

Quo Vadis SQLTXPLAIN: SQL-Statement-Tuning 2017

Clemens Bleile

dbi services

Rue de la Jeunesse 2, CH-2800 Delémont

Schlüsselworte

SQLTXPLAIN, SQLT, SQLd360, SQL Tuning

Einleitung

Während meiner Jahre im Oracle Support als einer der Leiter des Oracle Performance-Teams EMEA (2008 - 2015) war das Werkzeug SQLTXPLAIN das Mittel zur Wahl um die Ursache für einen suboptimalen Ausführungsplan zu ermitteln. Wie sieht es heute (zu Zeiten von Oracle 12.2. oder bald Oracle 18) aus? Mein ehemaliger Kollege Mauro Pagano propagiert sein Werkzeug SQLd360 als zeitgemäßere Alternative. Der Vortrag ist eine subjektive Betrachtung des Werkzeugs SQLTXPLAIN und seiner Alternativen.

SQLTXPLAIN bzw. SQLT

Im Jahre 2007 stellte Carlos Sierra, damals Mitglied von Oracles „Center of Expertise“, das Werkzeug SQLTXPLAIN vor. Er hatte es im Laufe seiner Beratungstätigkeiten entwickelt und stellte es dann der Öffentlichkeit zur Verfügung. SQLTXPLAIN sollte alle Diagnose-Informationen rund um ein zu optimierendes SQL-Statement zur Verfügung stellen: Globale Informationen (z.B. System-Statistiken, dbms_stats-Setup, etc.), Pläne, Plan-Kontroll-Informationen (z.B. SQL Profile), Tabellen- und Index-Informationen (z.B. Spalten-Statistiken), Cursor-Sharing und Bind-Variablen, Ausführungs-Statistiken (z.B. aus Active Session History oder SQL-Statistiken), Objekt-Informationen (z.B. Abhängigkeiten und Metadaten) und (falls lizenziert) SQL Tuning Advisor-Informationen. Diese Informationen sollten über Hyper-Links in einem HTML-Report verknüpft sein um schnell und gezielt auf die Daten zugreifen zu können. Im Laufe der Zeit entwickelte sich SQLTXPLAIN zu **DEM** Werkzeug des Performance-Teams im Oracle Support, um Kundenanfragen bezüglich des Optimizers in kürzester Zeit beantworten zu können. Aber auch bei Beratern und DBAs entwickelte sich SQLT zum beliebten Werkzeug wenn es um SQL-Tuning ging. Man war nicht länger auf eine Sammlung von Skripten angewiesen um ein SQL-Statement analysieren zu können. Desweiteren stellte SQLT hilfreiche Funktionen bereit, um z.B.

- einen Ausführungsplan in einer separaten Umgebung (ohne Daten in den Tabellen) reproduzieren zu können
- bestimmte Pläne für SQL-Statements einfach festzulegen (mit einem SQL Profile, SQL Plan Baseline, Outline oder SQL Patch)
- SQL-Statements auf aktiven Standby-DBn zu untersuchen
- die Auswirkung von möglichen Optimizer-Einstellungen (dokumentierte und undokumentierte Parameter und Fix Controls) auf die Planerstellung zu testen
- eine 10053-Trace-Datei auswerten zu können
- 10046-(SQL-)Trace-Informationen auswerten zu können

Neben all den Funktionen führte SQLT auch noch „Health Checks“ durch und gab Hinweise auf Besonderheiten, die den Plan negativ beeinflussen können (z.B. „unusable Indexes“) oder Hinweise bezüglich der Möglichkeiten dem Optimizer mehr Informationen zukommen zu lassen (z.B. Spalte ist ein Kandidat für ein NOT NULL Constraint).

Für Manager das wohl wichtigste Attribut von SQLT war, dass es nichts kostete – einen Oracle-Support-Vertrag mit „My Oracle Support“-Zugriff vorausgesetzt.

Global

- [Observations](#)
- [SQL Text](#)
- [SQL Identification](#)
- [Environment](#)
- [CBO Environment](#)
- [Fix Control](#)
- [CBO System Statistics](#)
- [DBMS_STATS Setup](#)
- [Initialization Parameters](#)
- [NLS Parameters](#)
- [I/O Calibration](#)
- [Tool Configuration Parameters](#)

Cursor Sharing and Binds

- [Cursor Sharing](#)
- [Adaptive Cursor Sharing](#)
- [Peeked Binds](#)
- [Captured Binds](#)

SQL Tuning Advisor

- [STA Report](#)
- [STA Script](#)

Plans

- [Summary](#)
- [Performance Statistics](#)
- [Performance History \(delta\)](#)
- [Performance History \(total\)](#)
- [Execution Plans](#)

Plan Control

- [Stored Outlines](#)
- [SQL Patches](#)
- [SQL Profiles](#)
- [SQL Plan Baselines](#)
- [SQL Plan Directives](#)

SQL Execution

- [Active Session History](#)
- [AWR Active Session History](#)
- [SQL Statistics](#)
- [SQL Detail ACTIVE Report](#)
- [Monitor Statistics](#)
- [Monitor ACTIVE Report](#)
- [Monitor HTML Report](#)
- [Monitor TEXT Report](#)
- [Segment Statistics](#)
- [Session Statistics](#)
- [Session Events](#)
- [Parallel Processing](#)

Tables

- [Tables](#)
- [Statistics](#)
- [Statistics Extensions](#)
- [Statistics Versions](#)
- [Modifications](#)
- [Properties](#)
- [Physical Properties](#)
- [Constraints](#)
- [Columns](#)
- [Indexed Columns](#)
- [Histograms](#)
- [Partitions](#)
- [Indexes](#)

Objects

- [Objects](#)
- [Dependencies](#)
- [Fixed Objects](#)
- [Fixed Object Columns](#)
- [Nested Tables](#)
- [Policies](#)
- [Audit Policies](#)
- [Tablespaces](#)
- [Metadata](#)

Abb. 1: Beispiel der Startseite eines SQLTXPLAIN-Reports mit direkten Links in die unterschiedlichen Bereiche

Method	Script
SQLT XTRACT	sqlt/run/sqltextract.sql
SQLT XECUTE	sqlt/run/sqltexecute.sql
SQLT XTRXEC	sqlt/run/sqlttrxecsql
SQLT XPLAIN	sqlt/run/sqltxplain.sql
SQLT XTRSBY	sqlt/run/sqltxtrsbysql

Abb. 1: Haupt-Methoden von SQLTXPLAIN

Im Laufe der Jahre entwickelte sich SQLTXPLAIN gezwungenermaßen (mit den zusätzlichen Funktionen neuer Oracle-Versionen) weiter (z.B. Informationen über SQL Plan Direktiven kamen in 12cR1 hinzu). Aber die „adaptiven“ Funktionen des Optimizers (adaptive Pläne, adaptive Statistiken) bereitete den Entwicklern von SQLT auch einige Probleme, da die dynamische Änderung von Plänen schwer darstellbar ist und sich die Analyse der Ursache einer „Fehlfunktion“ des Optimizers erschwerte. Es gab folgerichtig auch kritische Stimmen, die meinten, dass SQLTXPLAIN zu wenig Laufzeit-Informationen liefere, die andere DB-Funktionen wie SQL Monitoring oder Plan-Informationen in „Active Session History“ inzwischen liefern konnten. Die Installation von SQLT in Produktiv-Umgebungen war schon immer schwierig, da zusätzliche Schemata notwendig sind für die man oft keine Produktions-Genehmigung bekam. In diesem Zusammenhang ein Kommentar aus einem Blog:

„I like very much Tanel Poder snapper and Carlos Sierra SQLTXPLAIN . They represent valuable performance diagnostic tools. Unfortunately I am still waiting to find a customer site where I will be allowed or granted necessary privileges to install and to use them. There are client sites where I have been asked to tune queries without having the possibility to execute dbms_xplan.display_cursor. Let alone installing SQLTXPLAIN under SYS user or having grant select on x\$ tables.“
 (<https://hourim.wordpress.com/2013/11/18/sqltxplain-under-oracle-12c/>)

Ausserdem sei der Haupt-HTML-Report von SQLTXPLAIN für komplexe SQL-Statements zu groß und dann nur schwer navigierbar (Navigation auf der HTML-Seite im Browser zu langsam). Mauro Pagano, ein ehemaliger Oracle-Kollege, der nach Carlos Sierras Weggang von Oracle einige Zeit für

SQLTXPLAIN zuständig war, entwickelte deshalb im Jahr 2013 das Werkzeug SQLd360. Einige Unzulänglichkeiten von SQLT sollten adressiert werden:

- Grafische Darstellung von Informationen – speziell die zeitliche Entwicklung von Metriken und Statistiken
- Es sollte keine Modifikation auf der zu analysierenden DB notwendig sein
- Jede Information im Haupt-HTML-Report sollte in separate HTML-Dateien zur schnelleren Navigation ausgelagert werden
- Jeder sollte Zugriff auf das Werkzeug haben, d.h. Zugriff auf das Werkzeug auch ohne Oracle-Support-Vertrag und somit ohne “My Oracle Support“-Zugriff

Mauro hielt auch einige Präsentationen in den vergangenen Jahren zum Thema SQLd360 (unter anderem auf der DOAG-Konferenz 2016).

Wie sieht es heute mit SQLTXPLAIN aus? Ist es noch **DAS** Mittel der Wahl wenn es um SQL-Tuning bei Oracle-DBn geht? Oder ist der Optimizer inzwischen schon so weit fortgeschritten, dass DBAs die Informationsflut von SQLT gar nicht mehr benötigen? Vor allem, gibt es inzwischen bessere Alternativen?

Es sind solche Fragen, die sich diese Präsentation widmen will:

- Ist SQLTXPLAIN noch zeitgemäss?
- Wie sollten SQL-Tuning-Probleme 2017 adressiert werden?
- Ist SQLd360 eine Alternative?

Zur besseren Veranschaulichung werden kleinere Demonstrationen durchgeführt.

Kontaktadresse:

Clemens Bleile
dbi services
Rue de la Jeunesse 2
CH-2800 Delémont

Telefon: +41 32 422 96 00
Mobil: +41 78 677 51 09
Fax: +41 32 422 96 15
E-Mail: clemens.bleile@dbi-services.com
Internet: www.dbi-services.com