

Oracle Global Temporary Tables

Question: What are global temporary table and how do they differ fro regular Oracle tables? Also, please show when to use global temporary tables.

Answer: Oracle introduced Global Temporary Tables (GTT) starting in Oracle8i for removing complex subqueries and allowing us to materialize the intermediate data that we need to solve a complex problem with SQL. Global temporary tables are an alternative to using the WITH clause to materialize intermediate query results.

Tuning Oracle SQL with Global Temporary Tables

12c note: Starting in 12c, Oracle will allow you to invoke session-level `dbms_stats` to gather statistics specific to your own global temporary table. Prior to 12c, statistics were shared from a master copy of the CBO statistics.

In addition to data dictionary queries, global temporary tables can dramatically improve the performance of certain SQL self-join queries that summarize data values.

The Elapsed-Time Section of the Date Range Report is a very sophisticated DBA report, and one that can run for many hours without the use of global temporary tables because of Oracle s use of the CARTESIAN access method.

However, with the use of global temporary tables, the table and index counts can be summarized and saved in the temp tables for fast analysis. We also use the same technique to sum the number of bytes in all tables and indexes into temporary tables, and then quickly interrogate the summary tables for total sizes of our database.

You can use the global temporary tables (GTT) syntax to improve the speed of queries that perform complex summarization activities.

We can also use the following query to display all Oracle global temporary tables:

```
select table_name from all_tables where temporary = 'Y';
```

Upon close examination, we see that we create global temporary tables to hold the total counts, and we also create two temporary tables to hold the sum of bytes for each table and index. Once the sums are pre-calculated, it becomes fast and easy for Oracle SQL to compute the total bytes for the whole database.

The Oracle database codified this approach starting and Oracle8i with their global temporary tables construct. Here is an example of using global temporary tables. Note that the SQL data from the global temporary tables are preserved and passed (as if it was a "real" table) to a subsequent SQL query:

```
drop table store_sales;
```

```
drop table store_cnt;
```

```
drop table store_qty;
```

```
create global temporary table store_qty
```

```
on commit preserve rows
```

```
as select sum(quantity) all_sales from sales;
```

```
create global temporary table store_cnt
```

```
on commit preserve rows
```

```
as select count(*) nbr_stores from store;
```

```
create global temporary table store_sales
on commit preserve rows
as select store_name, sum(quantity) store_sales
from store natural join sales group by store_name;
```

```
select
    store_name,
    store_sales,
    all_sales / nbr_stores avg_sales
from
    store_qty,
    store_cnt,
    store_sales
where
    store_sales > (all_sales / nbr_stores);
```