

Table of Contents

CHAPTER 1 *Introduction*

1.1	ORDBMS: The Next Great Wave	1
1.2	Extensible DBMS	2
1.3	Overview	3

CHAPTER 2 *Background on User-Defined Routines*

2.1	User-Defined Routines	5
2.2	Definition, Implementation, and Execution of New UDR	6
2.2.1	User-Defined Scalar Functions	7
2.2.2	User-Defined Aggregate Functions	9
2.2.3	User-Defined Table Functions	10
2.2.4	User-Defined Functions and Large Objects	11
2.3	Comparison with Stored Procedures	12
2.4	Optimization of Queries with UDF	12

CHAPTER 3 *Parallel Processing of User-Defined Functions*

3.1	Introduction	14
3.2	Limits of Current ORDBMS	15
3.3	Parallel Processing of UDF	17
3.3.1	Two Step Parallel Aggregation of UDAF	17
3.3.2	Partitioning Classes and Partitionable Functions	18
3.3.3	Parallel Sorting as a Preprocessing Step for UDAF	21
3.3.4	Extended Syntax for Function Registration	22
3.4	Example Applications	24
3.4.1	The UDAF Most_Frequent	24
3.4.2	The UDSF Running_Average	25
3.4.3	The UDAF Median	25
3.4.4	Further Applications	26
3.5	Plausibility Considerations Regarding Performance	28
3.6	Related Work	30
3.7	Summary	31

CHAPTER 4 *Intra-function Parallelism*

4.1	Introduction	33
4.2	Compose/Decompose Operators for Intra-function Parallelism	34
4.2.1	Compose/Decompose Operators	34
4.2.2	Extensibility of Compose Operators by Combine Functions	36
4.2.3	Application of Intra-function Parallelism	37
4.2.4	Intra-function Parallelism for Function Pipelines	38
4.3	Experimental Performance Study	39
4.3.1	Experimental Scenario and Implementation	39
4.3.2	Performance Results	41
4.4	Related Work	43
4.5	Summary	44

CHAPTER 5 *The Multi-operator Method*

5.1 Introduction 45

5.2 Performance Problems with Complex UDF in Current ORDBMS 46

5.2.1 The PBSM Algorithm as a Sophisticated UDP Implementation 47

5.3 The Multi-operator Method as a New Technique to Implement Complex UDF 49

5.3.1 The Multi-operator Method and Its Benefits 49

5.3.2 A Multi-operator Implementation of the PBSM Algorithm 51

5.4 Supporting the Multi-operator Method 53

5.4.1 Executing Query Execution Plans 53

5.4.2 Example for a Textual Specification of Query Execution Plans 55

5.4.3 Parallel Evaluation 55

5.5 Performance Evaluation 56

5.5.1 Experimental Scenario 56

5.5.2 Performance Results 62

5.6 Related Work 64

5.7 Summary 65

CHAPTER 6 *User-Defined Table Operators*

6.1 Introduction 67

6.2 User-Defined Table Operators 68

6.2.1 A Generalization Relationship for Row Types 68

6.2.2 Defining and Implementing UDTO 69

6.2.3 The Different Usages of the UDTO Concept 74

6.2.4 Parallel Processing of Procedural UDTO 77

6.2.5 Extension to Multiple Output Tables 80

6.3 Example Applications for UDTO 81

6.3.1 Computing a Spatial Join 81

6.3.2 Different UDTO for the Same Predicate 85

6.3.3 Computing the Median: An Aggregation Operator 89

6.3.4 A UDTO for a Complex Aggregation 90

6.3.5 Association Rule Mining 94

6.4 Related Work 101

6.5 Summary and Conclusions 102

CHAPTER 7 *Implementation of UDTO*

7.1 Introduction 106

7.2 The MIDAS Prototype 106

7.2.1 Architectural Overview 107

7.2.2 Query Compilation and Execution 108

7.2.3 The MIDAS System Tables 111

7.2.4 UDSF in MIDAS 112

7.3 Implementation of SQL Macros 113

7.3.1 DDL Statements 113

7.3.2 SQL Macro Expansion in DML Statements 115

7.3.3 Expanding SQL Macros in Preprocessors and Middleware 116

7.4	Implementation of Procedural UDTO	123
7.4.1	Extensions to the SQL Compiler	123
7.4.2	Extensions to the Optimizer and the Parallelizer	125
7.4.3	Extensions to the Scheduler	126
7.4.4	Extensions to the Execution Engine	126
7.4.5	Extensions to Transaction Management	128
7.4.6	Implementation of Input and Output Tables	131
7.5	Optimization Issues for UDTO	134
7.5.1	UDTO and Implied Predicates	134
7.5.2	Estimating Costs and Selectivity of UDTO	135
7.5.3	Application of Traditional Optimization Rules	137
7.6	Using UDTO to Generate Alternative Execution Plans for UDF	138
7.7	Evaluation of the Implementation	139
7.7.1	Evaluation of SQL Macros	140
7.7.2	Evaluation of Procedural UDTO	142
7.8	Summary	144

CHAPTER 8 Summary, Conclusions, and Future Work

8.1	Summary	145
8.2	Conclusions	146
8.3	Future Work	149

References

References	151
------------------	-----

Appendix A

A.1	The Program <code>sequential_invert</code>	157
A.2	The Program <code>parallel_invert</code>	158
A.3	The Query Execution Plan for the Spatial Join with SQL Macro	159



<http://www.springer.com/978-3-540-42781-0>

New Concepts for Parallel Object-Relational Query
Processing

Jaedicke, M.

2001, XII, 163 p., Softcover

ISBN: 978-3-540-42781-0