

# **Index Structures for Data Warehouses**

Dissertation  
am  
**Fachbereich für Mathematik und Informatik**  
an der  
**Freien Universität Berlin**

eingereicht am 22.12.1999

von

**Marcus Jürgens**

**Betreuer**

Prof. Dr. Hans-Joachim Lenz (Freie Universität Berlin)  
Prof. Dr. Heinz Schweppe (Freie Universität Berlin)  
Prof. Dr. Johann Christoph Freytag (Humboldt-Universität zu Berlin)

Tag der mündlichen Prüfung: 16. Feb. 2000

to Chris



## Abstract

This thesis investigates which index structures support query processing in typical data warehouse environments most efficiently. Data warehouse applications differ significantly from traditional transaction-oriented operational applications. Therefore, the techniques applied in transaction-oriented systems cannot be used in the context of data warehouses and new techniques must be developed.

The thesis shows that the time complexity for the computation of *optimal* tree-based index structures prohibits its use in real world applications. Therefore, we *improve* heuristic techniques (e. g.  $R^*$ -tree) to process range queries on aggregated data more efficiently. Experiments show the benefits of this approach for different kinds of typical data warehouse queries. Performance models *estimate* the behavior of standard index structures and the behavior of the extended index structures. We introduce a new model that considers the distribution of data. We show experimentally that the new model is more precise than other models known from literature. Two techniques *compare* two tree-based index structures with two bitmap indexing techniques. The performance of these index structures depends on a set of different parameters. Our results show which index structure performs most efficiently depending on the parameters.

## Acknowledgements

I am very grateful to have had the opportunity to write my Ph.D. Thesis under the supervision of Professor Hans-Joachim Lenz who brought the area of data warehouses to my attention. In countless meetings he gave me helpful feedback. I would like to thank Professor Heinz Schweppe for his constructive suggestions and the invitation to cooperate with the database group at the Freie Universität Berlin. Professor Freytag supported me with beneficial ideas and outstanding comments.

The graduate school in Distributed Information Systems would not be possible in this efficient form without its speaker Professor Oliver Günther. His commitment made this school a constructive and pleasant environment.

I would like to express my thanks to all members of database groups participating in this graduate school for their interesting and encouraging talks and discussions. In particular, I am grateful for the constructive discussions with Agnès Voisard and Annika Hinze. The *Deutsche Forschungsgemeinschaft (DFG)* supported me as a fellowship recipient. Professor Joseph Bronstad and Leslie Hazelwood did not resign trying to correct my English.

# Contents

<b>1. Introduction</b>	<b>1</b>
1.1. Goals	2
1.2. Outline	3
<b>2. State of the Art of Data Warehouse Research</b>	<b>5</b>
2.1. Introduction	5
2.2. Traditional transaction-oriented systems	5
2.3. Data warehouses for decision support	7
2.4. OLAP vs. OLTP	9
2.5. Accelerating query speed	10
2.5.1. Denormalized schemas	10
2.5.2. Materialized views	11
2.5.3. No locking	13
2.5.4. On-line aggregation	13
2.5.5. Index structures	13
2.6. Summary	14
<b>3. Data Storage and Index Structures</b>	<b>15</b>
3.1. Introduction	15
3.2. Memory hierarchy	15
3.3. Mechanics of disks	16
3.4. Data space and queries	18
3.4.1. Data space	18
3.4.2. Queries	18
3.5. Tree-based indexing	19
3.5.1. Top-down, bottom-up, and bulk loading	20
3.5.2. Point quadtrees	21
3.5.3. kd-tree	22
3.5.4. kdb-tree	22
3.5.5. R-tree	22
3.5.6. R*-tree	23
3.5.7. Other relatives of the R-tree family and other tree structures	24
3.5.8. Generic tree structures	26
3.6. Bitmap indexing	27

3.6.1.	Standard bitmap indexing . . . . .	27
3.6.2.	Multi-component equality encoded bitmap index . . . . .	29
3.6.3.	Range-based encoding . . . . .	31
3.6.4.	Multi-component range-based encoding . . . . .	32
3.6.5.	Other compression techniques / combination of bitmaps and trees . . . . .	33
3.7.	Arrays . . . . .	33
3.8.	Summary . . . . .	34
<b>4.</b>	<b>Mixed Integer Problems for Finding Optimal Tree-Based Index Structures</b>	<b>35</b>
4.1.	Introduction . . . . .	35
4.2.	Optimization problem parameters . . . . .	35
4.3.	Mapping into a mixed integer problem . . . . .	36
4.4.	Problem complexity . . . . .	38
4.5.	Model evaluation . . . . .	39
4.6.	Summary . . . . .	41
<b>5.</b>	<b>Aggregated Data in Tree-Based Index Structures</b>	<b>43</b>
5.1.	Introduction . . . . .	43
5.2.	“Fit for aggregation” access method . . . . .	47
5.3.	Materialization of data . . . . .	48
5.4.	Modified operations . . . . .	50
5.4.1.	Insert operation . . . . .	50
5.4.2.	Delete operation . . . . .	51
5.4.3.	Update operation . . . . .	51
5.4.4.	Creating index structures, bottom-up index structures . . . . .	51
5.4.5.	Point query algorithm . . . . .	52
5.4.6.	Range query algorithm . . . . .	52
5.5.	Storage cost . . . . .	52
5.6.	Height of tree . . . . .	54
5.7.	Overlaps of regions . . . . .	56
5.8.	Experiments . . . . .	57
5.8.1.	Cost model . . . . .	57
5.8.2.	Physical index structure . . . . .	58
5.8.3.	Implementation . . . . .	58
5.8.4.	Generation of test data . . . . .	58
5.8.5.	Query profile . . . . .	60
5.8.6.	Results of experiments . . . . .	60
5.9.	Summary . . . . .	62
<b>6.</b>	<b>Performance Models for Tree-Based Index Structures</b>	<b>63</b>
6.1.	Introduction . . . . .	63
6.2.	Fit for modeling . . . . .	63



6.3.	Performance models for access leaf nodes . . . . .	64
6.3.1.	GRID model . . . . .	64
6.3.2.	SUM model . . . . .	66
6.3.3.	Equivalence of GRID model and SUM model . . . . .	67
6.3.4.	FRACTAL model . . . . .	69
6.3.5.	Equivalence between FRACTAL model, SUM model, and GRID model . . . . .	71
6.4.	PISA model . . . . .	71
6.5.	Computational Efficiency of SUM model and PISA model . . . . .	74
6.6.	Adapting PISA model to different distributions . . . . .	76
6.6.1.	Uniformly distributed data . . . . .	76
6.6.2.	Skewed data . . . . .	77
6.6.3.	Normally distributed data . . . . .	79
6.7.	Model evaluation . . . . .	80
6.7.1.	Uniformly distributed data . . . . .	81
6.7.2.	Skewed data . . . . .	81
6.7.3.	Normally distributed data . . . . .	84
6.8.	PISA model for dependent data . . . . .	84
6.9.	Extension of models . . . . .	85
6.10.	Applications of models . . . . .	86
6.10.1.	Savings of $R^*_a$ -tree depending on the query box size and form	86
6.10.2.	Savings of $R^*_a$ -tree depending on the number of dimensions	86
6.11.	Summary . . . . .	87
<b>7.</b>	<b>Techniques for Comparing Index Structures</b>	<b>89</b>
7.1.	Introduction . . . . .	89
7.2.	Experimental parameters . . . . .	89
7.2.1.	Data specific parameters . . . . .	89
7.2.2.	Query specific parameters . . . . .	90
7.2.3.	System specific parameters . . . . .	91
7.2.4.	Disk specific parameters . . . . .	91
7.2.5.	Configuration . . . . .	92
7.3.	Index structures and time estimators . . . . .	93
7.3.1.	Time Measures for tree-based index structures . . . . .	93
7.3.2.	Time measures for bitmap indexing techniques . . . . .	94
7.4.	Classification trees . . . . .	96
7.4.1.	Applied methods . . . . .	97
7.4.2.	Value sets of Parameters . . . . .	97
7.4.3.	Results . . . . .	99
7.5.	Statistics in two dimensions . . . . .	102
7.5.1.	Sum aggregation . . . . .	103
7.5.2.	Median aggregation . . . . .	103
7.5.3.	Count aggregation . . . . .	103
7.5.4.	Results . . . . .	106

---

7.6. Summary . . . . .	111
<b>8. Conclusion and Outlook</b>	<b>113</b>
<b>Bibliography</b>	<b>116</b>
<b>Index</b>	<b>125</b>
<b>A. List of Symbols</b>	<b>127</b>
<b>B. Approximation of PISA Model</b>	<b>133</b>
<b>Ph D Related Material</b>	
Zusammenfassung der Ergebnisse . . . . .	135
Lebenslauf . . . . .	136
Verwendete Hilfsmittel . . . . .	137