

УДК 004.652+004.655+004.657

Д. Б. БУЙ¹, В. Г. СКОБЕЛЕВ²¹ *Київський національний університет ім. Тараса Шевченка, г. Київ, Україна*² *Інститут прикладної математики і механіки НАН України, г. Донецьк, Україна*

СЛОЖНОСТЬ ОПЕРАЦИЙ В БАЗАХ ДАННЫХ (ОБЗОР)

В работе исследовано современное состояние проблемы обеспечения эффективности функционирования систем управления базами данных (СУБД) с позиции моделей и методов, предназначенных для оптимизации запросов. На основе анализа особенностей процесса проектирования СУБД, основных моделей данных, основных понятий, используемых при оценке сложности алгоритмов, показано, что в этом контексте следует рассматривать именно связку «БД–СУБД». Охарактеризованы основные типы математических моделей, предназначенных для теоретического анализа эффективности функционирования связки «БД–СУБД» в терминах анализа сложности операций. Рассмотрена эволюция методов «оптимизации запросов» (являющихся, по своей сути, эвристическими методами).

Ключевые слова: базы данных, СУБД, оптимизация запросов

Введение

Известно, что основной формой организации информации в современных информационных системах (ИС) являются связки «база данных (БД) – система управления базой данных (СУБД)».

Парадигма БД сформировалась в середине 60-х годов XX столетия в результате многочисленных попыток избавиться от недостатков файловых систем организации информации, возникающих при многопользовательском доступе к данным. Основные недостатки хорошо известны [1]: изолированность данных; зависимость программ от данных; дублирование данных; отсутствие описания данных.

Естественным средством устранения этих недостатков является отделение процесса хранения данных от процесса их обработки, что и реализуется связкой «БД–СУБД». Отметим, что ядром БД является модель данных, т.е. совокупность структур данных и операций их обработки. Проблема проектирования связок «БД–СУБД» является одной из центральных в процессе разработки ИС, так как от ее решения зависит качество создаваемой ИС. Этой проблеме посвящены усилия многочисленных исследователей во всем мире (см., напр., [1-6]).

Основными требованиями, предъявляемыми к связкам «БД–СУБД» являются их безопасность и эффективность функционирования на протяжении всего жизненного цикла. Проблеме обеспечения безопасности посвящены многочисленные исследования, и в настоящее время в ее решении достигнуты значительные успехи [7]. Совершенно иная ситуация имеет место с проблемой обеспечения эффективности функционирования связок «БД–СУБД».

На практике, как правило, разработчики руководствуются принципом «мы делаем так, как умеем», не проводя детального теоретического анализа этой проблемы в конкретной ситуации.

Целью настоящего обзора и является попытка охарактеризовать современное состояние проблемы анализа эффективности функционирования связок «БД–СУБД».

1. Особенности процесса проектирования связки «БД–СУБД»

Процесс проектирования связки «БД–СУБД» состоит из следующих 4-х этапов: системный анализ предметной области, построение инфологической модели предметной области, построение логической модели, построение физической модели.

Результатом процесса проектирования связки «БД–СУБД» является готовая к заполнению реальная БД и готовое к использованию ПО.

Таким образом, в связке «БД–СУБД» выделяются инфологический, логический и физический уровни. Первые два уровня называются внешними и предназначены для поддержки санкционированного доступа к данным БД. Третий уровень называется внутренним и отображает организацию данных в среде хранения.

С позиции логической модели многообразие типов связок «БД–СУБД» в значительной мере определяется используемыми моделями данных. В настоящее время основными являются следующие модели данных: иерархическая модель [8], сетевая модель [9], реляционная модель [10], объектно-

ориентированная модель [11], объектно-реляционная модель [12], многомерная модель [13].

Естественно выделяются следующие два подхода к анализу эффективности обработки запросов. Предметом исследования первого подхода является математическая модель связки «БД–СУБД». Цель состоит в построении нижних оценок асимптотической временной сложности функционирования связки «в наихудшем случае» или «в среднем». Второй подход основан на исследовании реальной связки «БД–СУБД». Как правило, он осуществляется методами статистического анализа. Цель состоит в оценке реального времени обработки запросов.

2. Анализ сложности алгоритмов

В процессе работы любой алгоритм преобразования информации обрабатывает некоторые входные данные того или иного «размера», характеризующего «объем памяти», необходимой для их хранения. Два подхода к определению «размера» хорошо известны [14]: использование равномерного либо логарифмического весов.

Анализ временной сложности алгоритма при его работе над входом фиксированного размера при равномерном весе, состоит в оценке числа выполненных элементарных операций, а при логарифмическом весе – в оценке времени, затраченного на все сработавшие элементарные операции. При анализе временной сложности обработки запросов связкой «БД–СУБД» существенную роль играют операции ввода-вывода. Поэтому при таком анализе более адекватным является использование логарифмического веса.

Выделяют три типа временной сложности алгоритма при обработке данных фиксированного размера: сложность в наихудшем случае, сложность для почти всех входов и сложность в среднем. Исследование временной сложности алгоритма в наихудшем случае существенно проще, чем исследование его сложности для почти всех входов или сложности в среднем. При анализе временной сложности обработки запросов связкой «БД–СУБД» наиболее приемлемым является использование сложности в среднем. Именно она дает возможность оценить эффективность обработки запросов связкой при условиях, определенных в техническом задании.

Вычисление временной сложности алгоритма (за исключением тривиальных случаев) приводит к громоздким, мало обозримым формулам. Для того чтобы избежать такой ситуации, оценивают порядок роста этой сложности при неограниченном росте размера входных данных. Такая оценка называется *асимптотической временной сложностью алгоритма*.

Для теоретического анализа эффективности обработки запросов в процессе эксплуатации связки «БД–СУБД» наиболее приемлемым является вычисление асимптотической временной сложности в среднем при использовании логарифмического веса.

В современных связках «БД–СУБД» (особенно предназначенных для работы с хранилищами данных) используются специальные библиотеки функций, в основе которых лежит обработка списков.

3. Математические модели связок «БД–СУБД»

Формальный подход к теоретическому анализу алгоритмов преобразования информации основан на симбиозе математической логики, теории моделей и прикладной теории алгоритмов. Такой подход состоит в следующем [15-20].

Над фиксированным множеством M конечных структур строится теория T . Множество формул этой теории представляет собой некоторый язык L . Основными являются следующие три задачи:

1) проверка выполнимости (satisfiability checking), состоящая в том, чтобы ответить на вопрос: существует ли модель $S \in M$, для которой формула $\psi \in L$ выполняема?

2) проверка модели (model checking), состоящая в том, чтобы ответить на вопрос: верно ли, что для модели $S \in M$ и формулы $\psi \in L$ формула ψ истинна в модели S ?

3) эволюция запроса (query evolution), состоящая в том, чтобы для модели $S \in M$ и формулы $\psi(\bar{x}) \in L$ (где \bar{x} – множество свободных переменных) вычислить множество $\{\bar{a} \mid S \models \psi(\bar{a})\}$.

Входными данными для проверки модели являются $S \in M$ и $\psi \in L$. Поэтому естественно возникают следующие задачи:

1) как выражается сложность решения задачи проверки модели через сложность модели $S \in M$ и формулы $\psi \in L$?

2) какова сложность вычисления множества $\{\psi \in L \mid S \models \psi\}$ для заданной модели $S \in M$?

3) какова сложность вычисления множества $\{S \in M \mid S \models \psi\}$ для заданной формулы $\psi \in L$?

В случае анализа связки «БД–СУБД» построение теории T осуществляется над некоторой алгеброй $A = (\Xi, B)$, где Ξ – множество структур данных, а B – множество базовых алгоритмов их обработки. Алгебра A представляет собой основу для построения некоторого множества (конечных) структур D . Объектом исследования является язык запросов L , а предметом исследования – анализ

разрешимости и выразительности языка L [21-24], анализ временной сложности алгоритмов, представленных формулами языка L [25-31], а также выделение тех классов формул языка L, которые представляют запросы, реализуемые с полиномиальной временной сложностью [32-36].

Такой подход исследовался в деталях для основных моделей данных: иерархической [37-40], сетевой [41-45] и реляционной [46-60].

4. «Алгоритмы оптимизации» запросов

Одной из основных сложных задач, связанных с появлением связок «БД–СУБД», является «оптимизация обработки» запросов СУБД. Слово «оптимизация» является математическим термином и подразумевает выбор наилучшего объекта или алгоритма, принадлежащего формально определенному классу, соответственно, объектов или алгоритмов. Ясно, что такой подход неприменим ни с теоретической, ни с практической точки зрения к связке «БД–СУБД». Поэтому под «оптимизацией обработки» запросов СУБД понимают стратегии, предназначенные для повышения эффективности процедур обработки запросов. Такие стратегии представляют собой эвристические методы, целесообразность использования которых обосновывается статистическими методами, т.е. обработкой результатов проведенных экспериментов.

Для того, чтобы говорить об «оптимизации обработки» запросов, прежде всего требуется понимать, что понимается под «стоимостью» запроса. Выделяют следующие составляющие этого понятия: коммуникационная стоимость, стоимость доступа к вторичной памяти, стоимость запоминания, стоимость вычисления.

Таким образом, «оптимизация обработки» запросов представляет собой нетривиальную многокритериальную задачу.

Методы «оптимизации обработки» запросов, разработанные в [61-66], привели к пониманию необходимости разработки полномасштабного унифицированного подхода к решению этой задачи, основанного на нисходящем подходе (top-down approach). Анализ состояния этих исследований на 1984 г. представлен в обзоре [67].

В течение последнего 20-летия XX столетия сформировались следующие три направления исследования задачи «оптимизация обработки» запросов: разработка методов «оптимизации обработки» последовательности запросов [68-81], разработка методов лексической и семантической «оптимизации» запросов. [82-95], разработка методов «оптимизации обработки» запросов в параллельной распределенной связке «БД–СУБД» [96-111].

Заключение

В настоящей работе сделана попытка охарактеризовать современное состояние проблемы исследования сложности операций в связках «БД–СУБД». Анализ источников показывает, что современное состояние этой проблематики весьма далеко от того, что принято называть проработанной технологией.

Литература

1. Исаченко, А. Н. Модели данных и СУБД [Текст] / А. Н. Исаченко, С.П. Бондаренко. – Минск : Изд-во БГУ, 2007. – 205 с.
2. Ульман, Дж. Основы систем баз данных [Текст] / Дж. Ульман. – М. : Финансы и статистика, 1983. – 334 с.
3. Мейер, Д. Теория реляционных баз данных [Текст] / Д. Мейер. – М. : Мир, 1987. – 540 с.
4. Редько, В. Н. К основаниям теории реляционных моделей баз данных [Текст] / В. Н. Редько, Д. Б. Буй // Кибернетика и системный анализ. – 1996. – № 4. – С. 3-12.
5. Дейт, К. Введение в системы баз данных [Текст] / К. Дейт. – М. : Вильямс, 2001. – 1072 с.
6. Кудрявцев, К. Я. Создание бах данных [Текст] / К. Я. Кудрявцев. – М. : НИЯУ МИФИ, 2010. – 155 с.
7. Database security [Text] / S. Castano, M.G. Fugini, G. Martello, et al. – N.Y. : Addison Wesley Publishing Company, 1995. – 456 p.
8. Tsichritzis, D. C. Hierarchical data-base management: a survey [Text] / D. C. Tsichritzis, F. H. Lochovsky // ACM Computing. Survey. – 1976. – № 8. – P. 67-103.
9. Bachman, C. W. Software for random access processing [Text] / C. W. Bachman // Datamation. – 1965. – № 11. – P. 36-41.
10. Codd, E. F. a relationship model for large stored date banks [Text] / E. F. Codd // Communications of the ACM. – 1970. – № 13. – P. 377-387.
11. Грэхем, И. Объектно-ориентированные методы: принципы и практика [Текст] / И. Грэхем. – М. : Вильямс, 2004. – 880 с.
12. Chamberlin D. Anatomy of an object-relational database [Text] / D. Chamberlin // DB2 Magazine. – 1996. – № 1. – P. 24-37.
13. Pedersen, T. B. A foundation for capturing and querying complex multidimensional data [Text] / T. B. Pedersen, C. S. Jensen, C. E. Dyerson // Information Systems. – 2001. – № 5. – P. 383-423.
14. Кормен, Т. Алгоритмы: построение и анализ [Текст] / Т. Кормен, Ч. Лейзерсон, Р. Ривест. – М. : МНЦМО, 2000. – 960 с.
15. Aho, A. Universality of data retrieval languages [Text] / A. Aho, J. Ullman // Proc. of the 6th ACM Symposium on Principles of Programming Languages. – 1979. – P. 110-117.

16. Abiteboul, S. *A transaction language for data base update and specification [Text]* / S. Abiteboul, V. Vianu // *Proc. of the 6th ACM Symposium on Principles of Database Systems.* – 1987. – P. 260-268.
17. Derougemont, M. *Computability and databases [Text]* / M. Derougemont // *Technological Science Information.* – 1987. – № 8. – P. 609-622.
18. Abiteboul, S. *Datalog extensions for database queries and updates [Text]* / S. Abiteboul, V. Vianu // *Journal of Computer and Systems Sciences.* – 1991. – № 1. – P. 62-124.
19. Kanellakis, P. *Constraint query languages [Text]* / K. Kanellakis, G. Cupper, P. Revesz // *Journal of Computer and System Sciences.* – 1995. – № 1. – P. 26-52.
20. Bradfield, J. *Modal logic and mu-calculi [Текст]* / J. Bradfield, C. Stirling // *In: Handbook of process algebra.* – Elsevier, 2001. – P. 293-332.
21. Arvind, V. *Expressibility of 1st order logic with a non-deterministic inductive operator [Text]* / V. Arvind, S. Biswas // *LNCS.* – 1987. – V. 247. – P. 323-335.
22. Shmueli, O. *Decidability and expressiveness aspects of logic queries [Text]* / O. Shmueli // *Proc. of the 6th ACM Symposium on Principles of Database Systems.* – 1987. – P. 237-249.
23. Shauble, P. *On the expressive power of query languages [Text]* / P. Shauble, B. Wuthrich // *Theory of Information Systems.* – 1994. – № 1. – P. 69-91.
24. Замятин, А. П. *Элементы математической теории информационных систем: выразимость и вычислимость [Текст]* / А. П. Замятин, А. Б. Ливчак. – Екатеринбург : УрГУ, 1996. – 104 с.
25. Тайцлин, М. А. *Сравнение выразительной силы некоторых языков запросов для баз данных [Текст]* / М. А. Тайцлин // *Труды математического института им. В.А. Стеклова.* – 2011. – Т. 274. – С. 297-313.
26. Gurevich, Y. *Towards logic tailored for computational complexity [Text]* / Y. Gurevich // *Lecture Notes in Mathematics.* – 1984. – V. 1104. – P. 175-216.
27. Itai, A. *Unification as complexity measure for logic programming [Text]* / A. Itai, J. Makowsky // *Journal of Logic Programming.* – 1987. – № 4. – P. 105-117.
28. Vanbenthen, J. *Time, logic and computation [Text]* / J. Vanbenthen // *LNCS.* – 1989. – V. 354. – P. 1-49.
29. Varvel, D. L. *The computational completeness of extended database query languages [Text]* / D. L. Varvel, L. Shapiro // *IEEE Transactions on Software Engineering.* – 1989. – № 1. – P. 632-638.
30. Abiteboul, S. *Fixpoint logics, relational machines, and computational complexity [Text]* / S. Abiteboul, M. Y. Vardi, V. Vianu // *Proc. of the 7th Annual Conference on Structure in Complexity Theory.* – 1992. – P. 156-158.
31. Gradel, E. *Capturing complexity classes by fragments of the second-order logic [Text]* / E. Gradel // *Theoretical Computer Science.* – 1992. – № 1. – P. 35-57.
32. Dziembowski, S. *Bounded-variable fixpoint queries are PSPACE-complete [Text]* / S. Dziembowski // *LNCS.* – 1996. – V. 1258. – P. 89-105.
33. Fagin, R. *Generalized first-order spectra and polynomial-time recognizable sets [Text]* / R. Fagin // *SIAM-AMS Processing.* – 1974. – V. 7. – P. 43-73.
34. Gurevich, Y. *Algebras of feasible functions [Text]* / Y. Gurevich // *Proc. of the 24th Symposium on Foundations of Computer Science.* – 1983. – P. 210-214.
35. Afrati, F. *On Datalog vs polynomial time [Text]* / F. Afrati, S. S. Cosmadakis, M. Yannakakis // *Proc. of the 10th ACM Symposium on Principles of Database Systems.* – 1991. – P. 13-25.
36. Hella L. *Logical hierarchies in PTIME [Text]* / L. Hella // *Proc. of the 7th IEEE Symposium on Logic in Computer Science.* – 1992. – P. 360-368.
37. Gottlob, G. *Datalog LITE: A deductive query language with linear time model checking [Text]* / G. Gottlob, E. Gradel, H. Veith // *ACM Transactions on Computational logic.* – 2002. – № 3. – P. 42-79.
38. Arnold, A. *The mu-calculus alternation-depth is strict on binary trees [Text]* / A. Arnold // *RAIRO Informatique Theorique at Applications.* – 1999. – № 2. – P. 329-339.
39. Dahlhaus, E. *Query languages for hierarchical databases [Text]* / E. Dahlhaus, J. A. Makowsky // *Information and Computation.* – 1992. – № 1. – P. 1-32.
40. Dublish, P. *Query languages which express all P-time queries for trees and unicyclic graphs [Text]* / P. Dublish, S. N. Maheswari // *LNCS.* – 1987. – V. 452. – P. 246-253.
41. Grohe, M. *Definability and descriptive complexity on databases of bounded tree-width [Text]* / M. Grohe, J. Marino // *LNCS.* – 1999. – V. 1540. – P. 70-82.
42. Courselle, B. *The monadic second order logic of graphs II: infinite graphs of bounded width [Text]* / B. Courselle // *Mathematical Systems Theory.* – 1989. – № 1. – P. 187-221.
43. Cay, J. *An optimal lower bound on the number of variables for graph identification [Текст]* / J. Cay // *Combinatorica.* – 1992. – № 12. – P. 389-410.
44. Courselle, B. *The monadic second order logic of graphs IX: machines and their behaviors [Text]* / B. Courselle // *Theoretical Computer Science.* – 1995. – V. 151. – P. 125-162.
45. Causal, D. *On infinite transition graphs having a decidable monadic theory [Text]* / D. Causal // *LNCS.* – 1996. – V. 1099. – P. 194-205.
46. Courselle, B. *Monadic second order logic, graph coverings and unfolding of transition systems [Text]* / B. Courselle, I. Walukiewicz // *Annals of Pure and Applied Logic.* – 1998. – № 1. – P. 35-62.
47. Paradaens, J. *On the expressive power of relational algebra [Text]* / J. Paradaens // *Information Processing Letters.* – 1978. – № 2. – P. 107-111.

48. Banchilhon, F. On completeness of query languages for relational databases [Text] / F. Banchilhon // LNCS. – 1978. – V. 64. – P. 119-123.
49. Chandra, A. Computable queries for relational data bases [Text] / A. Chandra, D. Harel // Journal of Computer and System Sciences. – 1980. – № 1. – P. 156-178.
50. Бениаминов, Е. М. Алгебраический подход к моделям баз данных реляционного типа [Текст] / Е. М. Бениаминов // Семиотика и информатика. – 1980. – Вып. 14. – С. 44-80.
51. Борщев, Б. В. Логический подход к описанию реляционных баз данных [Текст] / Б. В. Борщев // Семиотика и информатика. – 1981. – Вып. 16. – С. 78-122.
52. Ливчак, А. Б. Реляционные модели баз данных и полиномиальная вычислимость [Текст] / А. Б. Ливчак // НТИ. Сер. 2. Информационные процессы и системы. – 1981. – № 6. – С. 28-29.
53. Vardi, M. Complexity of relational query languages [Text] / M. Vardi // Proc. of the 14th ACM Symposium on the Theory of Computing. – 1982. – P. 137-146.
54. Chandra, A. Structure and complexity of relational queries [Text] / A. Chandra, D. Harel // Journal of Computer and System Sciences. – 1982. – № 1. – P. 99-128.
55. Ливчак, А. Б. Полиномиальные запросы к реляционным базам данных [Текст] / А. Б. Ливчак // Программирование. – 1985. – № 2. – С. 66-72.
56. Immerman, N. Relational queries computable in polynomial time [Text] / N. Immerman // Information and Control. – 1986. – № 1. – P. 86-104.
57. Ливчак, А. Б. Сложностный анализ реляционных языков [Текст] / А. Б. Ливчак // Вопросы кибернетики. – 1989. – Вып. 149. – С. 67-85.
58. Барышников, А. Ю. Языки PQL и FO+LFP эквивалентны и без наличия порядка [Текст] / А. Ю. Барышников, А. Б. Ливчак // Известия Вузов. Математика. – 1996. – № 4. – С. 18-23.
59. Ливчак, А. Б. Недетерминированные запросы к реляционным базам данных [Текст] / А. Б. Ливчак, А. Я. Овсянников // НТИ. Сер. 2. Информационные процессы и системы. – 1994. – № 7. – С. 19-23.
60. Ливчак, А. Б. Усиление теорем о полиномиальных запросах [Текст] / А. Б. Ливчак // Программирование. – 1995. – № 3. – С. 9-16.
61. Palermo, F. P. A data base search problem [Text] / F. P. Palermo // Proc. of the 4th Symposium on Computer and Information Science. – 1972. – P. 67-101.
62. Astrahan, M. M. Implementation of structured English query language [Text] / M. M. Astrahan, D. D. Chamberlin // Communications of ACM. – 1975. – № 10. – P. 580-588.
63. Niebuehr, K. L. Algorithm for processing query by example [Text] / K. L. Niebuehr, K. W. Scholtz, S. E. Smith // IBM Technical Disclosure Bulletin. – 1976. – № 2. – P. 736-741.
64. Wong, E. Decomposition – a strategy for query processing [Text] / E. Wong, R. Youssefi // ACM Transaction on Database Systems – 1976. – № 3. – P. 223-241.
65. Schenk, K. L. An algorithm for servicing multi-relational queries [Text] / K. L. Schenk, J. R. Pinkert // Proc. of the International Conference on Management of Data. – 1977. – P. 10-20.
66. Makinouchi, A. The optimization strategy for query evaluation in RDB/V1 [Text] / A. Makinouchi, M. Tezuka, H. Kitakami, S. Adachi // Proc. of the 7th International Conference on Very Large Databases. – 1981. – P. 518-529.
67. Jarke, M. Query optimization in database systems [Text] / M. Jarke, J. Koch // Computing Surveys. – 1984. – № 2. – P. 111-152.
68. Finelstein, S. Common expression analysis in database application [Text] / S. Finelstein // Proc. of the International Conference on Management of Data. – 1982. – P. 235-245.
69. Park, J. Using common sub-expressions to optimize multiple queries [Text] / J. Park, A. Segev // Proc. of the 4th International Conference on Data Engineering. – 1988. – P. 311-319.
70. Sellis, T. Multiple query optimization [Text] / T. Sellis // ACM Transactions on Database Systems. – 1988. – № 1. – P. 23-52.
71. Sellis, T. On the multi query optimization problem [Text] / T. Sellis, S. Ghosh // IEEE Transactions on Knowledge and Data Engineering. – 1990. – № 6. – P. 262-266.
72. Cosar, A. Multiple query optimization with depth-first branch-and-bound and dynamic query ordering [Text] / A. Cosar, E.-P. Lim, J. Srivastava // Proc. of the International Conference on Information and Knowledge Management. – 1993. – P. 433-438.
73. Graefe, G. Extensibility and search efficiency in the Volcano optimizer generator [Text] / G. Graefe, W. G. McKenna // Technical Report CU-CS-91-563. Univ. of Colorado at Boulder, 1991.
74. Harinarayan, V. Implementing data cubes efficiently [Text] / V. Harinarayan, A. Rajaraman, J. Ulman // Proc. of the International Conference on Management of Data. – 1996. – P. 205-216.
75. Index selection for OLAP [Text] / H. Gupta, V. Harinarayan, A. Rajaraman, J. Ulman // Proc. of the International Conference on Data Engineering. – 1997. – P. 208-219.
76. Gupta, H. Selection of views to materialize in a data warehouse [Text] / H. Gupta // Proc. of the International Conference on Database Theory. – 1997. – P. 453-470.
77. Pellenkoft, A. The complexity of transformation-based join enumeration [Text] / A. Pellenkoft, C. A. Galindo-Legaria, M. Kersten // Proc. of the 23rd International Conference on Very Large Databases. – 1997. – P. 306-315.
78. Rao, J. Reusing invariants: a new strategy for correlated queries [Text] / J. Rao, D. Reiner // Proc. of

the International Conference on Management of Data. – 1998. – P. 37-48.

79. Subramanian, S. N. Cost based optimization of decision support queries using transient views [Text] / S. N. Subramanian, S. Venkataraman // Proc. of the International Conference on Management of Data. – 1998. – P. 319-330.

80. Kotidis, Y. DynoMat: a dynamic view management system for data ware-houses [Text] / Y. Kotidis, N. Roussopoulos // Proc. of the International Conference on Management of Data. – 1999. – P. 371-382.

81. Roy, P. Efficient and extensible algorithms for multi query optimization [Text] / P. Roy, S. Seshadri, S. Sudarshan, S. Bhohe // LNCS. – 2000. – V. 1909. – P. 249-260.

82. Smith, M. Optimizing the performance of a relational algebra performance interface [Text] / M. Smith, P. Y. W. Chang // Communications for the ACM. – 1975. – № 10. – P. 568-579.

83. Aho, A. Efficient optimization of a class of relational expressions [Text] / A. Aho, Y. Sagiv, J. Ullman // ACM Transactions on Database Systems. – 1979. – № 4. – P. 435-454.

84. Aho, A. Equivalences among relational expressions [Text] / A. Aho, Y. Sagiv, J. Ullman // Journal on Computing. – 1979. – № 2. – P. 218-246.

85. Sagiv, Y. Equivalences among relational expressions with union and difference operators [Text] / Y. Sagiv // Journal of the ACM. – 1980. – № 4. – P. 633-655.

86. Krishnamurthy, R. XML-SQL query translation literature: the state of the art and open problems [Text] / R. Krishnamurthy, R. Kaushik, J. Naughton // LNCS. – 2003. – V. 2824. – P. 1-18.

87. Frasinca, F. XAL: an algebra for XML query optimization [Text] / F. Frasinca, G.-J. Houben, C. Pau // Proc. of the 13th Australian Conference. – 2002. – Vol. 5. – P. 49-56.

88. Lukichev, M. XML query algebra for cost-based optimization [Электронный ресурс] / M. Lukichev, D. Barashev // Proc. of the 4th Spring Young Researches Colloquium on Databases and Information Systems. – Режим доступа: <http://ceur-ws.org/Vol-256/>.

89. Re, C. A complete and efficient algebraic compiler for Xquery [Text] / C. Re, J. Simeon, M. F. Fernandez // LNCS. – 2007. – V. 4797. – P. 81-96.

90. Пашинин, О.В. Оптимизация запросов к базам данных [Текст] / О. В. Пашинин // Математические структуры и моделирование. – 2007. – Вып. 17. – С. 100-107.

91. Mendkovich, N. New algorithms for lexical query optimization [Text] / N. Mendkovich, S. Kuznetsov // Proc. of the 31st International Conference on Information Technology Interfaces. – 2009. – P. 187-192.

92. Кузнецов, С. Д. Новые алгоритмы лексической оптимизации запросов [Text] / С. Д. Кузнецов // Моделирование и анализ информационных систем. – 2009. – № 4. – С. 22-33.

93. Shenoy, S. T. A system for semantic query optimization [Text] / S. T. Shenoy, Z. M. Ozsoyogly // SIGMOD Record. – 1987. – № 3. – P. 181-195.

94. Sun, W. Semantic query optimization for tree and chain queries [Text] / W. Sun, C. Yu // IEEE Transactions on Knowledge and Data Engineering. – 1994. – № 1. – P. 136-151.

95. Geng, K. Survey of XML query optimization [Text] / K. Geng, G. Dobbie, Y. Meng // Proc. of the 4th International Conference on Internet Computing for Science and Engineering. – 2009. – P. 297-300.

96. Parallel algorithms for the execution of relational database operations [Text] / D. Bitton, H. Boral, D. Dewitt, W.K. Wilkinson // ACM Transactions on Database Systems. – 1983. – № 3. – P. 324-353.

97. Wong, E. Distributing a database for parallelism [Text] / E. Wong, R. H. Katz // Proc. of the SIGMOD Annual Meeting. – 1983. – P. 23-29.

98. DeWitt, D. Parallel database systems: the future of high performance database systems [Text] / D. Dewitt, J. Gray // Communications of the ACM. – 1992. – № 6. – P. 1-26.

99. Hasan, W. Coloring away communication in parallel query optimization [Text] / W. Hasan, R. Motwani // Proc. of the 21st International Conference on Very Large Databases. – 1995. – P. 239-250.

100. Соколинский, Л. Б. Организация параллельного выполнения запросов в многопроцессорной машине баз данных с иерархической структурой [Текст] / Л. Б. Соколинский // Программирование. – 2001. – № 6. – С. 13-29.

101. Григорьев, Ю. А. Теоретические основы анализа процессов доступа к распределенным базам данных [Текст] / Ю. А. Григорьев, А. Д. Плутенко. – Новосибирск : Наука, 2002. – 222 с.

102. Соколинский, Л. Б. Обзор архитектур параллельных систем баз данных [Текст] / Л. Б. Соколинский // Программирование. – 2004. – № 6. – С. 1-15.

103. Щетинин, Д. И. Оптимизация запросов в системах баз данных на параллельных структурах [Текст] / Д. И. Щетинин // Проблемы програмування. Спеціальний випуск. – 2006. – № 2-3. – С. 102-109.

104. Григорьев, Ю. А. Оценка времени выполнения запросов и выбор архитектуры параллельной системы баз данных [Текст] / Ю. А. Григорьев, В. Л. Плужников // Организация баз данных. – 2009. – № 3. – С. 3-12.

105. Fast computation of database operations using graphic processors [Text] / N. Govindaraju, B. Lloyd, W. Wang at all // Proc. of International Conference on Management of Data. – 2004. – P. 215-226.

106. GPU TeraSort: high performance graphics co-processor sorting for large database management [Text] / N. Govindaraju, J. Gray, R. Kumar at all // Proc. of International Conference on Management of Data. – 2006. – P. 325-336.

107. Mars: A MapReduce framework on graphics processors [Text] / B. He, W. Feng, Q. Luo at all //

Proc. of the 17th International Conference on Parallel Architectures and Compilation Techniques. – 2008. – P. 260-269.

108. *Relational query co-processing on graphic processors [Text] / B. He, M. Lu, K. Yang at al // ACM Transactions on Database Systems. – 2009. – № 4. – P. 1-39.*

109. *Baccum, P. Accelerating SQL database operations on GPU with CUDA [Text] / P. Baccum, K. Skadron // Proc. of Workshop on General-Purpose*

Computation on Graphics Processing Units. – 2010. – P. 94-103.

110. *Лыфарь, Д. А. Параллельные алгоритмы обработки реляционных баз данных [Текст] / Д. А. Лыфарь // Вестник НГУ. Серия: Информационные технологии. – 2010. – Вып. 4. – С. 72-80.*

111. *Zheng, Y. A quantitative performance analysis model for GPU architectures [Text] / Y. Zheng, J.D. Owens // Proc. of the 17th International Symposium on High-Performance Computer Architecture. – 2011. – P. 382-393.*

Поступила в редакцію 7.02.2013, рассмотрена на редколлегии 25.03.2013

Рецензент: д-р техн. наук, проф. С. В. Листровой, Харьковская государственная академия железнодорожно-го транспорта, Харьков, Украина.

СКЛАДНІСТЬ ОПЕРАЦІЙ З БАЗАМИ ДАНИХ (ОГЛЯД)

Д. Б. Буй, В. Г. Скобелев

У роботі досліджено сучасний стан проблеми забезпечення ефективності функціонування систем управління базами даних (СУБД) з позиції моделей і методів, призначених для оптимізації запитів. На основі аналізу особливостей процесу проектування СУБД, основних моделей даних, основних понять, що використовуються при оцінці складності алгоритмів, показано, що в цьому контексті слід розглядати саме зв'язку «БД–СУБД». Охарактеризовано основні типи математичних моделей, призначених для теоретичного аналізу ефективності функціонування зв'язки «БД–СУБД» в термінах аналізу складності операцій. Розглянуто еволюцію методів «оптимізації запитів» (що є, по своїй суті, евристичними методами).

Ключові слова: бази даних, СУБД, оптимізація запитів.

COMPLEXITY OF OPERATIONS IN DATABASE SYSTEMS (A SURVEY)

D. B. Bui, V. G. Skobelev

This paper attempts to explore the current state of the problem of providing efficiency of database management systems (DBMS) from the perspective of models and methods for optimizing queries. On the basis of analyzing the process of designing databases, basic data models, the basic concepts used in the theory of the complexity of algorithms, it is shown that in this context should be seen a bunch of "DB–DBMS". The main types of mathematical models for the theoretical analysis of the efficiency of the bunch "DB–DBMS" are described. The evolution of methods of "query optimization" (which are, in essence, heuristics) is described too.

Keywords: database systems, DBMS, query optimization.

Буй Дмитрий Борисович – д-р физ.-мат. наук, профессор, профессор кафедры теории и технологии программирования факультета кибернетики Киевского национального университета им. Тараса Шевченко, Киев, Украина, e-mail: buy@unicyb.kiev.ua

Скобелев Владимир Геннадиевич – д-р физ.-мат. наук, д-р техн. наук, профессор, ведущий научный сотрудник Института прикладной математики и механики НАН Украины, Донецк, Украина, e-mail: skbv@iamm.ac.donetsk.ua.