

УДК 621.396.97

МЕТОДИКА ПРОЕКТИРОВАНИЯ ЭФФЕКТИВНЫХ ЛОГИЧЕСКИХ СТРУКТУР БАЗ ДАННЫХ СИСТЕМ ТЕХНИЧЕСКОЙ ДИАГНОСТИКИ

Баин А.М., Каунг Сан

Национальный исследовательский университет «МИЭТ», Москва, Россия (124498, Москва, Зеленоград, проезд 4806, дом 5), e-mail: evgen_uis@mail.ru

Эффективность функционирования систем технической диагностики, быстродействие информационных обменов во многом определяется оптимальностью построения составляющих их структур локальных баз данных. В статье предложен формализованный подход, который позволяет проектировать эффективные структуры баз данных для использования в системах технической диагностики. Предложенный в статье подход основан на решении задачи нормализации графа логической структуры, каждого узла вычислительной системы; определения несвязных и слабо связанных компонентов (подграфов) графа логической структуры каждого узла; проектирования локальных баз данных, поддерживаемых конкретными системами управления базой данных. Проведенное моделирование показало, что общее время обработки запроса для предложенной структуры базы данных уменьшилось в среднем на 15 % по сравнению с традиционным подходом к их проектированию.

Ключевые слова: техническая диагностика; база данных, система управления, проектирование, математическая модель, быстродействие.

EFFECTIVE TECHNIQUE FOR DESIGNING LOGICAL DATABASE STRUCTURES OF TECHNICAL DIAGNOSTICS SYSTEMS

Bain A.M., Kaung San

National Research University of Electronic Technology, Moscow, Russia (124498, Moscow, Zelenograd, travel 4806, 5), e-mail: evgen_uis@mail.ru

Effective systems of technical diagnostics, performance information exchanges largely determines the optimal construction of their constituent structures of local databases. This paper proposes a formal approach, which allows you to design effective database structure for use in technical diagnostics. Proposed article approach is based on solving the problem of normalization of the logical structure of the graph, each node of a computer system; determination disconnected and weakly connected components (subgraphs) of the graph of the logical structure of each node, the design of local databases, supported by specific database management system. The simulation showed that the overall query processing time for the proposed structure of the database has decreased on average by 15 % compared with the traditional approach to their design.

Keywords: information transfer protocol, redundancy, system remote control system, the communication channel, controlled point.

Эффективность обработки информации при проведении технической диагностики сложных объектов вычислительных систем (ВС) во многом определяется структурой БД системы технической диагностики [1-4]. Локальные БД являются функционально законченными элементами и для них должны быть спроектированы логические и физические структуры, поддерживаемые конкретными типами систем управления базами данных (СУБД) и операционных систем [6].

Основными временными характеристиками функционирования БД являются время реализации заданного множества запросов T^z и время реализации заданного множества заданий на корректировку T^k , которые в сумме дают общее время выполнения нагрузки БД, т. е.

$$T = \sum_{P=1}^{P0} T_p^z + \sum_{S=1}^{S0} T_s^k, \quad (1)$$

где T_p^z – время реализации p -го запроса пользователя, а T_s^k – время реализации p -го задания на корректировку.

Процесс реализации запроса пользователя состоит из выполнения ряда этапов: определения с помощью сетевых каталогов места расположения локальных БД в ВС, содержащих требуемые типы записей; выбор и реализация программ прикладного и сеансового уровней протоколов, обеспечивающих разложение запроса на подзадачи; выбор маршрута, установление виртуального соединения между узлами ВС и передача запросов (подзадач) в локальные БД для их реализации.

Время реализации p -го запроса пользователя складывается из следующих составляющих:

$$T_p^z = t_p^{pz} + t_p^e + t_p^{c\phi} + t_p^u, \quad (2)$$

где t_p^{pz} – время работы программ прикладного и сеансового уровней протоколов, обеспечивающих декомпозицию запроса на подзадачи и управление их реализацией;

t_p^e – время реализации p -го запроса в локальной БД;

$t_p^{c\phi}$ – время работы программ представительского и прикладного уровней протоколов, связанных с выполнением процедур сборки массивов промежуточных данных и формирования окончательного результата;

t_p^u – время инициирования и передачи запроса (подзадачи) в локальную БД (узел ВС).

Время работы программ прикладного и сеансового уровней протоколов определяется как

$$t_p^{pz} = t_1^\phi k_p^y, \quad (3)$$

где t_1^ϕ – время формирования одной подзадачи;

k_p^y – количество формируемых подзадач, определяемое числом различных локальных БД (узлов ВС), к которым необходимо обратиться для поиска требуемых типов записей.

Реализация запроса в локальных БД включает процедуры поиска и считывания информации, формирования массивов промежуточных данных и передачи их в запрашивающий узел ВС, т.е.

$$t_p^e = t_p^c + t_p^\Pi, \quad (4)$$

где t_p^c – время, затрачиваемое на поиск и считывание информации, необходимой для реали-

зации p -го запроса, из локальной БД в оперативную память;

t_p^{Π} – время работы программ транспортной сети, обеспечивающих выбор маршрута, установление виртуального соединения и передачу информации по каналу связи.

Время формирования выходных сообщений, представляемых пользователю, определяется в виде

$$t_p^{c\delta} = t_1^{cl} k_p^y, \quad (5)$$

где t_1^{cl} – время работы программ представительного и прикладного уровней протокола, обеспечивающих выполнение операций сборки результатов двух подзадач в процессе формирования выходного сообщения;

k_p^y – количество процедур сборки, определяемое количеством порожденных подзадач.

Время поиска требуемых типов записей определяется как

$$t_p^c = (t_1^{\delta ocm} + t_1^{o\delta c}) + t_1^{o\delta m} k_p^B, \quad (6)$$

где $t_1^{\delta ocm}$ – время поиска требуемых записей (блоков) в локальной БД; $t_1^{o\delta c}$ – время ожидания доступа и считывания информации из БД; $t_1^{o\delta m}$ – время обмена между ВЗУ и оперативной памятью; k_p^B – количество считываемых блоков при выполнении p -го запроса.

Обозначим

$$t_1^{\delta o} = t_1^{\delta ocm} + t_1^{o\delta c}, \quad (7)$$

где $t_1^{\delta o}$ – время ожидания доступа к БД и поиска требуемых записей (блоков).

Время поиска и время обмена определяют общее процессорное время, т.е.

$$t_1^{n\text{проц}} = t_1^{\delta ocm} + t_1^{o\delta m}. \quad (8)$$

Время поиска информации в локальной БД определяется характеристиками физических методов организации данных и параметрами устройств ВС [5].

Время обмена данными между локальной БД и оперативной памятью определяется характеристиками процедур считывания (записи) информации и объемом считываемой информации:

$$t_1^{o\delta m} = \tau_e V_B, \quad [9]$$

где τ_e – время считывания (записи) единицы информации из локальной БД;

V_B – объем считываемой информации.

Для расчета среднего времени ожидания $t_1^{ож}$ целесообразно использовать методы теории массового обслуживания. Рассмотрим ВС как стационарную однолинейную систему массового обслуживания с неограниченным ожиданием при пуассоновском входящем потоке заявок с интенсивностью λ . Полное время обслуживания заявки ($t_1^{проц}$) состоит из суммы независимых случайных величин: времени поиска объекта ($t_1^{доцм}$) и длительности передачи информации ($t_1^{обм}$), которые имеют равномерные распределения в интервалах $(\overline{0}, t_0)$ и $(\overline{0}, \tau_0)$ соответственно, где t_0 – время полного цикла записи и τ_0 – максимальная длительность передачи информации в оперативную память.

Под оптимальной логической структурой локальной БД понимается множество типов записей, размещенных в узле ВС и соединенных системой адресных указателей, обеспечивающих оптимальное по заданному критерию эффективности выполнение требований пользователей на обработку информации [6].

Проектирование оптимальных логических структур локальных БД основывается на их синтезе, эффективность которых определяется общесистемным критерием оптимальности функционирования БД. Оптимальные логические структуры локальных БД проектируются в этом случае на основе результатов синтеза логической структуры БД, выбора типа и проектирования структуры сетевого каталога. Проектирование осуществляется путем нормализации графа логической структуры отдельного узла ВС, формируемого как результат синтеза оптимальной логической структуры БД, и определения в графе несвязных и слабосвязных подграфов, являющихся основой логических структур локальных БД, поддерживаемых конкретными системами управления базой данных (СУБД).

Содержательная постановка задачи формулируется следующим образом. По известным характеристикам оптимальной логической структуры БД и сетевого каталога:

$H = \{ h_i \}$ – множеству типов записей; $\Psi^{3n} = \| \psi_{ir}^{3n} \|$ – матрице размещения типов записей по узлам ВС; $A^{3n} = \| a_{i1i2}^{3n} \|$ – матрице семантической смежности типов записей; $B^{3n} = \| b_{i1i2}^{3n} \|$ – матрице типов отношений между логическими записями; $N^{3n} = \| n_{ii}^{3n} \|$ – матрице состава типов записей – необходимо определить количество локальных БД, расположенных на каждом узле ВС, их логические структуры, обеспечивающие оптимальное функционирование БД (в смысле критерия эффективности оптимальной логической структуры БД) и удовлетворяющие структурным и системным ограничениям, накладываемым конкретными системами управления базой данных (СУБД) и операционными системами.

Для определения оптимального количества локальных БД в узлах ВС необходимо решить задачи нормализации графа логической структуры, r -го узла ВС; определения несвяз-

ных и слабо связных компонентов (подграфов) графа логической структуры r -го узла; проектирования локальных БД, поддерживаемых конкретными системами управления базой данных (СУБД).

Граф логической структуры r -го узла ВС определяется как множество типов записей, размещенных на r -м узле ВС, которые соединены между собой множеством логических связей.

На основании информации, зафиксированной в сетевом каталоге, характеризующей распределенность типов записей по узлам ВС и формализованной в виде матрицы $\Psi^{3n} = \|\Psi_{ir}^{3n}\|$, определим подмножество типов записей $H_r \subset H$, расположенных на r -м узле ВС: $\Psi^{3n} = \|\Psi_{ir}^{3n}\|$.

На множестве H_r с использованием информации об отношениях между типами записей, зададим матрицу $A_r^{3n} = \|a_{i_1 i_2}^{3n}\|$, отражающую взаимосвязь между типами логических записей, расположенных на r -м узле ВС. Матрица A_r^{3n} есть матрица семантической смежности множества типов записей и является исходной для процедур проектирования логических структур локальных БД. Множество H_r и матрица A_r^{3n} определяют граф логической структуры r -го узла ВС $G_r^{3n} = (H_r, R_r^{3n})$.

Для определения подграфов графа логической структуры r -го узла ВС решается следующая задача. Пусть $H_r^{(1)} = \{h_{pr}^1\}$ – множество типов записей первого уровня иерархии графа G_r^{3n} . Множество $H_r^{(1)}$ есть минимальное множество вершин графа G_r^{3n} , из которого достижимы все вершины, т. е. является базой графа [3]. Пусть $F(H_r^{(1)})$ – множество достижимости базы $H_r^{(1)}$, тогда $F(H_r^{(1)}) = H_r$, и $\forall H_r^* \subset H_r, F(H_r^*) \neq H_r$. Определим множество достижимости для каждого типа записи множества $H_r^{(1)} = \{h_{pr}^1\}$:

$$F(h_{pr}^1) = \{h_{pr}^1\} \cup \Gamma^1(h_{pr}^1) \cup \Gamma^2(h_{pr}^1) \cup \dots \cup \Gamma^n(h_{pr}^1),$$

где $n \leq I_0 - 1$.

Анализ попарных пересечений множеств достижимости позволяет выявить несвязные компоненты (подграфы) в графе логической структуры r -го узла ВС. Два подграфа G_{p_1} и G_{p_2} , графа G_r^{3n} , такие что $h_{p_1 r}^{(1)} \in H_r^{(1)}$ и $h_{p_2 r}^{(1)} \in H_r^{(1)}$ где $h_{p_1 r}^{(1)}$ и $h_{p_2 r}^{(1)}$ – корневые вершины графов G_{p_1} и G_{p_2} являются несвязными, если

$$F(h_{p_1 r}^{(1)}) \cap F(h_{p_2 r}^{(1)}) = 0.$$

Множество типов записей и отношений между ними, образующих несвязный под-

граф графа логической структуры r -го узла ВС, заносится во множество, используемое при проектировании, логических структур локальных БД.

Для выделения несвязных компонент графа G_r^{3n} используется следующая процедура:

1. Определяется множество типов записей первого уровня иерархии $H_{ri}^{(1)} = \{h_{pri}^{(1)}\}$ для i -й связной компоненты графа G_r^{3n} .
2. Определяется множество достижимости для каждого типа записи $H_{pri}^{(1)} \in \{h_{ri}^{(1)}\}$

$$F(h_{pri}^{(1)}) = \{h_{pri}^{(1)}\} \cup \Gamma^1(h_{pri}^{(1)}) \cup \Gamma^2(h_{pri}^{(1)}) \cup \dots \cup \Gamma^n(h_{pri}^{(1)}).$$
3. Определяется множество $M^{(p)} = \{m_j^{(p)}\}$ попарных пересечений множеством достижимости

$$m_j^{(p)} = F(h_{p1ri}^{(1)}) \cap F(h_{p2ri}^{(1)}).$$
4. Анализ элементов множества $M^{(p)}$. Если множество попарных пересечений состоит из пустых подмножеств, то перейти к пункту 7, иначе – к пункту 5.
5. Выбирается минимальное подмножество m^{\min} множества $M^{(p)}$,
 $card(m^{\min}) = \min[card(m_j^{(p)})]$, элементы которого составляют множество вершин выделяемого подграфа графа G_r^{3n} .
6. Из множества вершин графа G_r^{3n} удаляются вершины выделенного подграфа и связи, ведущие к его корневой вершине. Переход к пункту 2.
7. К множеству выделяемых подграфов добавляются несвязные компоненты (подграфы) графа G_r^{3n} , получаемые в результате удалений вершин и связей. Конец процедуры.

Необходимость выделения слабосвязных компонент графологической структуры r -го узла, т. е. выделение подграфов логических структур локальных БД, определяется ограничениями на объем хранимой информации в узлах ВС, требованиями к оперативности копирования и восстановления БД и др.

Пусть граф G_r^* , вершинами которого являются корневые вершины выделенных подграфов, а дугами – связи между ними, есть остовный граф или остов графа логической структуры r -го узла. С целью определения возможности разрезания остовного графа определяется степень связности его вершин. В качестве оценки степени связности вершин остовного графа применяется суммарная частота использования дуг остовного графа запросами и корректировками пользователей.

Решение об удалении связи и образовании несвязного подграфа, рассматриваемого в качестве логической структуры локальной БД, принимается проектировщиком на основе информации о составе (типах записей) выделяемых подграфов и степени связности вершин

основного графа G_r^* с учетом особенностей технологии и опыта эксплуатации локальных БД конкретного узла ВС.

На основе информации о типах записей и их взаимосвязи, определяющих выделенный подграф, проектируется логическая структура локальной БД. При этом учитываются особенности и ограничения конкретной системы управления базами данных (СУБД).

Для подтверждения эффективности предложенной методики проведено моделирование обработки запросов к БД при использовании существующей и предложенных структур БД, результаты которого представлены на рисунке [4,5]. Как видно из рисунка, общее время обработки запроса для предложенной структуры БД – $t_{БД}$ уменьшилось в среднем на 15 % .

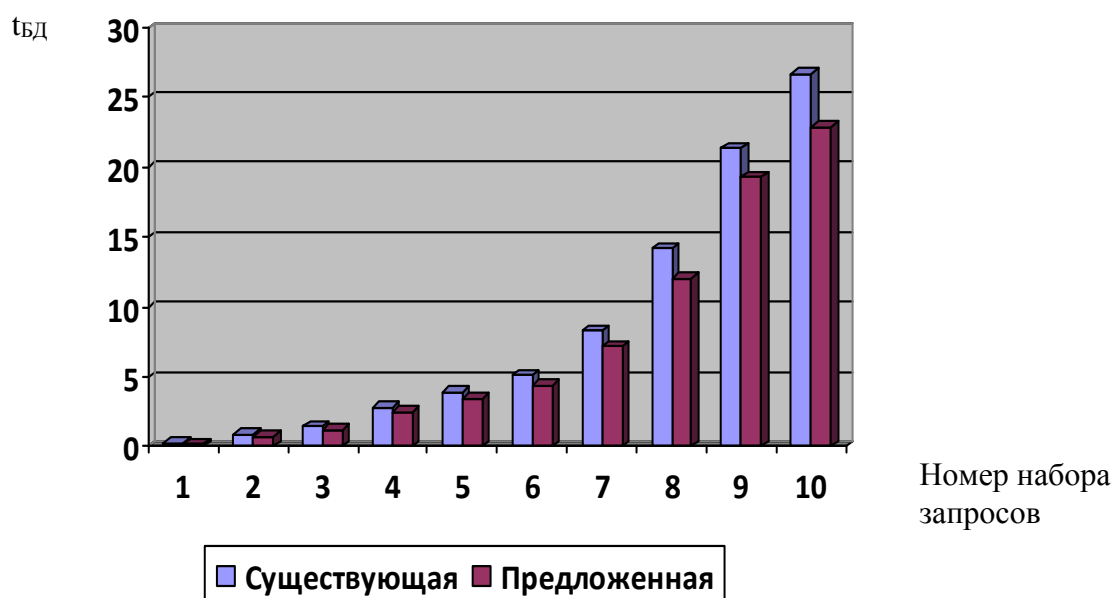


Рисунок. Эффективность методики проектирования структур БД

Таким образом, предложенная методика позволяет проектировать эффективные структуры БД для использования в системах технической диагностики и уменьшать время обработки запросов по сравнению с традиционной [1,4].

Список литературы

1. Баин А.М., Портнов Е.М. Методика синтеза многофункциональных систем управления энергообеспечением // XXXVIII Международная конференция «Информационные технологии в науке, социологии, экономике и бизнесе» // Труды конференции, Украина, Крым, Ялта-Гурзуф, 2012. – С.154-157.

2. Каунг Сан. Методика определения предполагаемого количества отказов при технической диагностике // Теоретические и прикладные вопросы науки и образования: сб. науч. тр. по материалам Междунар. науч.-практ. конф. 31 августа 2013 г.: Ч. 3. – Тамбов, 2013. URL: http://www.ucom.ru/doc/conf/2013_08_31_3.pdf (дата обращения: 25.03.2013).
3. Каунг Сан. О выборе критерии качества функционирования автоматизированных систем для технической диагностики в приборостроении // Актуальные проблемы информатизации в науке, образовании и экономике – 2013: тезисы докл. Всерос. конф. (Москва, 7–8 нояб. 2013 г.). – Зеленоград, 2013. – С.130.
4. Портнов Е.М., Баин А.М., Чумаченко П.Ю. Методика разработки систем автоматизации бизнес-процессов предприятий // Оборонный комплекс – научно-техническому прогрессу России. – 2011. – № 2. – С. 69-73.
5. Портнов Е.М., Каунг Сан. Разработка способа повышения эффективности информационных обменов по магистральным каналам связи // Интернет журнал «Науковедение». – 2013. – Вып.6. – <http://naukovedenie.ru/PDF/29TVN613.pdf>.
6. Советов Б.Я., Цехановский В.В., Чертовской В.Д. Базы данных. Теория и практика. – М.: Юрайт, 2013. – 464 с.

Рецензенты:

Портнов Е.М., д.т.н., профессор кафедры «Информатика и программное обеспечение вычислительных систем» Национального исследовательского университета «МИЭТ», г.Москва.

Гагарина Л.Г., д.т.н., профессор, зав. кафедрой «Информатика и программное обеспечение вычислительных систем» Национального исследовательского университета «МИЭТ», г. Москва.