

Leonid Nossov
Hanno Ernst
Victor Chupis

Formal SQL Tuning for Oracle Databases

Practical Efficiency - Efficient Practice

 Springer

Formal SQL Tuning for Oracle Databases

Leonid Nossov • Hanno Ernst •
Victor Chupis

Formal SQL Tuning for Oracle Databases

Practical Efficiency - Efficient Practice

 Springer

Leonid Nossov
ORACLE Deutschland B.V. & Co. KG
München, Germany

Hanno Ernst
T-SYSTEMS INTERNATIONAL GMBH
Bamberg, Germany

Victor Chupis
Vodafone GmbH
Düsseldorf, Germany

Translation from the German language edition “Formales SQL-Tuning für Oracle-Datenbanken”. © Springer-Verlag 2016.

Translated from the German by Jane Scolah and David Thackray.

Graphical Illustrations by Anna Nosova.

ISBN 978-3-662-50416-1

ISBN 978-3-662-50417-8 (eBook)

DOI 10.1007/978-3-662-50417-8

Library of Congress Control Number: 2016950905

© Springer-Verlag Berlin Heidelberg 2016

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer-Verlag GmbH Berlin Heidelberg

I know that I know nothing
Socrates

Foreword by Watson

The mantra I chant relentlessly is “you must understand your data; you must understand your queries.” By this I mean that if a developer has a complete understanding of the information he is processing, he will be able to write code that will run efficiently. The critical decisions that Oracle’s cost-based optimizer must make are join order, join method, and access method. Use of structures such as `DISTINCT` to remove duplicates when none exist, or `OUTER JOIN` when `INNER JOIN` will do, or needless functions around predicate columns, or inappropriate use of `NULL` will force the CBO to develop plans that do not make the best decisions and therefore cripple performance. There are more complex issues such as whether views are mergeable or predicate pushable, whether correlated subqueries can be factored out into common table expressions—any number of other optimizations. The CBO (cost based optimizer) will do its best, but even though it is probably the most complex software artifact with which you will ever work, all it can do is follow rules. It does not have the knowledge of what is actually possible given the business environment that a developer will (ideally) have.

If the developer has a thorough understanding of his data and queries, he will be able to write code that lets the optimizer make the best decisions. However, this knowledge may take an immense amount of time to acquire, and the necessary information may not be available. Certainly an outside consultant on a short-term contract has no chance of gaining it, and all too often the DBA (database administrator) cannot do so either. The methodology presented in this book will allow any competent DBA to tune SQL, even if he has minimal knowledge of the environment. That is the power of this method: You do NOT need to have a comprehensive understanding of the data or the SQL. You do not even need to know what the SQL being executed is or be able to rewrite it.

The approach taken in this book is vital to the division of responsibilities in an Oracle environment. Historically, Oracle’s approach to SQL tuning was that the DBA should identify the statements that have (or are causing) problems and throw them back to the developer for tuning. However, all too often when asked to tune a statement, a developer would reply “how do I do that?” and as a result many DBAs spend 8 hours a day tuning SQL. In later releases of the database, Uncle Oracle has realized this, and now Oracle’s approach appears to be that developers should concentrate on writing code that fulfills a business need and DBAs should be

responsible for making it run efficiently. The formal method described will help DBAs to do just that.

The step-by-step method will identify points in an execution plan that are problematic for performance and suggest how to tune them. Comprehensive instruction on how to capture, read, and interpret an execution plan is essential for this, and the book delivers this in spades. Essential reading for all DBAs.

Oracle Certified Master DBA
Director of Database Services
Skillbuilders Inc.
Wakefield
RI, USA
June 2016

John Watson

Foreword by Gosejacob

Leonid Nossov's current work deals specifically with the central aspect of performance tuning for Oracle databases, the analysis and acceleration of SQL statements. I was, of course, delighted when Leonid asked me to write a few introductory lines for this book too.

I can well imagine some potential readers being somewhat deterred by the title "Formal SQL Tuning," which sounds very dry and theoretical. In my view, however, it best reflects the nature of the method described. The term "formal tuning" can also be seen here as a synonym for a structural approach.

The current version of the Oracle database provides a comprehensive arsenal of analysis possibilities and tools, which simplify, or even automate, the tuning of SQL statements. However, situations repeatedly occur in which these possibilities can only be exploited to a limited extent or, in some cases, not at all. One of my colleagues jokingly calls this the expert mode. We regularly encounter disbelieving faces when this method achieves amazing runtime improvements. The colleague in question has now been awarded the title Dr. SQL by some clients.

Particularly when swift action and, hopefully, positive results are expected, stress levels for DBAs can increase sharply. From my own experience, I remember situations in which the DBA was trying to solve performance problems in front of the screen at the keyboard while the rest of the room was filled with a dozen people, including the production manager. The consistent formal approach presented in this book can be a lifeline in such a situation.

Leonid and his co-authors succeed in serving up this rather dry material in an easily digestible and appetizing manner in the lively dialogues with Peter Smith, with whom I, as a reader, can easily identify. This is made possible by the use of numerous examples which present the material under consideration in a clear manner and make the relevant information directly accessible to the reader. Possible solutions for the diagnosed problem are supplied at the same time.

Let me congratulate you Leonid on another fine book, and I wish you the reader an enjoyable reading experience and the time to learn formal SQL tuning at your leisure, so that you can shine with this knowledge when the next critical situation comes up.

Munich, Germany
August 2015

Martin Gosejacob

Foreword by Schwinn

I must admit that this is the first time I have held one of Leonid Nossov's books in my hands. I became aware of this book through my colleague Martin, as I myself am often confronted with questions about SQL tuning or database monitoring. SQL tuning is also "trendy" and has long been a popular topic in the database community. Almost anyone who is anyone in this field has a blog on the subject of SQL tuning. Especially after the release of new database patches or even a new version, there is an increase in the amount of information on the Internet about new optimization methods with associated tips and tricks.

Consequently, I was very curious about this book by Leonid and his co-authors Victor Chupis and Hanno Ernst with the title "Formal SQL Tuning," which was so different from other books and articles I had read. I had no idea what to expect. As a mathematician, I was, of course, familiar with formal methods. Mathematical methods are often regarded as very "dry" and of little interest to nonscientists. I was, therefore, all the more surprised by the relaxed, easily readable "question and answer" style which Leonid uses in his book. This enables one to familiarize oneself with the material quickly and with a minimum of effort. If you really read every chapter—even those designated for beginners—you can even learn how execution plans are interpreted. In this way, database administrators and developers get a chance to gain positive, personal, hands-on experience of tuning tasks even though they are not SQL tuning experts and do not possess a lot of previous knowledge about data modeling. It is then also interesting to read what co-authors Victor Chupis and Hanno Ernst have to say. They put the formal SQL tuning method to the test and successfully solve real-life problems.

The three authors are very successful in focusing on the essentials of SQL tuning. Equipped with formal SQL tuning methodology, one can then calmly and confidently cope with changes to the database due to new optimizer releases or in-house application changes. I hope all readers enjoy this book, which I heartily recommend.

Munich, Germany
10 August 2015

Ulrike Schwinn