Automatic Database Diagnostic Monitoring

Overview:

The main goal of the manageability infrastructure is to provide a self-monitoring and self-tuning database. This feature should free the DBA/APPSDBA from critical tasks and allow more time for providing strategic direction. Rather than recommending the DBA manually analyze AWR and ASH data, Oracle provides the Automatic Database Diagnostic Monitor (ADDM) to automatically analyze the data collected by the monitoring tools. By following a set of tuning rules developed by Oracle experts. The ADDM analysis will list areas for improvement in its reports that are not necessarily critical. We should remember that the goal of tuning is to seek the areas where we can make the biggest improvement.

Tools like AWR and ASH, Enterprise Manager (EM) provides a rich, graphical interface to ADDM. EM’s interface for these tools provides the easiest method of interaction. However, we may still perform analysis and build reports manually as a database user with the ADVISOR privilege by using the Oracle-provided DBMS_ADVISOR package.

An ADDM analysis task is performed and its findings and recommendations stored in the database every time an AWR snapshot is taken provided the STATISTICS_LEVEL parameter is set to TYPICAL or ALL. The ADDM analysis includes the following.

- CPU load
- Memory usage
- I/O usage
- Resource intensive SQL
- Resource intensive PL/SQL and Java
- RAC issues
- Application issues
- Database configuration issues
- Concurrency issues
- Object contention

ADDM Setup

Although the Automatic Database Diagnostic Monitoring is enabled by default, there are a couple of parameters that you need to be aware of.
The initialization parameter STATISTICS_LEVEL must be set either to TYPICAL or ALL to enable ADDM functionality. It defaults to TYPICAL; setting it to BASIC will disable ADDM and many other features.

Let check the below examples:

- To set the initialization parameter STATISTICS_LEVEL to TYPICAL or ALL. We need to follow the below step.
  alter system set statistics_level = typical;
  alter system set statistics_level = all;

- To set the initialization parameter STATISTICS_LEVEL to BASIC. We need to follow the below step.
  alter system set statistics_level = basic;

- To check the STATISTICS_LEVEL currently set.
  show parameter statistics_level;

There is a special ADDM-related task parameter, DBIOEXPECTED, which ADDM uses to analyze the performance of the I/O subsystem. The value for the DBIOEXPECTED parameter defines the average time it takes to read a single database block in microseconds. The default value for this parameter is 10 milliseconds (10,000 microseconds). We may change this default value.

Let's take an example, we want to change the default value to 30,000 microseconds. For this we need to execute the following as SYS user.

```sql
begin
    dbms_advisor.set_default_task_parameter (‘ADDM’,’DBIOEXPECTED’,30000);
end;
/
PL/SQL procedure successfully completed.
```

The value for the DBIOEXPECTED parameter is saved in an internal table. We need to execute the preceding procedure to change it. The following SQL script shows how we can check the current value of this parameter:
col parameter_name for a20
col parameter_value for a20
select advisor_name,
        parameter_name,
        parameter_value,
from   dba_advisor_def_parameters
where  parameter_name like 'DBIO%';

<table>
<thead>
<tr>
<th>ADVISOR_NAME</th>
<th>PARAMETER_NAME</th>
<th>PARAMETER_VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM</td>
<td>DBIOEXPECTED</td>
<td>30000</td>
</tr>
</tbody>
</table>

**Using Enterprise Manager to access ADDM**

The Oracle EM Database Control is the primary interface to the ADDM. The "Performance Analysis" section on the "Home" page is a list of the top five findings from the last ADDM analysis task.

In the above screenshot, it shows the number of ADDM performance findings for the last ADDM analysis period is shown under Diagnostic Summary toward the bottom right corner.
In the above screenshot, it shows the number of ADDM performance findings for the last ADDM analysis period is shown under Diagnostic Summary toward the bottom right corner.

**Running ADDM report manually**

Running the ADDM report from within the Oracle EM is the preferred and the simplest method. However, we can use Oracle-supplied scripts and package procedures to generate the ADDM diagnosis report. We need to know any two AWR snapshots to produce such a report. The snapshots must be available in AWR and there must not be any database restarts between those snapshots.

There are two scripts in the `$ORACLE_HOME/rdbms/admin` directory that can generate the ADDM diagnosis report: `admrpt.sql` and `admrpti.sql`. The former generates the report for the local database instance, while the latter can generate the report for other instances (in the RAC environment, for example). In addition, the Oracle-supplied package `DBMS_ADVISOR` has procedures (API) to generate the ADDM diagnosis report.

To run the scripts or use the API scripts, we must have the ADVISOR privilege.

When running `admrpt.sql`, we will be prompted to provide the beginning and ending snapshot ID from a list of available snapshots and a report name of our choice. Oracle will generate the ADDM diagnosis report for the specified range of snapshot IDs. The report can also be run in a non-interactive mode.
To generate the ADDM diagnosis report using the DBMS_ADVISOR package directly needs some more setup, as follows:

- Create an advisor task of ADDM type, using the CREATE_TASK procedure and after that we need to set the START_SNAPSHOT and END_SNAPSHOT parameters to run the task using the SET_TASK_PARAMETER procedure. The ADDM diagnosis report will be generated for the range of these snapshots.

```sql
BEGIN
   -- Create an ADDM task.
   DBMS_ADVISOR.create_task (  
       advisor_name => 'ADDM',  
       task_name => '970_1032_AWR_SNAPSHOT',  
       task_desc => 'Advisor for snapshots 970 to 1032.');

   -- Set the start and end snapshots.
   DBMS_ADVISOR.set_task_parameter (  
       task_name => '970_1032_AWR_SNAPSHOT',  
       parameter => 'START_SNAPSHOT',  
       value => 970);

   DBMS_ADVISOR.set_task_parameter (  
       task_name => '970_1032_AWR_SNAPSHOT',  
       parameter => 'END_SNAPSHOT',  
       value => 1032);

   -- Execute the task.
   DBMS_ADVISOR.execute_task(task_name => '970_1032_AWR_SNAPSHOT');
END;
```

- To Display the reports:

```sql
   -- Display the report.
   SET LONG 100000
   SET PAGESIZE 50000
   SELECT DBMS_ADVISOR.get_task_report('970_1032_AWR_SNAPSHOT') AS report  
       FROM dual;
   SET PAGESIZE 24
```
Information related to ADDM may be accessed in the DBA_ADVISOR_% views. There are other views as well to get more information as shown below.

- **DBA_ADVISOR_FINDINGS**: It provides information on the findings of ADDM.
- **DBA_ADVISOR_LOG**: It provides information on the current state of all tasks in the database.
- **DBA_ADVISOR_RATIONALE**: It provides information on the rationale for all recommendations by ADDM.
- **DBA_ADVISOR_RECOMMENDATIONS**: It provides information on the recommendations for all diagnostic tasks.
- **DBA_ADVISOR_TASKS**: It provides information on the existing tasks in the database.