Adaptive Cursor Sharing

Adaptive cursor sharing was introduced in 11gR1 to address SQL performance consistency issues related to bind variable peeking. When a statement is populated with very different data in bind values, bind peeking can result in an improper execution plan being used for subsequent statements.

Cursor Sharing itself was designed to work around performance problems caused by applications that do not use bind variables. The database notices similar statements when parsing. The SQL area used to process the first occurrence of a statement is used to process subsequent occurrences. Only one shared SQL area exists for a unique statement. Because shared SQL areas are in Shared Memory, any process can use a shared SQL area. Sharing of SQL areas reduces memory use on the database server, thereby increasing performance.

With Adaptive Cursor Sharing, the database compares the effectiveness of execution plans between executions with different bind variable values. If it notices suboptimal plans, it allows certain bind variable values, or ranges of values, to use alternate execution plans for the same statement.

Parameter CURSOR_SHARING

CURSOR_SHARING can be set to FORCE, SIMILAR or EXACT.

Setting CURSOR_SHARING to EXACT allows SQL statements to share the SQL area only when their texts match exactly. This is the default behavior.

SIMILAR results in the database sharing cursors also for statemements that are not exactly identical, but differ in their bind variables. This option is deprecated sind Oracle 11gR2.

Setting CURSOR_SHARING to FORCE forces similar statements to share the executable SQL area, potentially deteriorating execution plans.

Bind aware hint

Starting in 11.1.0.7, the BIND_AWARE hint can be used to force bind-aware cursor sharing from the first execution. The hint only works if all of the other criteria required for bind-aware cursor sharing are met; for instance, the query must have binds, the binds must appear in where clause predicates, and the columns in those predicates must have the proper statistics to allow the plan to actually change when the query is executed with different bind values.

```
/+ BIND_AWARE/
```

If you want to get the 10g behavior and if you want to disable this feature, you may want to add the hint:

```
/*+ NO_BIND_AWARE */.
```

Turning off Adaptive Cursor Sharing completely

According to MOS note 11657468.8, adaptive cursor sharing can be disabled by setting the following 2 parameters:

```
_optimizer_adaptive_cursor_sharing = false
_optimizer_extended_cursor_sharing_rel = "none"
```

These can be set at session level.

See also:

- MOS note 740052.1
- Adaptive Cursor Sharing in Oracle Database 11g Release 1
- On Silver Bullets 🍔
- Why are there more cursors in 11g for my query containing bind variables?
- Explain adaptive cursor sharing behavior with cursor_sharing = similar and force.