verified in the confirmation dialog box. Verify that all of the properties have been correctly set and click "Next".
 9. After you have verified the property settings in the confirmation dialog box, you are

ready to install oraAdapter. Click "Next" to continue.

10. While oraAdapter is being installed, the following two environment variables are set. Note that in order for the system to make use of the two new environment variables, it will be necessary to log out and log back in.
 - ORA_ADAPTER_HOME: this environment variable is added with the oraAdapter home directory that was specified earlier during the installation process
 - PATH: this environment variable is added with the value of "ORA_ADAPTER_HOME/bin".
11. The dialog box appears after oraAdapter has been successfully installed.

2.3 Post-installation Tasks

2.3.1 Setting Environment Variables

After oraAdapter has been installed, it is necessary to set environment variables, add a library path, and set the database and national character sets.

For more detailed information on setting environment variables, please refer to the section entitled Configuration later in this chapter.

- ORA_ADAPTER_HOME

This is set automatically when oraAdapter is installed.

- Add a Library Path

Add the path to the OCI library. In AIX, the OCI library path can be added as below:

```
$ export LIBPATH=$SIBPATH:$ORACLE_HOME/LIB
```

- NLS_LANG

This is used to indicate the character set that is being used in Altibase.

2.3.2 Checking the Installation Directory

After oraAdapter installation is complete, verify that the bin, conf, msg, and trc directories have been created in the \$ORA_ADAPTER_HOME directory. The structure and role of each directory is as follows:

- bin directory

This directory contains the oraAdapter (Adapter for Oracle) and oaUtility (Adapter for Oracle Utility) executable files and some internally used files.

- conf directory

The oraAdapter.conf file, in which the oraAdapter property settings are stored, is located in this directory.

- msg directory

If any errors occur while oraAdapter is running, the messages in this directory will be used to write trace logs.

- trc directory

The oraAdapter writes trace logs to files located in this directory.

2.4 Configuration

In order to use oraAdapter, it is necessary to configure environment variables.

2.4.1 ORA_ADAPTER_HOME

This is the directory in which oraAdapter was installed. This environment variable is set automatically during oraAdapter installation.

2.4.2 Library_Path

oraAdapter uses an OCI library which is compatible with 10gOCI in order to apply modified data of ALTIBASE HDB to Oracle DB. Therefore, the path in which OCI library is located should be added into the environment variables of the user account which will run oraAdapter.

The environment variables in which a library is generally added is LD_LIBRARY_PATH, however, they vary depending on the OS which will be executing oraAdapter.

oraAdapter is supposed to dynamically link with the Oracle OCI 10g library(libclntsh.so.10.1). In order to use other version's Oracle OCI library, the version of OCI library should be linked with libclntsh.so.10.1.

For example, if Oracle 11g Instant Client is installed, libclntsh.so.11.1 should be linked with libclntsh.so.10.1 as follows.

```
$ cd $ORACLE_HOME/lib
$ ln -s libclntsh.so.11.1 libclntsh.so.10.1
```

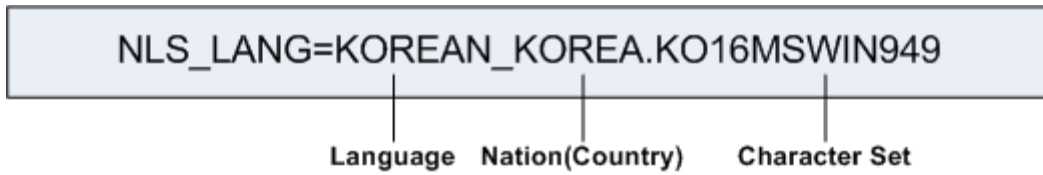
Refer to Oracle Call Interface for the compatibility between Oracle 10g OCI and other version's OCI library. Refer to the *Interface Manual* for the library compatibility between Oracle 10g OCI and other OCI version.

2.4.3 NLS_LANG

The NLS_LANG environment variable is used with the OCI. This environment variable is used to indicate the character set of the strings that are inputted to the OCI. The OCI refers to this environment variable when converting data retrieved from Altibase to the Oracle database character set. Because oraAdapter receives data from Altibase, the NLS_LANG environment variable must be set to the Oracle DB character set that corresponds to the Altibase character set.

Note that if the character set of the Oracle DB is not the same as that of Altibase, performance will suffer as a result of the data conversion task performed by the OCI.

The syntax of the NLS_LANG environment variable is as follows.



The following table shows the character sets that are supported for use with Altibase and the corresponding Oracle DB character sets and oraAdapter NLS_LANG environment variable settings.

Table 2-1 Database Character Set

Altibase	NLS_LANG	Oracle DB
BIG5	.ZHT16BIG5	ZHT16BIG5
EUC-JP	.JA16EUC	JA16EUC
GB231280	.ZHS16CGB231280	ZHS16CGB231280
KO16KSC5601	.KO16KSC5601	KO16KSC5601
MS932		JA16SJIS
MS936		ZHS16GBK
MS949	.KO16MSWIN949	KO16MSWIN949
SHIFT-JIS	.JA16SJIS	JA16SJIS
US7ASCII	.US7ASCII	US7ASCII
UTF-8	.UTF8	UTF8

Table 2-2 National Character Set

Altibase	NLS_LANG	Oracle DB
UTF-8	.UTF8	UTF8
UTF-16	.AL16UTF16	AL16UTF16

2.5 Properties

The properties that are set within oraAdapter are used by the Altibase Log Analyzer and Oracle Call Interface, and are also necessary for making the settings that determine how oraAdapter will be executed.

The property file is oraAdapter.conf, which is located in the \$ORA_ADAPTER_HOME/conf/ directory.

The properties that are used within oraAdapter can be classified as follows:

- [Properties for ALA](#)
- [Properties for Checking Constraints](#)
- [Properties for OCI](#)
- [DML-Related Properties](#)

2.5.1 Property Constraints

Space or tab cannot be used when setting the property value. Also, quotation marks should be used in order to use letters including special characters.

2.5.2 Properties for ALA

The following properties must be set in order for oraAdapter to be able to use the Altibase Log Analyzer. For more detailed information about ALA properties, please refer to the *Log Analyzer User's Manual*.

2.5.2.1 ALA_LOGGING_ACTIVE

This is used to set whether the Altibase Log Analyzer will output Trace Logs.

- 0 : Trace logs are not output
- 1 : Trace logs are output (Default Value)

2.5.2.2 ALA_RECEIVER_PORT

This is the port number at which the XLog Collector will listen to receive XLogs. Set it to the port number at which oraAdapter is listening.

- Range : 1024~65535

2.5.2.3 ALA_RECEIVER_XLOG_TIMEOUT

This property specifies the standby time for XLog collector to receive XLogs.

- Default Value : 10
- Range : 1 – 4294967295

2.5.2.4 ALA_REPLICATION_NAME

This is the name of the replication object used as the XLog Sender. It is set to the name of a replication object created within Altibase.

2.5.2.5 ALA_SOCKET_TYPE

This is used to set the protocol that is used by the Altibase Log Analyzer. Adapter for Oracle supports TCP/IP and the Unix Domain Protocol. However, in order to use a UNIX domain socket, Altibase and Adapter for Oracle must be located on the same server.

- TCP : TCP/IP will be used(Default)
- UNIX : the Unix Domain Protocol will be used

2.5.2.6 ALA_SENDER_IP

This is the IP of the XLog Sender. Set it to the IP of the server on which Altibase is installed.

- Default Value : 127.0.0.1

2.5.2.7 ALA_XLOG_POOL_SIZE (Unit : Number)

This is the number of XLogs that oraAdapter can allocate.

The record modifications are accumulated with XLogs before a transaction of the original DB is committed in oraAdapter. If a transaction taking place changes multiple records, the oraAdapter cannot normally replicate due to the insufficient XLogs. Therefore, the value of this property should be adjusted in accordance with the type of transaction in the original DB. If ALA Sender performs sync task in the original DB, a commit is implemented for the number of times specified in REPLICATION_SYNC_TUPLE_COUNT property. Therefore, if the value of ALA_XLOG_POOL_SIZE property is smaller than that of the REPLICATION_SYNC_TUPLE_COUNT, sync task cannot be proceeded due to the insufficient XLogs; thus, the property value should be set to a large value.

- Default Value : 10,000
- Range : 1~2147483647

2.5.3 Properties for Checking Constraints

2.5.3.1 ALTIBASE_IP

This is the IP address of the server on which Altibase is installed.

- Default Value : 127.0.0.1

2.5.3.2 ALTIBASE_PASSWORD

This is used to specify the password for the user account through which the connection to Altibase will be established.

2.5.3.3 ALTIBASE_PORT

This is the number of the port at which Altibase listens.

- Range : 1024 ~ 65535

2.5.3.4 ALTIBASE_USER

This is used to specify the name of a user account with which to connect to Altibase.

2.5.4 Properties for OCI

The following properties must be set in order for Adapter for Oracle to be able to use the Oracle Call Interface (OCI).

2.5.4.1 ORACLE_ARRAY_DML_MAX_SIZE

"Array DML" means grouping multiple DML statements of the same kind. This realizes a performance improvement by reducing network costs.

- Default Value: 10
- Range: 1~32767

This property sets the maximum number of DML statements that are grouped in this way.

- It improves performance when the ORACLE_GROUP_COMMIT property is enabled.
- At present, this property only affects the INSERT and DELETE statements.
- To disable Array DML, set this property to 1.

2.5.4.2 ORACLE_ASYNCCHRONOUS_COMMIT

This property determines whether or not to wait until commit logs have been written to disk.

- 0: This setting ensures that a commit message (i.e. a message indicating that the transaction has been committed) is returned to the client only after a commit log has been written to a persistent online redo log.
- 1: At this setting, a commit message is returned to the client regardless of whether the commit log has been completely written to disk. This is so-called "asynchronous commit", which can improve performance. Note that using asynchronous commit compromises durability in the interests of speed. Therefore, at this setting, when the Oracle database crashes, it may be necessary to synchronize the Oracle database with the Altibase database.

2.5.4.3 ORACLE_GROUP_COMMIT

If this property is enabled, the logs for multiple transactions that are occurring at the same time are accumulated in a buffer. After a certain amount of logs has amassed in the buffer, they are written to disk all at once. This is so-called "group commit", which allows redo information for multiple transactions to be written to disk in a single I/O operation.

Group Commit is suitable for environments in which transactions are frequently committed. The resultant improvement in performance comes at the cost of increased response times for individual transactions. It also introduces a problem whereby, if one of the transactions in the group cannot be committed, none of the transactions in the group are committed.

- 0: Group commit is not used.
- 1: Group commit is used.

2.5.4.4 ORACLE_PASSWORD

This is used to specify the password for the user account through which the connection to the Oracle DB will be established.

2.5.4.5 ORACLE_SERVER_ALIAS

This property is used to specify an alias for an Oracle DB server that is set in the tnsnames.ora file, which contains information about servers to which Oracle Clients connect. If this is not specified, connection will be made to the Oracle DB that is set as the default host.

If, for example, the tnsname.ora file contains the information shown below, ORACLE_SERVER_ALIAS should be set to orcl10g.

```
orcl10g =  
(DESCRIPTION =
```

```
(ADDRESS = (PROTOCOL = TCP)(HOST = server1.us.oracle.com)(PORT = 1521))
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = orcl)
)
)
```

2.5.4.6 ORACLE_UPDATE_STATEMENT_CACHE_SIZE

Statement caching improves performance by caching executable statements that are used repeatedly, such as in a loop or in a method that is called repeatedly.

- Default Value: 20
- Range: 0~4294967295 (unit: number of statements)
This property is used to set the size of the cache in which prepared UPDATE statements are stored.
- Because the INSERT and DELETE statements already have their own cache, only UPDATE statements are stored in this cache.
- If this property is set to 0, oraAdapter does not cache UPDATE statements.

2.5.4.7 ORACLE_USER

This is used to specify the name of a user account with which to connect to the Oracle DB.

2.5.5 DML-Related Properties

These properties are used to set whether DML statements executed in Altibase will also be executed in the Oracle DB.

2.5.5.1 ORACLE_SKIP_DELETE

This property determines whether DELETE statements executed in Altibase are also executed in the Oracle DB. If it is set to 1 ("skip"), DELETE statements are skipped, meaning that they are not executed on the Oracle DB.

- 0: Do not skip the statements; that is, execute the statements normally.
- 1: Skip the statements.

2.5.5.2 ORACLE_SKIP_INSERT

This property determines whether INSERT statements executed in Altibase are also executed in the Oracle DB. If it is set to 1 ("skip"), INSERT statements are skipped, meaning that they are not executed on the Oracle DB.

- 0: Do not skip the statements; that is, execute the statements normally.
- 1: Skip the statements.

2.5.5.3 ORACLE_SKIP_UPDATE

This property determines whether UPDATE statements executed in Altibase are also executed in the Oracle DB. If it is set to 1 ("skip"), UPDATE statements are skipped, meaning that they are not executed on the Oracle DB.

- 0: Do not skip the statements; that is, execute the statements normally.
- 1: Skip the statements.

3. Managing Adapter for Oracle

This chapter explains how to start up and shut down oraAdapter and how to use the Adapter for Oracle Utility.

3.1 Properties for Checking Constraints

3.1.1 oraAdapter Constraints

There are various constraints in using oraAdapter(or oraAdapter). If the following conditions are not satisfied, oraAdapter will not be able to use.

3.1.2 Prelquisites

- If a conflict occurs when executing INSEART/UPDATE/DELETE in the target database(Oracle or other altibase), the execution is cancelled and a message is left to an error log file or the execution is neglected in accordance with the configuration.
- An error which was occurred during the replication can be withdrawn. That is to say, if there is a duplicated data when inserting multiple data, the execution is completed except the duplicated data.
- The replication speed might be slower than that of the service.

3.1.3 Data Constraints

- If schemas of the original DB and target DB are not identical or if retrieving schemas of the replication target table fails, the command line option should be directly executed since oaAdapter cannot be executed with oaUtility.
- A primary key must exist in the table which will be replicated.
- There should be no modifications on the primary key which will be replicated.
- The tables in both servers which will be replicated ought to have a column type, sequence, a primary key, and NOT NULL constraint that are identical or compatible from each other.

3.1.4 Connection Constraints

The maximum number of possible replication connection for XLog Sender and replication per one ALTIBASE HDB database is determined by specified value of REPLICATION_MAX_COUNT property.

3.1.5 Allowable DDL Statements

In general, replication target tables cannot execute the data definition language(DDL). However, the followings statements are able to excute DDL regardless of XLog Sender. Refer to the *Executing DDL Statements on Replication Target Tables* of deploying replication section in the *Replication Manual* for other allowable DDL statements.

- ALTER INDEX REBUILD PARTITION
- GRANT OBJECT
- REVOKE OBJECT
- CREATE TRIGGER
- DROP TRIGGER

3.1.6 Allowable DDL Statements on Target Replication Tables

Generally, if DDL is executed in a replication target table, all modifications made before the current DDL are applied to the database, and then the databse is terminated. Once oraAdapter(altiAdapter) is terminated, the replication can be executed by re-starting oraAdapter(altiAdapter) after making table schemas identical through executing the same DDL on the target database.

Refer to the *Executing DDL Statements on Replication Target Tables* for other allowable DDL statements.

3.2 Startup and Shutdown

The following is an explanation of how to start up and shut down oraAdapter.

3.2.1 Startup

How to start up oraAdapter will now be explained step by step.

In order to use oraAdapter, Altibase and the Oracle DB must first both be running.

Additionally, verify that the Adapter for Oracle property settings and environment variables have been set appropriately for the environment in which oraAdapter will be run.

If any environment variables or property settings are changed after oraAdapter has been started, it will be necessary to restart oraAdapter in order to apply the changes. For more information on setting environment variables, please refer to Post-installation Tasks in Chapter Two.

1. Check whether the Altibase REPLICATION_PORT_NO property¹ has been set to a port number that is actually available for use with replication. If it is necessary to change this property setting, it will also be necessary to restart Altibase.
2. Before starting oraAdapter, it is also necessary to configure the XLog Sender so that the Altibase Log Analyzer (ALA) can be used. The XLog Sender is used to send XLogs and Meta information from Altibase.

In the following statement, an XLog Sender is created so that the data in table *t1*, which belongs to the sys user in Altibase, can be replicated to table *t2*, which belongs to the user *scott* in the Oracle DB.

```
CREATE REPLICATION ala FOR ANALYSIS WITH '127.0.0.1', 25090
FROM sys.t1 TO scott.t2;
```

CREATE REPLICATION ala FOR ANALYSIS WITH '127.0.0.1', 25090 FROM sys.t1 TO scott.t2;

ALA_REPLICATION_NAME

Address of oraAdapter

Table Name in Altibase DB

ALA_RECEIVER_PORT

Table Name in Oracle DB

3. Now it is time to start oraAdapter. oraAdapter can be started either by executing it directly, or using the Adapter for Oracle Utility. For detailed information on how to start oraAdapter using the Adapter for Oracle Utility, please refer to the Adapter for Oracle Utility section later in this chapter. In LINUX, the oraAdapter can be started using the command below:

¹REPLICATION_PORT_NO specifies the replication port number to use on a local server when the local server establishes a replication connection. For a complete description of this and other Altibase properties, please refer to the *General Reference*.


```
$ cd $ORA_ADAPTER_HOME/bin  
$ ./oraAdapter
```

4. Start the XLog Sender for the Altibase Log Analyzer. Any attempt to start the XLog Sender before oraAdapter has been started will fail.

```
iSQL> ALTER REPLICATION ala START;  
Alter success.
```

3.2.2 Shutdown

The process of shutting down oraAdapter includes the process of stopping the operation of the XLog Sender. If oraAdapter is shut down forcefully using the Adapter for Oracle Utility, it will be shut down successfully, but the Altibase XLog Sender will continue to attempt to connect to oraAdapter.

```
iSQL> ALTER REPLICATION ala STOP;  
Alter success.
```

3.3 Data Types

When data in Altibase are applied to an Oracle DB, the data types are converted as shown in the following table.

Table 3-1 Data Type Conversion

Altibase	Oracle DB	Additional Information for Oracle DB
BIGINT	NUMBER	
CHAR	CHAR	
DATE	DATE	
DOUBLE	NUMBER	BINARY_DOUBLE can also be used.
FLOAT	NUMBER	
INTEGER	NUMBER	
NUMERIC	NUMBER	
REAL	NUMBER	BINARY_FLOAT can also be used.
SMALLINT	NUMBER	
NCHAR	NCHAR	
NVARCHAR	NVARCHAR2	
VARCHAR	VARCHAR2	

3.3.1 Examples

The following example illustrates how the data types are converted when data in Altibase are applied to an Oracle DB, assuming that the National Character Set has been set to UTF-16.

Assuming that the data types in Altibase have been set as follows:

```
iSQL> CREATE TABLE T1(
          A1 INTEGER PRIMARY KEY,
          A2 CHAR(20),
          A3 VARCHAR(20),
          A4 NCHAR(20),
```

```
A5 NVARCHAR(20)
);
```

the data types in the Oracle DB would be set as shown below.

```
SQL> CREATE TABLE T1(
    A1 NUMBER PRIMARY KEY,
    A2 CHAR(20),
    A3 VARCHAR2(20),
    A4 NCHAR(20),
    A5 NVARCHAR2(20)
);
```

3.4 Adapter for Oracle Utility

Adapter for Oracle Utility (oaUtility) is a script that runs oraAdapter as a daemon and check its status. This script runs in a bash shell installed with GNU sed.

The Adapter for Oracle Utility supports the following options:

- [oaUtility Start](#)
- [oaUtility Stop](#)
- [oaUtility Status](#)
- [oaUtility Check](#)

3.4.1 oaUtility Start

3.4.1.1 Syntax

```
oaUtility {start}
```

3.4.1.2 Description

This is used to start oraAdapter as a daemon.

3.4.2 oaUtility Stop

3.4.2.1 Syntax

```
oaUtility {stop}
```

3.4.2.2 Description

This is used to forcefully terminate the Adapter for Oracle process.

When oraAdapter is forcefully shut down using this command, only Adapter for Oracle is shut down. The XLog Sender and Adapter for Oracle continually attempt to perform handshaking.

3.4.3 oaUtility Status

3.4.3.1 Syntax

```
oaUtility {status}
```

3.4.3.2 Description

This is used to check whether oraAdapter is running.

3.4.4 oaUtility Check

3.4.4.1 Syntax

```
oaUtility {check [alive|constraints]}
```

3.4.4.2 Description

If neither the alive nor constraints option is specified, oaUtility continually checks whether oraAdapter is running, and if it has been shut down (regardless of whether it was shut down normally or forcibly), restarts it.

If either the alive or constraints option is specified, the corresponding task is performed only once, and then oaUtility shuts down (i.e. does NOT perform continual checking).

Specifying the alive option tells oaUtility to check whether oraAdapter is running.

Specifying the constraints option tells oaUtility to check whether each of the primary keys in the tables to be ported from Altibase to Oracle DB are defined consistently, i.e. on the basis of columns having the same name.

3.5 Command-Line Options

The following command-line options are supported for use with oraAdapter.

3.5.1 Syntax

```
oraAdapter [-v | -version]
```

3.5.2 Description

Replication can be prepared by directly executing the binary of oraAdapter (altiAdapter). If this option is specified, it is available to know which Altibase version was compiled with oraAdapter(altiAdapter).

3.5.3 Example

```
./oraAdapter  
ALTIBASE Adapter started.
```

```
./oraAdapter -v  
ALTIBASE Adapter for Oracle version 5.5.1.1.2  
...
```

3.6 FAQ

3.6.1 What do I have to do after I change an environment variable or a property?

Once oraAdapter has been started, if any environment variables or properties are changed, it will be necessary to restart oraAdapter in order to implement the changes.

3.6.2 What if the data are not successfully applied to the Oracle DB?

If oraAdapter fails to write a data item to the Oracle DB, it merely writes a log message and proceeds to the next data item. This log message will be left in a trace log file in the '\$ORA_ADAPTER_HOME/trc' directory.

Index

A

Adapter for Oracle	11, 12
Adapter for Oracle Utility	36
Add a Library Path	20
ALA_LOGGING_ACTIVE	17, 23
ALA_RECEIVER_PORT	17, 23
ALA_RECEIVER_XLOG_TIMEOUT	24
ALA_REPLICATION_NAME	17, 24
ALA_SENDER_IP	17, 24
ALA_SOCKET_TYPE	17, 24
ALA_XLOG_POOL_SIZE	17, 24
ALI	12
ALTIBASE_IP	18, 25
ALTIBASE_PASSWORD	18, 25
ALTIBASE_PORT	18, 25
ALTIBASE_USER	18, 25

B

bin directory	20
---------------------	----

C

Command Line Options	36
Command-Line Options	38
conf directory	20
Configuration	21

D

Data Type Conversion	34
Database Character Set	22
Database Character Sets	16
Database Versions	16
DML-Related Properties	27

E

Environment Variables	20
-----------------------------	----

H

Handshake	13
Handshaking	13

I

Installing oraAdapter	17
Interface	16

L

Library_Path	21
Log Analysis API	13

M

msg directory	20
---------------------	----

N

naming convention for the oraAdapter installer	17
National Character Set	16, 22
NLS_LANG	20, 21

O

oaUtility	36
oaUtility Check	37
oaUtility Start	36
oaUtility Status	36
oaUtility Stop	36
OCI	12, 13
ORA_ADAPTER_HOME	19, 20, 21

oraAdapter	12
oraAdapter Constraints	30
ORACLE_ARRAY_DML_MAX_SIZE	18, 25
ORACLE_ASYNCHRONOUS_COMMIT	18, 26
ORACLE_GROUP_COMMIT	18, 26
ORACLE_PASSWORD	18, 26
ORACLE_SERVER_ALIAS	18, 26
ORACLE_SKIP_DELETE	18, 27
ORACLE_SKIP_INSERT	18, 27
ORACLE_SKIP_UPDATE	18, 28
ORACLE_UPDATE_STATEMENT_CACHE_SIZE ...	18, 27
ORACLE_USER	18, 27

P

Properties	23
Properties for ALA	23
Properties for OCI	25

Property Constraints	23
----------------------------	----

S

Shutdown	33
Startup	32

T

The Structure of Adapter for Oracle	12
trc directory	20

X

XLog	13
XLog Collector	13
XLog Pool	13
XLog Sender	13