

University of Wollongong - Research Online

Thesis Collection

Title: Adaptive reorganization of database structures through dynamic vertical partitioning of relational tables

Author: Liu Zhenjie

Year: 2007

Repository DOI:

Copyright Warning

You may print or download ONE copy of this document for the purpose of your own research or study. The University does not authorise you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site.

You are reminded of the following: This work is copyright. Apart from any use permitted under the Copyright Act 1968, no part of this work may be reproduced by any process, nor may any other exclusive right be exercised, without the permission of the author. Copyright owners are entitled to take legal action against persons who infringe their copyright. A reproduction of material that is protected by copyright may be a copyright infringement. A court may impose penalties and award damages in relation to offences and infringements relating to copyright material.

Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.

Unless otherwise indicated, the views expressed in this thesis are those of the author and do not necessarily represent the views of the University of Wollongong.

Research Online is the open access repository for the University of Wollongong. For further information contact the UOW Library: research-pubs@uow.edu.au

2007

Adaptive reorganization of database structures through dynamic vertical partitioning of relational tables

Liu Zhenjie
University of Wollongong

Follow this and additional works at: <https://ro.uow.edu.au/theses>

University of Wollongong

Copyright Warning

You may print or download ONE copy of this document for the purpose of your own research or study. The University does not authorise you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site.

You are reminded of the following: This work is copyright. Apart from any use permitted under the Copyright Act 1968, no part of this work may be reproduced by any process, nor may any other exclusive right be exercised, without the permission of the author. Copyright owners are entitled to take legal action against persons who infringe their copyright. A reproduction of material that is protected by copyright may be a copyright infringement. A court may impose penalties and award damages in relation to offences and infringements relating to copyright material.

Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.

Unless otherwise indicated, the views expressed in this thesis are those of the author and do not necessarily represent the views of the University of Wollongong.

Recommended Citation

Liu, Zhenjie, Adaptive reorganization of database structures through dynamic vertical partitioning of relational tables, MCompSc thesis, School of Information Technology and Computer Science, University of Wollongong, 2007. <http://ro.uow.edu.au/theses/33/>

NOTE

This online version of the thesis may have different page formatting and pagination from the paper copy held in the University of Wollongong Library.

UNIVERSITY OF WOLLONGONG

COPYRIGHT WARNING

You may print or download ONE copy of this document for the purpose of your own research or study. The University does not authorise you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site. You are reminded of the following:

Copyright owners are entitled to take legal action against persons who infringe their copyright. A reproduction of material that is protected by copyright may be a copyright infringement. A court may impose penalties and award damages in relation to offences and infringements relating to copyright material. Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.



Adaptive Reorganization of Database Structures Through Dynamic Vertical Partitioning of Relational Tables

A thesis submitted in fulfillment of the
requirements for the award of the degree

Master of Computer Science by Research

from

UNIVERSITY OF WOLLONGONG

by

Zhenjie Liu

School of Information Technology and Computer Science
October 2007

© Copyright 2007

by

Zhenjie Liu

All Rights Reserved

*Dedicated to
My parents and wife Yanting*

Declaration

This is to certify that the work reported in this thesis was done by the author, unless specified otherwise, and that no part of it has been submitted in a thesis to any other university or similar institution.

Zhenjie Liu
October 23, 2007

Publications

Zhenjie Liu and Janusz R. Getta. Optimization of query processing through constrained vertical partitioning of relational tables. In *DBA'06: Proceedings of the 24th IASTED international conference on Database and applications*, pages 221-227, Anaheim, CA, USA, 2006. ACTA Press.

Abstract

Performance tuning of relational database systems is always a challenging task for database administrator. Automated performance tuning has been proposed recently as a new approach to detect and to eliminate performance problems and to support the decisions of database administrators.

This work considers one of the techniques used in automated performance tuning, dynamic vertical partitioning. Dynamic vertical partitioning of relational tables is one of the ways in which the physical structures of a relational database can be reorganised automatically in order to improve the performance of future database applications. The thesis presents how dynamic vertical partitioning can be used for the comprehensive analysis and optimisation of an adaptive reorganisation of database structures. In particular, we propose the algorithms to use to predict the future workload of the system, to analyse the characteristics of the workload, and to find a near optimal vertical partitioning of relational tables. Then, we discuss the implementation aspects of vertical partitioning with the materialized view and index-based techniques.

Our contributions to automated performance tuning of relational database systems can be summarised as follows:

1. Propose a cost model to perform a detailed analysis of the costs of query and data manipulation processing over a given configuration of a relational database;
2. Propose a new algorithm for vertical partitioning of relational schemas in a database system with a given level of redundancies for a given workload;
3. Discuss the limitations of static vertical partitioning and propose dynamical vertical partitioning;
4. Discuss the characteristics of workload in order to predict the future workloads of the system;

5. Implementation aspects of vertical partition discussion: materialized view based and index based;
6. Discussion of the implementation of a vertical partition as a virtual view;
7. Conduct experiments to confirm the correctness of the cost model used by the vertical partitioning algorithm and demonstrate the expected performance gains from the partitioning

Acknowledgements

The research work for this thesis was undertaken at the University of Wollongong.

Firstly, I would like to thank my supervisor, Dr. Janusz R. Getta for his guidance and help in my research. Without him, this thesis would not have been possible.

Also, I wish to acknowledge the support I have received from all the staff in the School of IT & CS, University of Wollongong.

Finally, I would like to thank my family and friends for their enduring love and support.

Contents

Publications	v
Abstract	vi
Acknowledgements	viii
1 Introduction	1
1.1 Database Performance Tuning	1
1.1.1 Overview of Automatic Database Administration	1
1.1.2 Automatic Query Performance Tuning	2
1.2 Partitioning	3
1.2.1 Horizontal Partitioning	3
1.2.2 Vertical Partitioning	5
1.2.3 Comparison of Horizontal and Vertical partitioning	5
1.3 The Challenge	6
1.4 Strategy of Solution	7
1.5 Thesis Organisation	7
2 Technical Backgrounds	9
2.1 Autonomic Database Management Systems	9
2.2 Automatic Performance Optimisation	11
2.2.1 Performance Measurement Methodologies	11
2.2.2 Automatic Performance Monitoring and Diagnosis	11
2.2.3 Automatic Performance Tuning	12
2.3 Dynamic Query Optimisation	13
2.3.1 Query Execution Optimisation	13
2.3.2 Physical database structure Reorganization	13
2.4 Static Vertical Partitioning	14

2.4.1	Non-Overlapping Partitioning	14
2.4.2	Overlapping Partitioning	14
2.4.3	Finding interesting partitions	15
2.5	Dynamical Vertical Partitioning	16
2.5.1	Workload Monitoring and Analyzing	16
2.5.2	Implementing Vertical Partition	17
2.6	Summary	17
3	Static partitioning	18
3.1	Preliminaries	18
3.2	Cost Model	21
3.3	Partition Configuration	23
3.3.1	Algorithm overview	23
3.3.2	Choosing the partitions	24
3.3.3	Recycling of the original tables	25
3.3.4	Merging the partitions	26
3.4	Experiments	27
3.4.1	Evaluation of partitioning	27
3.4.2	Trend of Storage and Benefit	28
3.5	Summary and open problem	29
4	Dynamic Partitioning	31
4.1	Workload Patterns	31
4.2	Preliminaries	32
4.3	Cost Model	35
4.4	Overview of Dynamic Partitioning Algorithm	35
4.5	Potential Low Workload Time Detecting	37
4.6	Finding Most Similar Workload	38
4.6.1	Function to calculate similarity of two signatures	38
4.6.2	Most Similar Workload Searching	39
4.7	Finding configuration	39
4.7.1	Choosing partitions	39
4.8	Merging the partitions	41
4.9	Partition Implementation	41
4.10	Comparison between Static and Dynamic partitioning	42
4.11	Summary	42

5	Implementation of Dynamic Vertical Partitioning	44
5.1	Issues of implementing partitions	44
5.1.1	Materialized View-based Implementation	45
5.1.2	Index based Implementation	46
5.2	Comparison of Index-based and Materialized view-based	47
5.2.1	Consistency Control Comparison	47
5.2.2	Storage Cost Comparison	48
5.2.3	Comparison by Time Cost	49
5.3	Open problems	50
5.4	Experiments	51
5.4.1	Query Rewrite of Materialized View	51
5.4.2	Fast Full Index Scan	51
5.4.3	Comparison of Materialized View and Index	52
5.4.4	View Update	52
5.5	Summary	53
6	Conclusions and Future work	56
6.1	Conclusions	56
6.2	Future work	57
	Bibliography	59

List of Tables

List of Figures

1.1	Principle of horizontal partitioning	4
1.2	Principle of vertical partitioning	6
3.1	Outline of algorithms	24
3.2	Storage and performance	28
3.3	Performance comparison	28
4.1	Total workload	33
4.2	Example of session and signature	34
4.3	Workload before dynamic partitioning	36
4.4	Workload after dynamic partitioning	36
4.5	Overview of dynamic partitioning	37
5.1	Example of Materialized View	46
5.2	Horizontal traversal of index	47
5.3	Query Rewrite of Materialized View	52
5.4	Fast Full Index Scan	52
5.5	Comparison of MV and Index	53
5.6	View Update	54