УЧЕНЫЕ ЗАПИСКИ ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЙ ЛИТОВСКОЙ ССР
ЭКОНОМИКА, 27, 1984

АДАПТИВНОСТЬ БАЗЫ ДАННЫХ

А.-А. БАРТКУС

Как известно, одним из важнейших свойств базы данных автоматизированного банка данных является независимость базы данных от прикладных программ (1). Это свойство позволяет в принципе видоизменять программы потребителей независимо от логической и физической организации базы данных, и, наоборот, менять логическую и физическую организацию базы данных без внесения каких-либо изменений в программы потребителей. Тем не менее при функциональном подходе, который сейчас доминирует при проектировании банков данных, база данных строится исходя из характеристик потребителей (таких, как важность отдельных потребителей, типы запросов, их частота и т. п.) с целью организации рациональной работы с базами данных. Поэтому структура базы данных в какой-то мере отражает характеристики совокупности потребителей, и в этом смысле база данных является зависимой от прикладных программ, при помощи которых реализуются информационные потребности потребителей.

При изменении характеристик потребителей существующая база данных может недостаточно эффективно удовлетворять их информационные потребности, и, таким образом, в определенный момент возникает необходимость в реструктуризации базы данных.

Структура базы данных может оцениваться следующими показателями (2, с. 20—24): трудоемкостью формирования; объемом дополнительной памяти; временем поиска; временем корректировки; трудоемкостью реорганизации; трудоемкостью преобразования данной структуры в другие структуры (трудоемкостью реструктуризации).

Пусть во время эксплуатации данной базы данных метод организации данных остается без изменений (примем для конкретности, что на логическом уровне база данных имеет иерархическую структуру, а на физическом — данные организованы в индексно-последовательные файлы).

Объем дополнительной памяти зависит от динамичности хранимых данных, трудоемкости реорганизации и практически не зависит от способа организации данных (при ранее оговоренных ограничениях). Большого значения не имеет также трудоемкость формирования, так как она производится один раз за время существования данной базы данных и, следовательно, является небольшой в сравнении с затратами на эксплуатацию базы.

При реструктуризации базы данных происходит пересмотр системы главных и вспомогательных наборов. Во время определения новых главных наборов необходимо переупорядочить системные массивы и, как следствие, определить новые адреса связи для вспомогательных наборов. Естественно, что такая реструктуризация связана с значительным расходом машинного времени.

Таким образом, эффективность базы данных целесообразнее всего оценивать по времени поиска и корректировки, а также трудоемкости реструктуризации.

Примем, что во время одного запроса может вызываться несколько экономических параметров. Если эти параметры имеют одно значение ассоциативной части, характеризуемое набором значений ряда ключевых признаков, то они находятся в одной записи системного файла. На затраты машинного времени будет иметь влияние не количество запрашиваемых параметров, а количество запрашиваемых записей системного файла.

Оценим затраты машинного времени на поиск. За критерий сравнения примем количество обращений к внешним запоминающим устройствам для поиска одной записи. Каждый тип запроса характеризуется тремя величинами:

— рядом ключевых признаков $k_i = <x_{i1}, x_{i2}, \ldots, x_{in}>$, по которым производится поиск (x_{ij} — элементарный ключ-реквизит);

— средним количеством запрашиваемых экземпляров записей системного файла l_i;

— частотой запроса h_i.

Система управления базой данных (СУБД) может быть такой — и это соответствует предложениям РГБД КОДАСИЛ (3),— что каждая запись имеет ключ базы данных и, с другой стороны, записи, относящиеся к одному экземпляру концевого набора (под концевым набором подразумеваются наборы, записи-члены которого не являются членами-владельцами других наборов), будут расположены физически одной группой и занимать одну или несколько рядом расположенных физических записей (сегментов).

Пусть главные наборы упорядочены по ряду ключевых признаков p. Тогда в поисковом предписании указанный ряд ключевых признаков k_i может не совпадать с рядом p либо совпадать. В первом случае, очевидно, поиск необходимо осуществлять по ключам базы данных, и одна запись будет найдена в среднем примерно за одно обращение к внешним запоминающим устройствам (ВЗУ) — $O_i = 1$.

Если $p = k_i$, то при некотором значении среднего количества выбираемых записей H из одного экземпляра концевого набора может оказаться целесообразным просматривать записи-члены набора последовательно.

Если в физической записи (сегменте) помещается L логических записей, а в наборе имеется N записей-членов, то один экземпляр набора будет размещен в $n = \left\lceil \frac{N}{L} \right\rceil$ либо в $n+1$ сегментах, причем вероятность, что будет использовано n физических записей при равновероятном использовании любой логической записи сегмента в качестве первой записи-члена набора

$$\eta_n = n - \frac{N-1}{L}. \qquad (1)$$

Вероятность, что будет использована $n+1$ запись

$$\eta_{n+1} = \frac{N-1}{L} - (n-1). \qquad (2)$$

Определим, сколько сегментов необходимо будет извлечь из ВЗУ для поиска H логических записей. Поскольку вероятность выбора одной записи равна $\frac{H}{N}$, то вероятность невыбора последних l записей равна

$$\left(1 - \frac{H}{N}\right)^l$$

Если набор занимает n сегментов, то как в первом, так и в последнем сегментах может быть от $i_n = N - (n-1)L$ до L записей; если набор за-

нимает $n+1$ сегмент, то в первом и в последнем сегменте число записей может колебаться от 1 до $i_{n+1}=N-(n-1)L$. Таким образом, вероятность невыбора последнего сегмента, если набор занимает n сегментов, равна

$$\xi_n=\frac{1}{L-i_{n+1}}\sum_{l=i_n}^{L}\left(1-\frac{H}{N}\right)^l \tag{3}$$

и если набор занимает $n+1$ сегментов —

$$\xi_{n+1}=\frac{1}{i_{n+1}}\sum_{l=1}^{i_{n+1}}\left(1-\frac{H}{N}\right)^l. \tag{4}$$

Вероятность невыбора второго и других полностью занятых сегментов набора

$$\xi_2=\xi_3=\quad=\left(1-\frac{H}{N}\right)^L \tag{5}$$

Отсюда математическое ожидание количества обращений при последовательном просмотре на одну искомую запись равняется

$$O_p=\frac{1}{H}\left[\eta_n\left(n-\sum_{s=1}^{n-1}s\prod_{l=n-s}^{n-1}\xi_i\right)+\eta_{n+1}\left(n+1-\sum_{s=1}^{n}s\prod_{l=n-s}^{n}\xi_i\right)\right]. \tag{6}$$

Если $O_p\geqslant 1$, то последовательный просмотр экземпляра набора себя не оправдывает, и во всех случаях поиск следует производить используя ключи базы данных, а главные наборы никакой роли в поисках данных не играют.

Если $O_p<1$, то можно подобрать оптимальные главные наборы, минимизирующие функцию

$$Z=\sum_i l_i h_i x_i, \tag{7}$$

где $x_i=\begin{cases}\dfrac{O_p l_l}{H_l}, & \text{если } k_i=p\\ 1, & \text{если } k_i\neq p.\end{cases}$

Поскольку переменная x_i может принять только два дискретных значения, а число i невелико, минимальное значение функции Z проще всего найти методом перебора.

Для этого воспользуемся следующим соображением. Максимальное значение функция Z приобретает тогда, когда ни один тип запроса по последовательности ключей не соответствует упорядочению ряда ключей главных наборов. Поскольку по нашему условию подобное соответствие может быть достигнуто только для одного типа запроса, необходимо найти такой единственный набор p, который даст максимальное снижение значения Z:

$$\Delta Z_p=\max_l\left\{l_i h_i\left(1-\frac{O_i l_l}{H_l}\right)\right\}. \tag{8}$$

По этому набору и производится упорядочение логических записей на физическом уровне.

Следует отметить, что частота обращений с запросами определенного типа h_i и количество обрабатываемых при этом записей l_i являются функциями времени, так как информационные потребности отдель-

ных лиц во времени могут меняться. Более того, даже в период разработки и загрузки базы данных величины h_i и l_i должны восприниматься как математические ожидания некоторых случайных величин $h_i(t)$ и $l_i(t)$.

Допустим, что при загрузке базы данных ($t=0$) в качестве главного набора ключей был определен набор p. Однако с момента t_1 были обнаружены значительные отклонения $l_i(t)$, $h_i(t)$ и $H_i(t)$ от первоначальных. Кроме того, известны траектории изменения этих величин вплоть до момента времени t_2. Возникает вопрос: не следует ли произвести реструктуризацию базы данных, изменив главный набор ключей? Потери машинного времени $\Delta E(t)$ в период t_1-t_2 в связи с нереструктуризацией базы данных определяются выражением

$$\Delta E(t_1-t_2)=\max_{i\neq p}\int_{t_1}^{t_2}\Delta Z_i(t)\,dt-\int_{t_1}^{t_2}\Delta Z_p(t)\,dt. \qquad (9)$$

Пусть затраты машинного времени на реструктуризацию равны W. Тогда рекструктуризацию целесообразно производить только в том случае, если

$$\Delta E(t_1-t_2)>W. \qquad (10)$$

Если интервал времени t_1-t_2 небольшой, то отклонения значений $l_i(t)$, $h_i(t)$ и $H_i(t)$ могут носить флуктуационный характер, и условие (10) будет иметь сдерживающий характер, препятствующий преждевременной реструктуризации базы.

Рассмотрим некую условную базу данных иерархической структуры с ключами x_1, x_2 и x_3. Запросы к ней поступают двух типов: в последовательности ключей $k_1=\langle x_1, x_2, x_3\rangle$ и $k_2=\langle x_2, x_3, x_1\rangle$. Кроме того, в базу могут вноситься изменения, которые поступают в виде запросов, соответствующих главным ключам упорядочения. Будем считать, что для внесения одного изменения необходимо в два раза больше обращений, чем при запросе. Естественно, что $p=k_1$ либо $p=k_2$.

Пусть количество различных значений x_1, x_2, x_3 равно 20; тогда имеется 400 концевых наборов по 20 записей ($L=20$). Другие исходные и результатные данные представлены в таблице, в которой изменения сведены к запросам.

Таблица

Сравнение вариантов упорядочения данных

		n	H	O	ΔZ
1	2400	3	6	0,8312	1215
2	3600	2	9	0,6023	2863

Сравнение результатов последней строки показывает, что второй вариант является оптимальным.

Вильнюсский университет
им. В. Капсукаса
Кафедра экономической
информации

Редколлегии вручено
в октябре 1982 г.

ЛИТЕРАТУРА

1. Мартин Дж. Организация баз данных в вычислительных системах.— М.: Мир, 1980.
2. Николаев В. И., Анкуринов Г. И., Победнов В. А. Об оптимизации логической структуры базы данных.— Вопросы системотехники.— Л.: 1980, № 5.
3. Язык описания данных КОДАСИЛ / Пер. с англ. яз. Под ред. М. Р. Когаловского и Г. К. Столярова. М.: Статистика, 1981.