

ADABAS D

Using the OCI Call Interface under Windows 95



Manual Order Number: ESD6110???

This document is applicable to ADABAS D Version 6.1.1 PE and to all subsequent releases, unless otherwise indicated in new editions or technical newsletters.

Specifications contained herein are subject to change and these changes will be reported in subsequent revisions or editions.

Readers' comments are welcomed. Comments may be addressed to the Documentation Department at the address on the back cover.

© April 14, 2025, SOFTWARE AG, Germany & SOFTWARE AG of North America, Inc.

All rights reserved

Printed in the Federal Republic of Germany

The SOFTWARE AG documentation often refers to numerous hardware and software products by their trade names. In most, if not all cases, these designations are claimed as trademarks or registered trademarks by their respective companies.

1The OCI Library

C applications which use the Oracle Call Interface (OCI) can work with ADABAS D without change. Just a link to the ADABAS OCI library is required. Differences between the OCI Interface and Oracle's OCI which may force changes to the source files are described in the chapter "Special Remarks". A description of the OCI entries can be found in the respective Oracle user manuals.

Linking The OCI Application

```
ocilnk <options> <main> <external objects or libs>
```

The filename of the main program <main> must be specified after the linker options. The executable program receives the name of the file (without suffix). All the other file parameters are also noted without suffix and must be object modules (*.obj) or libraries (*.lib) in any sequence.

Example:

```
ocilnk test fn1 fn2
```

There are test.obj, fn1.lib, fn2.lib.

The executable program gets the name test.

Executing The Linked OCI Application

Options are passed to the program in the environment variable SQLOPT.

```
SET SQLOPT=-X -d MyDatabase  
<fn>
```

Enter the filename to execute the linked program.

RuntimeOptions

```
cachelimit      :: -y <cache limit>  
isolation level :: -I <isolation level>  
nodate          :: -N  
profile         :: -R  
serverdb        :: -d <serverdb>  
servernode      :: -n <servernode>  
timeout         :: -t <timeout>  
trace filename  :: -F <tracefn>  
trace long      :: -X  
      short     :: -T  
      alt       :: -Y <statement count>  
      time      :: -L <seconds>  
user            :: -u <usern>,<passw>  
userkey         :: -U <userkey>
```

The Trace File

The trace file shows the executed OCI entries, including their actual parameters and information sent to or received from the interface to the ADABAS kernel. The SQL statements are only recorded for parse requests to the kernel. The parse handle (parseid) can be used to find out which SQL statement is currently executed.

The information contained in the trace file depends on the TRACE option.

For a simple trace (TRACE SHORT), only the sequence of the executed OCI entries – including their actual parameters, the SQL statements sent to the database kernel and the return code of the OCI entries – are recorded.

Example:

```
=====
=ORLON (00410668,004106a8,00400fdd,5,00400fe3,-1,0)
SESSION   :   1
SQLMODE   :   ORACLE
SERVERDB   :   ADB
SERVERNODE: adanode
CONNECT "DEMO" " IDENTIFIED BY :A SQLMODE ORACLE
=====
=OPEN (004107a8,00410668,00000000,-1,-1,00000000,-1)
=====
=OPEN (004107e8,00410668,00000000,-1,-1,00000000,-1)
=====
=OCOF (00410668)
=====
=OPARSE(004107a8,00401072,-1,1,2)
MDECLARE SQL_CUR00000000 CURSOR FOR SELECT NVL(MAX(empno),0)
FROM emp
PARSE: 000014DBD00000013D012d00
DESCRIBE
=====
=ODEFIN(004107a8,1,7ffffb24,4,3,-1,
00000000,00000000,-1,-1,00000000,00000000)
=====
=OEXFET(004107a8,1,0,0)
EXECUTE: 000014DBD00000013D012d00
RESULTTABLE: SQL_CUR00000000
ROWCOUNT: 0
MFETCH SQL_CUR00000000 INTO :A
PARSE: 000014DBD01000013D002b00
EXECUTE: 000014DBD01000013D002b00
ROWNO**** 1
ROWCOUNT: 1
```

If the detailed form of the trace file is specified (TRACE LONG), the input and output values of the SQL parameters involved and the execution time of the statements are included.

Example:

```
=OPEN (004107e8,00410668,00000000,-1,-1,00000000,-1)
=====
=OCOF (00410668)
=====
=OPARSE(004107a8,00401072,-1,1,2)
MDECLARE SQL_CUR00000000 CURSOR FOR SELECT NVL(MAX(empno),0)
FROM emp
PARSE: 000014DBD00000013D012d00
START : DATE : 1995-06-14 TIME : 0012:26:59
END : DATE : 1995-06-14 TIME : 0012:26:59
DESCRIBE
=====
=ODEFIN(004107a8,1,7ffffb24,4,3,-1,
00000000,00000000,-1,-1,00000000,00000000)
=====
=OEXFET(004107a8,1,0,0)
EXECUTE: 000014DBD00000013D012d00
RESULTTABLE: SQL_CUR00000000
ROWCOUNT: 0
START : DATE : 1995-06-14 TIME : 0012:26:59
END : DATE : 1995-06-14 TIME : 0012:26:59
MFETCH SQL_CUR00000000 INTO :A
PARSE: 000014DBD01000013D002b00
START : DATE : 1995-06-14 TIME : 0012:26:59
END : DATE : 1995-06-14 TIME : 0012:26:59
EXECUTE: 000014DBD01000013D002b00
ARR-CNT** 1
ROWNO**** 1
OUTPUT : 1: PARAMETER : 8294
ROWCOUNT: 1
START : DATE : 1995-06-14 TIME : 0012:26:59
END : DATE : 1995-06-14 TIME : 0012:26:59
```

The option TRACE ALT has the effect that the trace output is made alternately to two files. When doing so, as many executed OCI entries are recorded in each file as are specified in <statement count>. If there are more OCI entries to be executed than indicated by <statement count>, the files will be overwritten cyclically. The trace files are named 'OCITRAC.pct' and 'OCITRA2.pct'. The OCI entries are recorded together with the contents of the host variables (trace long). Date and time specifications cannot be suppressed.

If the trace output should not be given the default name, the option TRACE FILE can be used to specify another name. If no other trace option was specified, the option TRACE SHORT is simultaneously enabled as a default. The filename is standardized according to the operating system conventions in the platform-specific user manuals. The name may also be specified as a character string constant. Only trace files with default trace filenames are not buffered on output.

The option TRACE NO DATE/TIME can be used to suppress the output of the date and time values for the start and end of the execution of an OCI entry made into the trace file.

With the option TRACE TIME, only those OCI entries are recorded whose execution time is greater than or equal to <seconds>. These OCI entries are recorded together with the contents of the host variables (trace long). Date and time specifications cannot be suppressed.

Profiling

When the option PROFILE is enabled during an application's run, statistical data on the processed OCI entries may be obtained. To obtain results, however, the table SYSPROFILE (system table) of the LOCALSYSDBA must have been created when initializing the database. If this has not been done, the table has to be created first.

Definition of the table:

```
EXEC SQL
CREATE TABLE SYSPROFILE (
    USERNAME CHAR (18) KEY,
    PROGNAME CHAR (18) KEY,
    MODNAME CHAR (18) KEY,
    LANGUAGE CHAR (12) KEY,
    LINENO FIXED ( 7) KEY,
    PARSEID CHAR (12) BYTE KEY,
    STMBEGIN CHAR (40),
    RUNDATE DATE,
    RUNCOUNT FIXED (10),
    SECONDS FIXED (12,3) )
END-EXEC.
```

For every SQL statement, the beginning of statement, date of runtime, number of calls and accumulated realtime is entered into the table. The realtime consists of the time taken by the processing of a statement within an application program including all data conversions and time needed by the ADABAS kernel. The time needed to enter this information into the table SYSPROFILE, however, is not included. The key of a row consists of the following specifications: user name, program name, module name, language of the application program, and line number of the statement within the application program related to the source and the internal parseid. With the enabled TRACE option, the time required for writing the trace to a file affects the profiling. Therefore, it is not convenient to activate the PROFILE and TRACE options at the same time.

The entries to the SYSPROFILE table are made within the transactions of the application program. Therefore, they are only stored in the table when the application program issues a COMMIT WORK.

When the option is enabled, old entries made for username, program name, and language are always deleted at the beginning of the program. After the run, the table LOCALSYSDBA.SYSPROFILE may be looked up by means of QUERY and the stored data may be evaluated. The entries remain in the table until they are deleted explicitly or the program is restarted with the PROFILE option.

Special Remarks

The subprograms "sqlld2" and "oopt" have no effect.

"odessp" provides a description of the parameters of the named ADABAS Stored Procedure.

In certain situations, ADABAS return codes may be passed to the application instead of Oracle return codes.

2Index

- E -

executing, OCI application.....1-2

- L -

Linking, OCI.....1-1

- O -

OCI Library.....1-1

ocilnk.....1-1

- P -

PROFILE option.....1-7

- R -

runtime options, OCI.....1-2

- S -

SQLOPT.....1-2

system tables.....1-7

- T -

Trace File.....1-3

Table of Contents

1	The OCI Library.....
1.1.1	Linking The OCI Application
1.1.2	Executing The Linked OCI Application
1.1.3	Runtime Options.....
1.1.4	The Trace File.....
1.1.5	Profiling.....
1.1.6	Special Remarks.....
2	Index.....

