134
DBMS_MONITOR

The `DBMS_MONITOR` package enables you to use PL/SQL for controlling additional tracing and statistics gathering.

The chapter contains the following topics:

- Summary of DBMS_MONITOR Subprograms

Summary of DBMS_MONITOR Subprograms

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

Table 134-1 DBMS_MONITOR Package Subprograms

Subprogram	Description
CLIENT_ID_STAT_DISABLE Procedure	Disables statistic gathering previously enabled for a given Client Identifier
CLIENT_ID_STAT_ENABLE Procedure	Enables statistic gathering for a given Client Identifier
CLIENT_ID_TRACE_DISABLE Procedure	Disables the trace previously enabled for a given Client Identifier globally for the database
CLIENT_ID_TRACE_ENABLE Procedure	Enables the trace for a given Client Identifier globally for the database
DATABASE_TRACE_DISABLE Procedure	Disables SQL trace for the whole database or a specific instance
DATABASE_TRACE_ENABLE Procedure	Enables SQL trace for the whole database or a specific instance
SERV_MOD_ACT_STAT_DISABLE Procedure	Disables statistic gathering enabled for a given combination of Service Name, `MODULE` and `ACTION`
SERV_MOD_ACT_STAT_ENABLE Procedure	Enables statistic gathering for a given combination of Service Name, `MODULE` and `ACTION`
SERV_MOD_ACT_TRACE_DISABLE Procedure	Disables the trace for `ALL` enabled instances for a or a given combination of Service Name, `MODULE` and `ACTION` name globally
SERV_MOD_ACT_TRACE_ENABLE Procedure	Enables SQL tracing for a given combination of Service Name, `MODULE` and `ACTION` globally unless an `instance_name` is specified
SESSION_TRACE_DISABLE Procedure	Disables the previously enabled trace for a given database session identifier (SID) on the local instance
SESSION_TRACE_ENABLE Procedure	Enables the trace for a given database session identifier (SID) on the local instance

CLIENT_ID_STAT_DISABLE Procedure

This procedure will disable statistics accumulation for all instances and remove the accumulated results from `V$CLIENT_STATS` view enabled by the `CLIENT_ID_STAT_ENABLE` Procedure.

Syntax

```
DBMS_MONITOR.CLIENT_ID_STAT_DISABLE(
   client_id            IN   VARCHAR2);
```

Parameters

Table 134-2 CLIENT_ID_STAT_DISABLE Procedure Parameters

Parameter	Description
client_id	Client Identifier for which statistic aggregation is disabled

Examples

To disable accumulation:

```
EXECUTE DBMS_MONITOR.CLIENT_ID_STAT_DISABLE('janedoe');
```

CLIENT_ID_STAT_ENABLE Procedure

This procedure enables statistic gathering for a given Client Identifier.

Statistics gathering is global for the database and persistent across instance starts and restarts. That is, statistics are enabled for all instances of the same database, including restarts. Statistics are viewable through `V$CLIENT_STATS` views.

Syntax

```
DBMS_MONITOR.CLIENT_ID_STAT_ENABLE(
   client_id            IN   VARCHAR2);
```

Parameters

Table 134-3 *CLIENT_ID_STAT_ENABLE Procedure Parameters*

Parameter	Description
client_id	Client Identifier for which statistic aggregation is enabled

Examples

To enable statistic accumulation for a client with a given client ID:

```
EXECUTE DBMS_MONITOR.CLIENT_ID_STAT_ENABLE('janedoe');
```

CLIENT_ID_TRACE_DISABLE Procedure

This procedure will disable tracing enabled by the `CLIENT_ID_TRACE_ENABLE` Procedure.

Syntax

```
DBMS_MONITOR.CLIENT_ID_TRACE_DISABLE(
 client_id     IN  VARCHAR2);
```

Parameters

Table 134-4 CLIENT_ID_TRACE_DISABLE Procedure Parameters

Parameter	Description
`client_id`	Client Identifier for which SQL tracing is disabled

Examples

```
EXECUTE DBMS_MONITOR.CLIENT_ID_TRACE_DISABLE ('janedoe');
```

CLIENT_ID_TRACE_ENABLE Procedure

This procedure will enable the trace for a given client identifier globally for the database.

Syntax

```
DBMS_MONITOR.CLIENT_ID_TRACE_ENABLE(
 client_id     IN  VARCHAR2,
 waits         IN  BOOLEAN DEFAULT TRUE,
 binds         IN  BOOLEAN DEFAULT FALSE,
 plan_stat     IN  VARCHAR2 DEFAULT NULL);
```

Parameters

Table 134-5 CLIENT_ID_TRACE_ENABLE Procedure Parameters

Parameter	Description
`client_id`	Database Session Identifier for which SQL tracing is enabled
`waits`	If `TRUE`, wait information is present in the trace
`binds`	If `TRUE`, bind information is present in the trace
`plan_stat`	Frequency at which we dump row source statistics. Value should be `'NEVER'`, `'FIRST_EXECUTION'` (equivalent to `NULL`) or `'ALL_EXECUTIONS'`.

Usage Notes

- The trace will be written to multiple trace files because more than one Oracle shadow process can work on behalf of a given client identifier.
- The tracing is enabled for all instances and persistent across restarts.

Examples

```
EXECUTE DBMS_MONITOR.CLIENT_ID_TRACE_ENABLE('janedoe', TRUE,
FALSE);
```

DATABASE_TRACE_DISABLE Procedure

This procedure disables SQL trace for the whole database or a specific instance.

Syntax

```
DBMS_MONITOR.DATABASE_TRACE_DISABLE(
   instance_name  IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 134-6 DATABASE_TRACE_DISABLE Procedure Parameters

Parameter	Description
instance_name	Disables tracing for the named instance

DATABASE_TRACE_ENABLE Procedure

This procedure enables SQL trace for the whole database or a specific instance.

Syntax

```
DBMS_MONITOR.DATABASE_TRACE_ENABLE(
   waits          IN BOOLEAN DEFAULT TRUE,
   binds          IN BOOLEAN DEFAULT FALSE,
   instance_name  IN VARCHAR2 DEFAULT NULL,
   plan_stat      IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 134-7 DATABASE_TRACE_ENABLE Procedure Parameters

Parameter	Description
waits	If TRUE, wait information will be present in the trace
binds	If TRUE, bind information will be present in the trace
instance_name	If set, restricts tracing to the named instance
plan_stat	Frequency at which we dump row source statistics. Value should be 'NEVER', 'FIRST_EXECUTION' (equivalent to NULL) or 'ALL_EXECUTIONS'.

SERV_MOD_ACT_STAT_DISABLE Procedure

This procedure will disable statistics accumulation and remove the accumulated results from `V$SERV_MOD_ACT_STATS` view.

Statistics disabling is persistent for the database. That is, service statistics are disabled for instances of the same database (plus `dblinks` that have been activated as a result of the enable).

Syntax

```
DBMS_MONITOR.SERV_MOD_ACT_STAT_DISABLE(
   service_name     IN VARCHAR2,
   module_name      IN VARCHAR2,
   action_name      IN VARCHAR2 DEFAULT ALL_ACTIONS);
```

Parameters

Table 134-8 SERV_MOD_ACT_STAT_DISABLE Procedure Parameters

Parameter	Description
`service_name`	Name of the service for which statistic aggregation is disabled
`module_name`	Name of the `MODULE`. An additional qualifier for the service. It is a required parameter.
`action_name`	Name of the `ACTION`. An additional qualifier for the Service and `MODULE` name. Omitting the parameter (or supplying `ALL_ACTIONS` constant) means enabling aggregation for all Actions for a given Service/ MODULE combination. In this case, statistics are aggregated on the module level.

Usage Notes

Regarding statistics gathering, when you change the module or action, the change takes effect when the next user call is executed in the session. For example, if a module is set to 'module 1' in a session, and the module is reset to 'module 2' in a user call in the session, then the module remains 'module 1' during this user call. The module is changed to 'module 2' in the next user call in the session.

SERV_MOD_ACT_STAT_ENABLE Procedure

This procedure enables statistic gathering for a given combination of Service Name, `MODULE` and `ACTION`.

Calling this procedure enables statistic gathering for a hierarchical combination of Service name, `MODULE` name, and `ACTION` name on all instances for the same database. Statistics are accessible by means of the `V$SERV_MOD_ACT_STATS` view.

Syntax

```
DBMS_MONITOR.SERV_MOD_ACT_STAT_ENABLE(
   service_name     IN VARCHAR2,
   module_name      IN VARCHAR2,
   action_name      IN VARCHAR2 DEFAULT ALL_ACTIONS);
```

Parameters

Table 134-9 SERV_MOD_ACT_STAT_ENABLE Procedure Parameters

Parameter	Description
service_name	Name of the service for which statistic aggregation is enabled
module_name	Name of the MODULE. An additional qualifier for the service. It is a required parameter.
action_name	Name of the ACTION. An additional qualifier for the Service and MODULE name. Omitting the parameter (or supplying ALL_ACTIONS constant) means enabling aggregation for all Actions for a given Service/ MODULE combination. In this case, statistics are aggregated on the module level.

Usage Notes

Enabling statistic aggregation for the given combination of Service/Module/Action names is slightly complicated by the fact that the Module/Action values can be empty strings which are indistinguishable from NULLs. For this reason, we adopt the following conventions:

A special constant (unlikely to be a real action names) is defined:

```
ALL_ACTIONS constant VARCHAR2 := '###ALL_ACTIONS';
```

Using ALL_ACTIONS for an action specification means that aggregation is enabled for all actions with a given module name, while using NULL (or empty string) means that aggregation is enabled for an action whose name is an empty string.

Regarding statistics gathering, when you change the module or action, the change takes effect when the next user call is executed in the session. For example, if a module is set to 'module 1' in a session, and the module is reset to 'module 2' in a user call in the session, then the module remains 'module 1' during this user call. The module is changed to 'module 2' in the next user call in the session.

Examples

To enable statistic accumulation for a given combination of Service name and MODULE:

```
EXECUTE DBMS_MONITOR.SERV_MOD_ACT_STAT_ENABLE( 'APPS1','PAYROLL');
```

To enable statistic accumulation for a given combination of Service name, MODULE and ACTION:

```
EXECUTE
DBMS_MONITOR.SERV_MOD_ACT_STAT_ENABLE('APPS1','GLEDGER','DEBIT_ENTRY');
```

If both of the preceding commands are issued, statistics are accumulated as follows:

- For the APPS1 service, because accumulation for each Service Name is the default.
- For all actions in the PAYROLL Module.
- For the DEBIT_ENTRY Action within the GLEDGER Module.

SERV_MOD_ACT_TRACE_DISABLE Procedure

This procedure will disable the trace at `ALL` enabled instances for a given combination of Service Name, `MODULE`, and `ACTION` name globally.

Syntax

```
DBMS_MONITOR.SERV_MOD_ACT_TRACE_DISABLE(
   service_name     IN  VARCHAR2,
   module_name      IN  VARCHAR2,
   action_name      IN  VARCHAR2 DEFAULT ALL_ACTIONS,
   instance_name    IN  VARCHAR2 DEFAULT NULL);
```

Parameters

Table 134-10 SERV_MOD_ACT_TRACE_DISABLE Procedure Parameters

Parameter	Description
`service_name`	Name of the service for which tracing is disabled.
`module_name`	Name of the `MODULE`. An additional qualifier for the service
`action_name`	Name of the `ACTION`. An additional qualifier for the Service and `MODULE` name.
`instance_name`	If set, this restricts tracing to the named `instance_name`

Usage Notes

Specifying `NULL` for the `module_name` parameter means that statistics will no longer be accumulated for the sessions which do not set the `MODULE` attribute.

Examples

To enable tracing for a Service named APPS1:

```
EXECUTE DBMS_MONITOR.SERV_MOD_ACT_TRACE_ENABLE('APPS1',
            DBMS_MONITOR.ALL_MODULES,  DBMS_MONITOR.ALL_ACTIONS,TRUE,
FALSE,NULL);
```

To disable tracing specified in the previous step:

```
EXECUTE DBMS_MONITOR.SERV_MOD_ACT_TRACE_DISABLE('APPS1');
```

To enable tracing for a given combination of `Service` and `MODULE` (all `ACTION`s):

```
EXECUTE DBMS_MONITOR.SERV_MOD_ACT_TRACE_ENABLE('APPS1','PAYROLL',
            DBMS_MONITOR.ALL_ACTIONS,TRUE,FALSE,NULL);
```

To disable tracing specified in the previous step:

```
EXECUTE DBMS_MONITOR.SERV_MOD_ACT_TRACE_DISABLE('APPS1','PAYROLL');
```

SERV_MOD_ACT_TRACE_ENABLE Procedure

This procedure will enable SQL tracing for a given combination of Service Name, MODULE and ACTION globally unless an instance_name is specified.

Syntax

```
DBMS_MONITOR.SERV_MOD_ACT_TRACE_ENABLE(
   service_name     IN VARCHAR2,
   module_name      IN VARCHAR2 DEFAULT ANY_MODULE,
   action_name      IN VARCHAR2 DEFAULT ANY_ACTION,
   waits            IN BOOLEAN DEFAULT TRUE,
   binds            IN BOOLEAN DEFAULT FALSE,
   instance_name    IN VARCHAR2 DEFAULT NULL,
   plan_stat        IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 134-11 SERV_MOD_ACT_TRACE_ENABLE Procedure Parameters

Parameter	Description
service_name	Name of the service for which SQL trace is enabled
module_name	Name of the MODULE for which SQL trace is enabled. An optional additional qualifier for the service. If omitted, SQL trace is enabled or all modules and actions in a given service.
action_name	Name of the ACTION for which SQL trace is enabled. An optional additional qualifier for the Service and MODULE name. If omitted, SQL trace is enabled for all actions in a given module.
waits	If TRUE, wait information is present in the trace
binds	If TRUE, bind information is present in the trace
instance_name	If set, this restricts tracing to the named instance_name
plan_stat	Frequency at which we dump row source statistics. Value should be 'NEVER', 'FIRST_EXECUTION' (equivalent to NULL) or 'ALL_EXECUTIONS'.

Usage Notes

- The procedure enables a trace for a given combination of Service, MODULE and ACTION name. The specification is strictly hierarchical: Service Name or Service Name/MODULE, or Service Name, MODULE, and ACTION name must be specified. Omitting a qualifier behaves like a wild-card, so that not specifying an ACTION means all ACTIONs. Using the ALL_ACTIONS constant achieves the same purpose.
- This tracing is useful when an application MODULE and optionally known ACTION is experiencing poor service levels.
- By default, tracing is enabled globally for the database. The instance_name parameter is provided to restrict tracing to named instances that are known, for example, to exhibit poor service levels.
- Tracing information is present in multiple trace files and you must use the trcsess tool to collect it into a single file.

- Specifying `NULL` for the `module_name` parameter means that statistics will be accumulated for the sessions which do not set the `MODULE` attribute.

Examples

To enable tracing for a Service named `APPS1`:

```
EXECUTE DBMS_MONITOR.SERV_MOD_ACT_TRACE_ENABLE('APPS1',
            DBMS_MONITOR.ALL_MODULES,  DBMS_MONITOR.ALL_ACTIONS,TRUE,
FALSE,NULL);
```

To enable tracing for a given combination of Service and `MODULE` (all `ACTION`s):

```
EXECUTE DBMS_MONITOR.SERV_MOD_ACT_TRACE_ENABLE('APPS1','PAYROLL',
            DBMS_MONITOR.ALL_ACTIONS,TRUE,FALSE,NULL);
```

SESSION_TRACE_DISABLE Procedure

This procedure will disable the trace for a given database session at the local instance.

Syntax

```
DBMS_MONITOR.SESSION_TRACE_DISABLE(
   session_id      IN     BINARY_INTEGER DEFAULT NULL,
   serial_num      IN     BINARY_INTEGER DEFAULT NULL);
```

Parameters

Table 134-12 SESSION_TRACE_DISABLE Procedure Parameters

Parameter	Description
`session_id`	Database Session Identifier for which SQL trace is disabled
`serial_num`	Serial number for this session

Usage Notes

If `serial_num` is `NULL` but `session_id` is specified, a session with a given `session_id` is no longer traced irrespective of its serial number. If both `session_id` and `serial_num` are `NULL`, the current user session is no longer traced. It is illegal to specify `NULL session_id` and non-`NULL serial_num`. In addition, the `NULL` values are default and can be omitted.

Examples

To enable tracing for a client with a given client session ID:

```
EXECUTE DBMS_MONITOR.SESSION_TRACE_ENABLE(7,4634, TRUE, FALSE);
```

To disable tracing specified in the previous step:

```
EXECUTE DBMS_MONITOR.SESSION_TRACE_DISABLE(7,4634);;
```

SESSION_TRACE_ENABLE Procedure

This procedure enables a SQL trace for the given Session ID on the local instance

Syntax

```
DBMS_MONITOR.SESSION_TRACE_ENABLE(
    session_id   IN  BINARY_INTEGER DEFAULT NULL,
    serial_num   IN  BINARY_INTEGER DEFAULT NULL,
    waits        IN  BOOLEAN DEFAULT TRUE,
    binds        IN  BOOLEAN DEFAULT FALSE,
    plan_stat    IN  VARCHAR2 DEFAULT NULL);
```

Parameters

Table 134-13 SESSION_TRACE_ENABLE Procedure Parameters

Parameter	Description
session_id	Client Identifier for which SQL trace is enabled. If omitted (or NULL), the user's own session is assumed.
serial_num	Serial number for this session. If omitted (or NULL), only the session ID is used to determine a session.
waits	If TRUE, wait information is present in the trace
binds	If TRUE, bind information is present in the trace
plan_stat	Frequency at which we dump row source statistics. Value should be 'NEVER', 'FIRST_EXECUTION' (equivalent to NULL) or 'ALL_EXECUTIONS'.

Usage Notes

The procedure enables a trace for a given database session, and is still useful for client/server applications. The trace is enabled only on the instance to which the caller is connected, since database sessions do not span instances. This tracing is strictly local to an instance.

If serial_num is NULL but session_id is specified, a session with a given session_id is traced irrespective of its serial number. If both session_id and serial_num are NULL, the current user session is traced. It is illegal to specify NULL session_id and non-NULL serial_num. In addition, the NULL values are default and can be omitted.

Examples

To enable tracing for a client with a given client session ID:

```
EXECUTE DBMS_MONITOR.SESSION_TRACE_ENABLE(7,4634, TRUE, FALSE);
```

To disable tracing specified in the previous step:

```
EXECUTE DBMS_MONITOR.SESSION_TRACE_DISABLE(7,4634);
```

Either

```
EXECUTE DBMS_MONITOR.SESSION_TRACE_ENABLE(5);
```

or

```
EXECUTE DBMS_MONITOR.SESSION_TRACE_ENABLE(5, NULL);
```

traces the session with session ID of 5, while either

```
EXECUTE DBMS_MONITOR.SESSION_TRACE_ENABLE();
```

or

```
EXECUTE DBMS_MONITOR.SESSION_TRACE_ENABLE(NULL, NULL);
```

traces the current user session. Also,

```
EXECUTE DBMS_MONITOR.SESSION_TRACE_ENABLE(NULL, NULL, TRUE, TRUE);
```

traces the current user session including waits and binds. The same can be also expressed using keyword syntax:

```
EXECUTE DBMS_MONITOR.SESSION_TRACE_ENABLE(binds=>TRUE);
```