

DBMS_SPD

The `DBMS_SPD` package provides subprograms for managing SQL plan directives (SPD).

This chapter contains the following topics:

- [Overview](#)
- [Security Model](#)
- [Views](#)
- [Summary of DBMS_SPD Subprograms](#)



See Also:

- [DBMS_STATS](#)
- *Oracle Database SQL Tuning Guide* regarding SQL plan directives

DBMS_SPD Overview

This package provides subprograms for managing SQL plan directives (SPD).

SPD are objects generated automatically by Oracle. For example, if Oracle detects that the single table cardinality estimated made by the optimizer is different from the actual number of rows returned when accessing the table, it will automatically create a directive to perform dynamic statistics for the table. When any SQL statement referencing the table is compiled, the optimizer will perform dynamic statistics for the table to get a more accurate estimate.

DBMS_SPD Security Model

`DBMS_SPD` is an invoker-rights package. The invoker requires `ADMINISTER SQL MANAGEMENT OBJECT` privilege for executing most of the subprograms in this package. Also, the subprograms commit the current transaction (if any), perform the operation, and then commit it again.

DBMS_SPD Views

The DBA view `DBA_SQL_PLAN_DIRECTIVES` shows all the directives created in the system and the view `DBA_SQL_PLAN_DIR_OBJECTS` displays the objects that are included in the directives.

Summary of DBMS_SPD Subprograms

This table lists and briefly describes the `DBMS_SPD` package subprograms.

Table 185-1 DBMS_SPD Package Subprograms

Subprogram	Description
ALTER_SQL_PLAN_DIRECTIVE Procedure	Changes different attributes of a SQL plan directive
CREATE_STGTAB_DIRECTIVE Procedure	Creates a staging table into which to pack (export) SQL plan directives.
DROP_SQL_PLAN_DIRECTIVE Procedure	Drops a SQL plan directive
FLUSH_SQL_PLAN_DIRECTIVE Procedure	Allows for manual flushing of the SQL plan directives that are automatically recorded in SGA memory while executing SQL statements.
GET_PREFS Function	Gets the values for preferences for SQL plan directives
PACK_STGTAB_DIRECTIVE Function	Packs (exports) SQL plan directives into a staging table.
SET_PREFS Procedure	Allows the setting of different preferences for SQL plan directives
UNPACK_STGTAB_DIRECTIVE Function	Unpacks (imports) SQL plan directives from a staging table.

ALTER_SQL_PLAN_DIRECTIVE Procedure

This procedure changes different attributes of a SQL plan directive.

Syntax

```
DBMS_SPD.ALTER_SQL_PLAN_DIRECTIVE (  
    directive_id      IN      NUMBER,  
    attribute_name     IN      VARCHAR2,  
    attribute_value    IN      VARCHAR2);
```

Parameters

Table 185-2 ALTER_SQL_PLAN_DIRECTIVE Procedure Parameters

Parameter	Description
directive_id	SQL plan directive ID
attribute_name	<ul style="list-style-type: none">ENABLEDAUTO_DROP
attribute_value	Possible values: <ul style="list-style-type: none">ENABLED:<ul style="list-style-type: none">If YES directive is enabled and may be usedIf NO directive is not enabled and will not be usedAUTO_DROP:<ul style="list-style-type: none">If YES directive will be dropped automatically if not used for SPD_RETENTION_WEEKS. This is the default behavior.If NO directive will not be dropped automatically

Exceptions

- ORA-38171 INSUFFICIENT_PRIVILEGE: The user does not have proper privilege to perform the operation.

- ORA-28104 INVALID_INPUT: The input value is not valid.
- ORA-13158 OBJECT_DOES_NOT_EXIST: The specified object does not exist.

Usage Notes

The ADMINISTER SQL MANAGEMENT OBJECT privilege is required to execute this procedure.

Examples

```
BEGIN
  DBMS_SPD.ALTER_SQL_PLAN_DIRECTIVE (12345, 'STATE', 'PERMANENT');
END;
```

CREATE_STGTAB_DIRECTIVE Procedure

This procedure creates a staging table into which to pack (export) SQL plan directives.

Syntax

```
DBMS_SPD.CREATE_STGTAB_DIRECTIVE (
  table_name      IN VARCHAR2,
  table_owner     IN VARCHAR2 := USER,
  tablespace_name IN VARCHAR2 := NULL);
```

Parameters

Table 185-3 *CREATE_STGTAB_DIRECTIVE Procedure Parameters*

Parameter	Description
table_name	Name of staging table
table_owner	Name of schema owner of staging table. Default is current schema.
tablespace_name	Name of tablespace. Default NULL means create staging table in the default tablespace:

Exceptions

- ORA-38171 INSUFFICIENT_PRIVILEGE: The user does not have proper privilege to perform the operation.
- ORA-28104 INVALID_INPUT: The input value is not valid.
- ORA-44001 INVALID_SCHEMA: The input schema does not exist.
- ORA-13159 TABLE_ALREADY_EXISTS: The specified table already exists.
- ORA-29304 TABLESPACE_MISSING: The specified tablespace does not exist.

Usage Notes

The ADMINISTER SQL MANAGEMENT OBJECT privilege is required to execute this procedure.

DROP_SQL_PLAN_DIRECTIVE Procedure

This procedure drops a SQL plan directive.

Syntax

```
DBMS_SPD.DROP_SQL_PLAN_DIRECTIVE (  
    directive_id      IN      NUMBER);
```

Parameters

Table 185-4 *DROP_SQL_PLAN_DIRECTIVE Procedure Parameters*

Parameter	Description
directive_id	SQL plan directive ID

Exceptions

- ORA-38171 `INSUFFICIENT_PRIVILEGE`: The user does not have proper privilege to perform the operation.
- ORA-28104 `INVALID_INPUT`: The input value is not valid.
- ORA-13158 `OBJECT_DOES_NOT_EXIST`: The specified object does not exist.

Usage Notes

The `ADMINISTER SQL MANAGEMENT OBJECT` privilege is required to execute this procedure.

Examples

```
BEGIN  
    DBMS_SPD.DROP_SQL_PLAN_DIRECTIVE (12345);  
END;
```

FLUSH_SQL_PLAN_DIRECTIVE Procedure

This procedure allows for manual flushing of the SQL plan directives that are automatically recorded in SGA memory while executing SQL statements.

The information recorded in the SGA is periodically flushed by an Oracle background process. This procedure provides a way to flush the information manually.

Syntax

```
DBMS_SPD.FLUSH_SQL_PLAN_DIRECTIVE;
```

Exceptions

ORA-38171 `INSUFFICIENT_PRIVILEGE`: The user does not have proper privilege to perform the operation.

Usage Notes

The `ADMINISTER SQL MANAGEMENT OBJECT` privilege is required to execute this procedure.

Examples

```
BEGIN
  DBMS_SPD.FLUSH_SQL_PLAN_DIRECTIVE;
END;
```

GET_PREFS Function

This function returns the value for the specified preferences for SQL plan directives.

Syntax

```
DBMS_SPD.GET_PREFS (
  pname      IN   VARCHAR2)
RETURN VARCHAR2;
```

Parameters

Table 185-5 *GET_PREFS Function Parameters*

Parameter	Description
pname	Preference name. The procedure supports the preference SPD_RETENTION_WEEKS.

Return Values

Preference value

Exceptions

- ORA-38171 *INSUFFICIENT_PRIVILEGE*: The user does not have proper privilege to perform the operation.
- ORA-28104 *INVALID_INPUT*: The input value is not valid.

Usage Notes

- The `ADMINISTER SQL MANAGEMENT OBJECT` privilege is required to execute this procedure.
- `SPD_RETENTION_WEEKS` - SQL plan directives are purged if not used for more than the value set for this preference.

Examples

```
SELECT DBMS_SPD.GET_PREFS('SPD_RETENTION_WEEKS') FROM DUAL;
```

PACK_STGTAB_DIRECTIVE Function

This function packs (exports) SQL plan directives into a staging table.

Syntax

```
DBMS_SPD.PACK_STGTAB_DIRECTIVE (
  table_name      IN VARCHAR2,
  table_owner     IN VARCHAR2 := USER
  directive_id    IN NUMBER   := NULL,
```

```
obj_list          IN OBJECTTAB := NULL)
RETURN NUMBER
```

Parameters

Table 185-6 *PACK_STGTAB_DIRECTIVE Function Parameters*

Parameter	Description
table_name	Name of staging table
table_owner	Name of schema owner of staging table. Default is current schema.
directive_id	SQL plan directive ID. Default NULL means all directives in the system.
obj_list	Used to filter the directives to be packed based on the objects used in directives. If obj_list is not NULL, a directive is packed only if all the objects in the directive exist in obj_list.

Return Values

Number of SQL plan directives packed.

Exceptions

- ORA-38171 INSUFFICIENT_PRIVILEGE: The user does not have proper privilege to perform the operation.
- ORA-28104 INVALID_INPUT: The input value is not valid.
- ORA-44001 INVALID_SCHEMA: The input schema does not exist.
- ORA-29304 INVALID_STGTAB: The specified staging table is invalid or does not exist.
- ORA-13158 OBJECT_DOES_NOT_EXIST: The specified object does not exist.

Usage Notes

The ADMINISTER SQL MANAGEMENT OBJECT privilege is required to execute this procedure.

Examples

```
-- Pack all directives in the system
SELECT DBMS_SPD.PACK_STGTAB_DIRECTIVE('mydirtab') FROM DUAL;

SET SERVEROUTPUT ON;
-- Pack directives relevant to objects in SH schema
DECLARE
    my_list DBMS_SPD.OBJECTTAB := DBMS_SPD.ObjectTab();
    dir_cnt NUMBER;
BEGIN
    my_list.extend(1);
    my_list(1).owner := 'SH';           -- schema name
    my_list(1).object_name := NULL;     -- all tables in SH
    my_list(1).object_type := 'TABLE';  -- type of object

    dir_cnt :=
        DBMS_SPD.PACK_STGTAB_DIRECTIVE('mydirtab', obj_list => my_list);
    DBMS_OUTPUT.PUT_LINE('dir_cnt = ' || dir_cnt);
END;

-- Pack directives relevant to tables SALES and CUSTOMERS in SH schema
DECLARE
```

```

my_list DBMS_SPD.OBJECTTAB := DBMS_SPD.ObjectTab();
dir_cnt NUMBER;
BEGIN
    my_list.extend(2);

    -- SALES table
    my_list(1).owner := 'SH';
    my_list(1).object_name := 'SALES';
    my_list(1).object_type := 'TABLE';

    -- CUSTOMERS table
    my_list(2).owner := 'SH';
    my_list(2).object_name := 'CUSTOMERS';
    my_list(2).object_type := 'TABLE';

    dir_cnt :=
        DBMS_SPD.PACK_STGTAB_DIRECTIVE('mydirtab', obj_list => my_list);
    DBMS_OUTPUT.PUT_LINE('dir_cnt = ' || dir_cnt);
END;
```

SET_PREFS Procedure

This procedure allows the setting of different preferences for SQL plan directives.

Syntax

```

DBMS_SPD.SET_PREFS (
    pname      IN   VARCHAR2,
    pvalue     IN   VARCHAR2);
```

Parameters

Table 185-7 *SET_PREFS Procedure Parameters*

Parameter	Description
pname	Preference name. The procedure supports the preference SPD_RETENTION_WEEKS.
pvalue	Preference value. <ul style="list-style-type: none"> SPD_RETENTION_WEEKS: SQL plan directives are purged if not used for more than the value set for this preference. Default is 53 (SPD_RETENTION_WEEKS_DEFAULT) weeks, which means a directive is purged if it has been left unused for little over a year. It can be set to any value greater than or equal to 0. Also value NULL can be passed to set the preference to default.

Exceptions

- ORA-38171 INSUFFICIENT_PRIVILEGE: The user does not have proper privilege to perform the operation.
- ORA-28104 INVALID_INPUT: The input value is not valid.

Usage Notes

- The ADMINISTER SQL MANAGEMENT OBJECT privilege is required to execute this procedure.
- SPD_RETENTION_WEEKS - SQL plan directives are purged if not used for more than the value set for this preference.

Examples

```
BEGIN
  DBMS_SPD.SET_PREFS('SPD_RETENTION_WEEKS', '4');
END;
```

UNPACK_STGTAB_DIRECTIVE Function

This procedure unpacks (imports) SQL plan directives from a staging table.

Syntax

```
DBMS_SPD.UNPACK_STGTAB_DIRECTIVE (
  table_name          IN VARCHAR2,
  table_owner         IN VARCHAR2  := USER,
  directive_id        IN NUMBER    := NULL,
  obj_list            IN OBJECTTAB := NULL)
RETURN NUMBER
```

Parameters

Table 185-8 *UNPACK_STGTAB_DIRECTIVE Function Parameters*

Parameter	Description
table_name	Name of staging table
table_owner	Name of schema owner of staging table. Default is current schema.
directive_id	SQL plan directive ID. Default NULL means all directives in the system.
obj_list	Used to filter the directives to be unpacked based on the objects used in directives. If obj_list is not NULL, a directive is unpacked only if all the objects in the directive exist in obj_list.

Return Values

Number of SQL plan directives unpacked.

Exceptions

- ORA-38171 INSUFFICIENT_PRIVILEGE: The user does not have proper privilege to perform the operation.
- ORA-28104 INVALID_INPUT: The input value is not valid.
- ORA-44001 INVALID_SCHEMA: The input schema does not exist.
- ORA-29304 INVALID_STGTAB: The specified staging table is invalid or does not exist.
- ORA-13158 OBJECT_DOES_NOT_EXIST: The specified object does not exist.

Usage Notes

The ADMINISTER SQL MANAGEMENT OBJECT privilege is required to execute this procedure.

Examples

```
-- Unack all directives in the staging table
SELECT DBMS_SPD.UNPACK_STGTAB_DIRECTIVE('mydirtab') FROM DUAL;
```



```
SET SERVEROUTPUT ON;
-- Unpack directives relevant to objects in SH schema
DECLARE
  my_list  DBMS_SPD.OBJECTTAB := DBMS_SPD.ObjectTab();
  dir_cnt  number;
BEGIN
  my_list.extend(1);
  my_list(1).owner := 'SH';           -- schema name
  my_list(1).object_name := null;     -- all tables in SH
  my_list(1).object_type := 'TABLE';  -- type of object

  dir_cnt :=
    DBMS_SPD.UNPACK_STGTAB_DIRECTIVE('mydirtab', obj_list => my_list);
  DBMS_OUTPUT.PUT_LINE('dir_cnt = ' || dir_cnt);
END;

-- Unpack directives relevant to tables SALES and CUSTOMERS in SH schema
DECLARE
  my_list  DBMS_SPD.OBJECTTAB := DBMS_SPD.ObjectTab();
  dir_cnt  NUMBER;
begin
  my_list.extend(2);

  -- SALES table
  my_list(1).owner := 'SH';
  my_list(1).object_name := 'SALES';
  my_list(1).object_type := 'TABLE';

  -- CUSTOMERS table
  my_list(2).owner := 'SH';
  my_list(2).object_name := 'CUSTOMERS';
  my_list(2).object_type := 'TABLE';

  dir_cnt :=
    DBMS_SPD.UNPACK_STGTAB_DIRECTIVE('mydirtab', obj_list => my_list);
  DBMS_OUTPUT.PUT_LINE('dir_cnt = ' || dir_cnt);
END;
```