73
DBMS_DEBUG_JDWP

The DBMS_DEBUG_JDWP provides the interface to initiate and control the debugging of PL/SQL stored procedures and Java stored procedures over Java Debug Wire Protocol (JDWP).

This chapter contains the following topics:

DBMS_DEBUG_JDWP Overview

Oracle supports the debugging of PL/SQL stored procedures and Java stored procedures over JDWP. Using the DBMS_DEBUG_JDWP package, you can:

- Retrieve the session ID of the current session and serial number
- Connect a database session to a debugger over JDWP
- Set the NLS parameters to change the formats in which PL/SQL program values are represented over JDWP

DBMS_DEBUG_JDWP Security Model

The debugging user invoking a DBMS_DEBUG_JDWP subprogram requires a privilege to connect the target database session to the debugger.

Table 73-1 System Privileges

System Privilege	Description
DEBUG CONNECT SESSION	Allows a user to connect his current session to a debugger
DEBUG CONNECT ANY	Allows a user to connect a session by any logon user to a debugger

In addition, the user can also be granted the following user privilege to debug another user's session.

Table 73-2 User Privileges

User Privilege	Description
DEBUG CONNECT ON USER <*user*>	Allows a user to connect any of the specified user's logon sessions to a debugger to debug another user's session or his own

Summary DBMS_DEBUG_JDWP Subprograms

This table lists the `DBMS_DEBUG_JDWP` subprograms and briefly describes them.

Table 73-3 DBMS_DEBUG_JDWP Package Subprograms

Subprogram	Description
CONNECT_TCP Procedure	Connects the specified session to the debugger waiting at `host:port`
CURRENT_SESSION_ID Function	Gets the current session's session ID
CURRENT_SESSION_SERIAL Function	Gets the current session's session serial number
DISCONNECT Procedure	Disconnects the specified session from any debugger with which it is connected
GET_NLS_PARAMETER Function	Gets the value of the specified NLS parameter affecting the format in which `NUMBER`, `DATE`, `TIME (WITH TIME ZONE)` and `TIMESTAMP (WITH TIME ZONE)` runtime values of PL/SQL programs are converted to strings as they are presented through JDWP
PROCESS_CONNECT_STRING Procedure	Connects a session to a debugger without having to directly modify an application's code
SET_NLS_PARAMETER Procedure	Sets the value of the specified NLS parameter affecting the format in which `NUMBER`, `DATE`, `TIME (WITH TIME ZONE)` and `TIMESTAMP (WITH TIME ZONE)` runtime values of PL/SQL programs are converted to strings as they are presented through JDWP

CONNECT_TCP Procedure

This procedure connects the specified session to the debugger waiting at `host:port`.

Syntax

```
DBMS_DEBUG_JDWP.CONNECT_TCP(
   host                 IN  VARCHAR2,
   port                 IN  VARCHAR2,
   session_id           IN  PLS_INTEGER := NULL,
   session_serial       IN  PLS_INTEGER := NULL,
   debug_role           IN  VARCHAR2  := NULL,
   debug_role_pwd       IN  VARCHAR2  := NULL,
   option_flags         IN  PLS_INTEGER := 0,
   extensions_cmd_set   IN  PLS_INTEGER := 128);
```

Parameters

Table 73-4 CONNECT_TCP Parameters

Parameter	Description
`host`	The host name the debugger is waiting at
`port`	The port number the debugger is waiting at

Table 73-4 (Cont.) CONNECT_TCP Parameters

Parameter	Description
session_id	Session ID
session_serial	Session number
debug_role	Debug role
debug_role_pwd	Debug password
option_flags	Values: • 1 : Does not suspend the program until the next client/server request begins. This can be used to hide the startup sequence from end users, who may only want to see their own code • 2 : Forces the connection even if the session appears to be connected to a debugger. This should best only be specified after some human-interaction confirmation step has occurred; i.e., if an attempt without this option raised ORA-30677, then if the user confirms, retry with this bit set. These may be added together to select multiple option choices.
extensions_cmd_set	The ID of the Oracle JDWP extension command set

Exceptions

Table 73-5 CONNECT_TCP Exceptions

Exception	Description
ORA-00022	Invalid session ID
ORA-01031	Insufficient privilege
ORA-30677	Session is already connected to a debugger
ORA-30681	Improper value for argument EXTENSIONS_CMD_SET
ORA-30682	Improper value for argument OPTION_FLAGS
ORA-30683	Failure establishing connection to debugger

Usage Notes

- To connect the current session to a debugger, you can pass NULL to both the session_id and session_serial parameters.
- To connect a different session, you need to find out its ID and serial. These are available in the v$plsql_debuggable_sessions view. The Instance Manager option of Oracle Enterprise Manager is one example of a user interface that displays these values to users. You can also find the values of these for your own session using the CURRENT_SESSION_ID Function and CURRENT_SESSION_SERIAL Function.
- The debug_role and debug_role_pwd arguments allow the user to name any role as the "debug role", which will be available to privilege checking when checking for permissions to

connect the session and when checking permissions available on objects within the debugged session. Both the role and its password are passed here as strings and not as identifiers, so double quotes should not be used but case matters. If the original role name wasn't double-quoted, it should be specified here in upper case.

- An `ORA-30677` indicates that the requested session is already being debugged. It is suggested in this case that the user be asked to confirm that (s)he desires to steal the session from the existing connection, and then either an explicit disconnect call or the use of the `connect_force_connect` option bit can be used to allow the connection to succeed on a second attempt. Note that using the `connect_force_connect` bit will avoid the session being allowed to run freely if it is currently suspended through the debugger - in other words, this bit lets you steal a session from one debugger to another without actually disturbing the state of the session.

CURRENT_SESSION_ID Function

This function gets the current session's session ID

Syntax

```
DBMS_DEBUG_JDWP.CURRENT_SESSION_ID
  RETURN PLS_INTEGER;
```

CURRENT_SESSION_SERIAL Function

This function gets the current session's session number.

Syntax

```
DBMS_DEBUG_JDWP.CURRENT_SESSION_SERIAL
  RETURN PLS_INTEGER;
```

DISCONNECT Procedure

This procedure disconnects the specified session from any debugger with which it is connected.

Syntax

```
DBMS_DEBUG_JDWP.DISCONNECT(
   session_id        IN  PLS_INTEGER  := NULL,
   session_serial    IN  PLS_INTEGER := NULL);
```

Parameters

Table 73-6 DISCONNECT Procedure Parameters

Parameter	Description
`session_id`	Session ID
`session_serial`	Session number

Usage Notes

- If the session to disconnect is the current session, the session will be allowed to run freely after disconnecting the debugger. Otherwise, the session will be terminated.

- The same rights are required for this call as for connect, except when disconnecting the current session and the effective user at the time of the call is the same as the login user of the session where no privilege is required.

GET_NLS_PARAMETER Function

This function gets the value of the specified NLS parameter affecting the format in which the `NUMBER`, `DATE`, `TIME (WITH TIME ZONE)` and `TIMESTAMP (WITH TIME ZONE)` runtime values of PL/SQL programs are converted to strings, as they are presented through JDWP.

These values are private to the current session, but further are private to the debugger mechanisms, separate from the values used to convert values within the debugged program itself.

Syntax

```
DBMS_DEBUG_JDWP.GET_NLS_PARAMETER(
   name    IN  VARCHAR2)
  RETURN VARCHAR2;
```

Parameters

Table 73-7 GET_NLS_PARAMETER Function Parameters

Parameter	Description
`name`	Specified NLS parameter

Usage Notes

- When any variable value is read or assigned through JDWP, or when either `GET_NLS_PARAMETER Function` or `SET_NLS_PARAMETER Procedure` is first invoked in a session, the debugger mechanisms make a private copy of the then-current `NLS_LANGUAGE`, `NLS_TERRITORY`, `NLS_CALENDAR`, `NLS_DATE_LANGUAGE`, `NLS_NUMERIC_CHARACTERS`, `NLS_TIMESTAMP_FORMAT`, `NLS_TIMESTAMP_TZ_FORMAT`, `NLS_TIME_FOMAT` and `NLS_TIME_TZ_FORMAT` values. These private copies may be read using this `GET_NLS_PARAMETER Function`call and changed using the following call to the SET_NLS_PARAMETER Procedure.
- Once the debugger's private copy of the NLS parameters is established, changes made to the NLS parameters in the current session using the `ALTER SESSION` statement will have no effect on the formatting of values as seen through JDWP. To modify the NLS parameters used for JDWP, one must use the `SET_NLS_PARAMETER Procedure` . By the same token, changes made to the debugger's private copy of the NLS parameters using`SET_NLS_PARAMETER Procedure` will have no effect on the debugged program itself.
- Date values are always formatted for JDWP use using the `NLS_TIMESTAMP_FORMAT`. The default format for `DATE (NLS_DATE_FORMAT)` used in a session most often does not show the time information that is in fact present in the value, and for debugging purposes it seems beneficial to always display that information.

PROCESS_CONNECT_STRING Procedure

This procedure connects a session to a debugger in two ways, so that you do not have to directly modify the application code.

The two ways are:

- Using the ORA_DEBUG_JDWP environment variable, when running an OCI program
- Setting a web browser cookie called OWA_DEBUG_<dad>, when running an application through the PL/SQL Web Gateway

Syntax

```
DBMS_DEBUG_JDWP.PROCESS_CONNECT_STRING (
   connect_string          IN  VARCHAR2,
   connect_string_type     IN  PLS_INTEGER);
```

Parameters

Table 73-8 PROCESS_CONNECT_STRING Procedure Parameters

Parameter	Description
connect_string	The ORA_DEBUG_JDWP environment variable or OWA_DEBUG_<dad> cookie value that contains the JDWP connection information such as the host and port number of the debugger to connect to
connect_string_type	Can have the following two values: • 1 if the connect string value is retrieved from the ORA_DEBUG_JDWP environment • 2 if the value is from the OWA_DEBUG_<dad> cookie

Exceptions

Table 73-9 PROCESS_CONNECT_STRING Procedure Exceptions

Exception	Description
ORA-00022	Invalid session ID
ORA-01031	Insufficient privilege
ORA-30677	Session is already connected to a debugger
ORA-30681	Improper value for argument EXTENSIONS_CMD_SET
ORA-30682	Improper value for argument OPTION_FLAGS
ORA-30683	Failure establishing connection to debugger
ORA-30689	Improper value for environment variable ORA_DEBUG_JDWP

SET_NLS_PARAMETER Procedure

This function sets the value of the specified NLS parameter affecting the format in which NUMBER, DATE, TIME (WITH TIME ZONE) and TIMESTAMP (WITH TIME ZONE) runtime values of PL/SQL programs are converted to strings as they are presented through JDWP.

These values are private to the current session, but further are private to the debugger mechanisms, separate from the values used to convert values within the debugged program itself.

Syntax

```
DBMS_DEBUG_JDWP.SET_NLS_PARAMETER(
   name      IN  VARCHAR2,
   value     IN  VARCHAR2);
```

Parameters

Table 73-10 SET_NLS_PARAMETER Procedure Parameters

Parameter	Description
name	Specified NLS parameter
value	Value of specified NLS parameter

Usage Notes

- When any variable value is read or assigned through JDWP, or when either GET_NLS_PARAMETER Function or SET_NLS_PARAMETER Procedure is first invoked in a session, the debugger mechanisms make a private copy of the then-current NLS_LANGUAGE, NLS_TERRITORY, NLS_CALENDAR, NLS_DATE_LANGUAGE, NLS_NUMERIC_CHARACTERS, NLS_TIMESTAMP_FORMAT, NLS_TIMESTAMP_TZ_FORMAT, NLS_TIME_FOMAT and NLS_TIME_TZ_FORMAT values. These private copies may be read by calling the GET_NLS_PARAMETER Function and changed using the following call to the SET_NLS_PARAMETER Procedure.
- Once the debugger's private copy of the NLS parameters is established, changes made to the NLS parameters in the current session using the ALTER SESSION statement will have no effect on the formatting of values as seen through JDWP. To modify the NLS parameters used for JDWP, one must use the SET_NLS_PARAMETER Procedure. By the same token, changes made to the debugger's private copy of the NLS parameters usingSET_NLS_PARAMETER Procedure will have no effect on the debugged program itself.
- Date values are always formatted for JDWP use using the NLS_TIMESTAMP_FORMAT. The default format for DATE (NLS_DATE_FORMAT) used in a session most often does not show the time information that is in fact present in the value, and for debugging purposes it seems beneficial to always display that information.