

Oracle Resource Management

Stefan Seck

OPITZ CONSULTING Essen GmbH

Essen

Schlüsselworte

Oracle Resource Manager, Resource Plan, Instance Caging, Workload

Einleitung

In diesem Vortrag möchte ich das Resource Management in einer Oracle Datenbank vorstellen. Dabei wird es vor allem um folgende Fragen gehen:

Was kann Resource Management leisten?

Welche Schritte sind notwendig, um Resource Management erfolgreich zu nutzen?

Dabei liegt der Fokus auf dem grundlegenden Aufbau eines Resource Managements.

Resource Management - eine Geschichte

Seit Oracle 8i gibt es die Möglichkeit über das Package `dbms_resource_manager` bestimmten Usersession nicht alle Ressourcen eines Rechners zur Verfügung zu stellen.

Mit jeder neuen Oracle Version wurden die Möglichkeiten erweitert.

In Oracle 9i konnte der Resource Manager z.B. die vermutete Laufzeit der Transaktion berechnen und diese bei Überschreiten eines Schwellwertes abbrechen. Ebenso konnte die UNDO-Nutzung begrenzt werden.

Mit Oracle 10g ist es möglich Sessions automatisch bestimmten Consumer Groups zuzuordnen.

Und schließlich ist es seit Oracle 11g möglich Sessions bei der Nutzung von I/O zu begrenzen.

So lassen sich z.B. Sessions beenden, die eine festgelegte Menge MB gelesen haben.

In folgende Bereichen ist es möglich, Resourcemanagement zu betreiben.

- CPU-Nutzung
- Parallelität
- Session-Anzahl
- Undo-Nutzung
- Idle-Zeit
- Execution Zeit
- I/O Limit

Schritte zum Resouremangement

Um das Resourcemanagement einsetzen zu können, ist es notwendig sich zunächst Gedanken über das Konzept zu machen.

Es ist zu klären welche Applikation, greifen wann und wie auf die Datenbank zu.

Dabei werden im besonderen die folgenden drei Begriffe relevant.

- Resource Consumer Group
- Resource Consumer Mapping
- Resource Plan

Wenn alle Informationen zu den Applikationen in einer Datenbank gesammelt sind, lässt sich nun ein entsprechender Plan aufbauen.

Folgende Schritte für den Aufbau müssen durchgeführt werden:

1. Aufbau einer pending area
2. Erstellen einer resource consumer group
3. Erstellen eines resource plan
4. Erstellen einer plan directive
5. Prüfen der pending area
6. Submit der pending area

Mit einfachen PL/SQL Befehlen sieht das dann aus:

```
begin
  dbms_resource_manager.clear_pending_area;
  dbms_resource_manager.create_pending_area;

  dbms_resource_manager.create_plan ( -
    plan=>'TAGWERK', -
    comment=>'Tagesplan');
  dbms_resource_manager.create_plan_directive( -
    plan=>'TAGWERK', -
    group_or_subplan=>'SYS_GROUP', -
    cpu_p1=>100, -
    comment=>'SYS User erhalten hoechste Prio');
  dbms_resource_manager.create_plan_directive(-
    plan=>'TAGWERK', -
    group_or_subplan=>'OLTP', -
    cpu_p2=>100, -
    comment=>'OLTP User');

  dbms_resource_manager.validate_pending_area;
  dbms_resource_manager.submit_pending_area;
```

Die pending area dient als Arbeitsbereich, um Pläne und Direktiven zu erstellen.

Beim Validieren wird geprüft, ob der Plan zu Fehler führt. Ist dies nicht der Fall, schreibt Oracle Plan und Direktiven ins data dictionary.

Zuletzt muss er noch aktiviert werden:

```
ALTER SYSTEM SET RESOURCE_MANAGER_PLAN = TAGWERK';
```

Nun ist der Plan aktiv und kann mittel verschiedener v\$-Views überwacht werden.

Instance Caging

Seit Oracle 11g gibt es nun auch die Möglichkeit einer Datenbank nur einen Teil der vorhandenen CPU zur Verfügung zu stellen. Dazu nutzt man den init.ora Parameter cpu_count. Der Default steht auf (MAX-CPU-1). Werden z.B. 4 Datenbanken auf einem Rechner mit 4 CPU betrieben, so darf jede Datenbank 3 CPU voll benutzen. Wenn auf allen 4 Datenbanken viel Last geschieht, kann das dazu führen, das jede DB nur 25% der CPU benutzt.

Mittels Instance Caging kann die Anzahl der CPU nun auf das vorhandene Lastprofil der jeweiligen Datenbanken angepasst werden. Weniger wichtige Datenbanken erhalten so z.B. nur 1 CPU. Im Zusammenspiel mit dem Resource manager kann mit Hilfe von Resourceplänen diese CPU nun auf die einzelnen Session verteilt werden.

Zusammenfassung

Mit dem Resource management bietet Oracle eine gute Möglichkeit, die Last innerhalb einer Datenbank zu verteilen. Wenn Applikationen bestimmten Lastprofilen zuzuordnen sind und man diese entsprechend priorisieren kann, so setzt der Resource Manager die erstellten Pläne um. Wichtige Abfragen oder Anwendungen erhalten mehr Rechnerkapazitäten als weniger wichtige.

Kontaktadresse:

Stefan Seck
OPITZ CONSULTING Essen GmbH
Altendorfer Str. 3
D-45127 Essen

Telefon: +49 (0) 201 89 29 94 0
Fax: +49 (0) 201 89 29 94 4775
E-Mail stefan.seck@opitz-consulting.com
Internet: www.opitz-consulting.com