

Der Optimizer

Klaus Reimers
ORDIX AG
Paderborn

Schlüsselworte

Optimizer, Ausführungsplan, Statistiken, dbms_stats, Parameter, Hints, Stored Outline, SQL Baseline, SQL Profile, Parsen, Histogramme

Einleitung

Der Optimizer ist einer der zentralen Komponenten des gesamten Oracle-Systems. Hier wird entschieden, in welcher Form ein SQL Statement intern abgearbeitet wird. Sowohl der Administrator, als auch der Entwickler haben einige Möglichkeiten auf den letztendlich entstehenden Ausführungsplan einzuwirken.

In diesem Vortrag wird anhand vieler Beispiele (in Form von Demos) dargestellt, wie der Optimizer arbeitet, welche Möglichkeiten der Steuerung von außen existieren und die notwendigen Arbeiten des Administrators verdeutlicht.

Parsen

Immer wenn ein SQL Statement formuliert wird, muss es zunächst geparkt werden.

Das Parsen besteht immer aus drei Teilschritten:

- Syntaxcheck
- Semantikcheck
- Erstellung des Ausführungsplans

Das SQL Statement wird in der SGA (Shared Pool) abgelegt. Zunächst erfolgt im Rahmen des Syntaxchecks die Prüfung auf Korrektheit der Anweisung.

Im semantischen Check wird das SQL anschließend in seine Einzelheiten zerlegt. Hier fragt Oracle im Data Dictionary nach, ob die in der Anweisung angesprochen Tabellen und Spalten real verfügbar sind. Diese Nachfrage erfolgt natürlich auf durch intern gebildete SQL-Abfragen, die sogenannten rekursiven SQLs.

Die Hauptarbeit des Parsen ist die Erstellung des Ausführungsplans. Hier errechnet der Optimizer den vermeintlich optimalen Ausführungsplan. Dazu stellen werden dem Optimizer verschiedene Daten zur Verfügung gestellt, die in sein Rechenmodell als Inputs eingehen. Eine mögliche Quelle sind Parameter, eine weitere stellen Statistiken dar.

Parameter

Die Parameterdatei (`init.ora` / `spfile`) wird grundsätzlich beim Starten der Instanz eingelesen. In Oracle 12c gibt es 366 offizielle Parameter, die fast alle modifiziert werden können. Allerdings muss bei 115 Parameter die Instanz durchgestartet werden. Etwa 25 bis 30 Parameter haben einen Einfluss auf den Denkprozess des Optimizer.

Nachfolgend sind die wichtigsten Parameter, die den Optimizer beeinflussen, aufgeführt:

NAME	VALUE
optimizer_features_enable	12.1.0.1
optimizer_mode	ALL_ROWS
optimizer_index_cost_adj	100
optimizer_index_caching	0
optimizer_dynamic_sampling	2
optimizer_secure_view_merging	TRUE
optimizer_use_pending_statistics	FALSE
optimizer_capture_sql_plan_baselines	FALSE
optimizer_use_sql_plan_baselines	TRUE
optimizer_use_invisible_indexes	FALSE
optimizer_adaptive_reporting_only	FALSE
optimizer_adaptive_features	TRUE

Die jeweilige Wirkungsweise der wichtigsten Parameter wird in den Demos verdeutlicht.

Statistiken

Mit dem PL/SQL-Package DBMS_STATS können Statistiken für den kostenbasierenden Optimizer generiert und verwaltet werden. Diese Statistiken können sowohl im Data Dictionary, als auch im Benutzer-Schema gespeichert werden.

Mit diesem Paket können die folgenden Funktionen ausgeführt werden:

- Parallele Generierung von Statistiken
- Erstellung individueller Statistiken
- Erstellen und Löschen von benutzerdefinierten Statistiktabelle
- Austausch von Statistiken zwischen dem Benutzerschema und dem Data Dictionary
- Statistiken zwischen Datenbanken austauschen
- Sperren von Statistiken
- Wiederherstellung von Statistiken
- Parametrierung der Sammlung von Statistiken

Entscheidend ist hier die konzeptionelle Frage, wann und in welcher Tiefe die Statistiken aktualisiert werden sollen. Grundsätzlich sind zwei verschiedene Vorgehensweisen zum Sammeln von Statistiken möglich.

Da neue Statistiken immer die Gefahr von kippenden Ausführungsplänen mit sich bringen, werden auf einigen Datenbanken keine neuen Statistiken erzeugt. Die Pläne sind dann stabil. Allerdings birgt diese Methode die Gefahr der schleichenden Verschlechterung, da auf veränderte Bedingungen nicht reagiert werden kann.

Die meisten Administratoren sammeln daher regelmäßig neue Statistiken. Seit der Version 9i bietet Oracle hier ein Verfahren an, bei dem alle die Tabellen mit neuen Statistiken versorgt werden, bei denen mehr als 10% der Sätze verändert worden sind. Ab Oracle 10g ist hierfür ein automatisierter Task hinterlegt.

Preferences

Seit Oracle 11g kann man individuell je Tabelle die Anforderung an die Art der Statistikerstellung einstellen. Für diesen Zweck wurden die Preferences eingeführt.

Folgende Preferences sind modifizierbar:

cascade	Definiert, ob bei einer neuen Tabellenstatistik auch die an der Tabelle hängenden Indizes mit gepflegt werden sollen.
degree	Grad der Parallelität
estimate_percent	Definition der Stichprobe in Prozent
method_opt	Umgang mit Histogrammen
no_invalidate	Definiert, ob offene Cursor invalidiert werden sollen
granularity	Umgang mit Partitionen
publish	Zeitpunkt der Veröffentlichung
incremental	Globale Statistiken auf partitionierten Tabellen
stale_percent	Legt den Grad fest, ab wann eine Tabelle als STALE definiert wird

Hints

Um dem Optimizer zur Ausführung eines SQL Statement Instruktionen zu übermitteln, wie er dieses Statement behandeln soll, können s.g. Hints (Hinweise) übergeben werden.

Diese Hinweise werden in Form eines Kommentars in das Statement eingefügt und gelten immer nur für diesen einen Befehl. Ein Hinweis muss dabei nach einem der Schlüsselwörter SELECT, UPDATE, INSERT oder DELETE folgen.

Werden Hints nicht korrekt angegeben oder ist der Hint technisch nicht durchführbar, so wird keine Fehlermeldung ausgegeben, sondern der Hint einfach ignoriert.

Gespeicherte Ausführungspläne

Ist ein SQL sehr komplex oder ein Ausführungsplan nicht stabil, so kann dieser in der Datenbank hart verdrahtet werden.

Hierfür bietet Oracle drei Möglichkeiten:

- Stored Outlines
- SQL Profiles
- SQL Baselines

Diese Varianten sind nicht in allen Oracle-Versionen verfügbar. Zudem ist die Nutzung teilweise kostenpflichtig.

Stored Outlines

Mit Hilfe der Stored Outlines hat man die Möglichkeit, Ausführungspläne in der Datenbank fest zu speichern. Damit ist gewährleistet, dass unabhängig vom momentanen Stand der Statistiken immer mit dem gleichen Ausführungsplan gearbeitet wird.

Wenn auf einer Tabelle ein neuer Index aufgesetzt wird, müssen evtl. die abgelegten Stored Outlines geändert werden, da dieser Index sonst nicht greift.

Mit Oracle 10g sind die Outlines offiziell aus der Wartung entfernt worden, sind aber intern noch nutzbar.

SQL-Profile

Der Optimizer durchdenkt alle theoretisch möglichen Ausführungspläne und errechnet auf Basis der Input-Daten (Parameter und Statistiken) den vermeintlich optimalen Plan. Ist ein SQL nun sehr komplex, so tendiert die Anzahl der möglichen Ausführungspläne gegen unendlich.

Damit die Parse-Phase nicht unnötig lang wird, bricht der Optimizer den Denkprozess nach einigen Millisekunden ab und es wird der bis zum Abbruch günstigste Plan verwendet. Dieser Plan muss nicht zwangsläufig der optimale Plan sein.

Im erweiterten Modus (Tuning Mode) durchläuft der Optimizer folgende Tests:

- Statistikanalyse
- SQL Profiling
- Access-Path-Analyse
- SQL-Structure-Analyse

Der Tuning Optimizer prüft zunächst, ob Statistiken vorhanden sind und ob diese auch aktuell sind. Stehen keine aktuellen Statistiken zur Verfügung, sammelt der Optimizer diese während der Optimierung. Dies kann natürlich eine gewisse Zeit in Anspruch nehmen.

Beim Profiling führt der Optimizer zahlreiche Tests durch. Hierzu gehören z.B. die Analyse verschiedener Optimizer-Modi (*all_rows* vs. *first_rows*) und das Sampling von Daten.

Weiterhin vergleicht der Optimizer unterschiedliche Ausführungspläne bzgl. der Kosten. Akzeptiert der Benutzer das vorgeschlagene Profil, so wird dieses in der Datenbank gespeichert und für spätere Ausführungen genutzt.

SQL Baselines

Bei der Verwendung dieser Methode werden die Ausführungspläne in *gute* und *schlechte* Pläne unterteilt. Es wird somit garantiert, dass kein als schlecht markierter Plan mehr ausgeführt wird. Darüber hinaus können Baselines als Fixum erstellt werden. Neue Ausführungspläne werden immer an dieser fixierten Basis gemessen und bewertet.

Fazit

In dieser Zusammenfassung sind einige wichtige Themen rund um den Optimizer angerissen worden. Im Vortrag werden viele dieser Aspekte in Demoform näher beleuchtet.

Kontaktadresse:

Klaus Reimers
ORDIX AG
Westermauer 12-16
D-33098 Paderborn

Telefon: +49 (0) 5251 / 1063-0
Fax: +49 (0) 180 1 67349 0
E-Mail: info@ordix.de
Internet: www.ordix.de