ОПТИМИЗАЦИЯ ЗАПРОСОВ К БАЗАМ ДАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ[1]

Кузнецов Л.А., Погодаев А.К., Овчинников В.В.
(*Липецкий государственный технический университет*)
pak@stu.lipetsk.ru

Введение

Информационные системы нашли широкое применение в системах, решающих производственные, экономические, технологические и другие задачи самого разного профиля. Эффективность таких систем зависит от нескольких факторов: технического обеспечения, структуры хранения данных, структуры запросов. Данная работа посвящена проблеме повышения эффективности запросов в СУБД, поддерживающих реляционную модель данных.

В работах отечественных исследователей уделяется внимание широкому кругу вопросов: статистическим, объектно-ориентированным, временным базам данных, базам данных реального времени [1, 2]. Все они, как правило, рассматривают проблемы, связанные с организацией хранения данных, а формализация и оптимизация запросов рассматривается лишь в контексте дедуктивных баз данных [3]. В зарубежных работах в основном акцентируется внимание на методы обработки неструктурированных данных и широковещательные базы данных [5].

В [4] даны основы метода решения задачи оптимизации запросов, различная модификация которого используется с успехом во многих дорогих современных реляционных СУБД со встроенными оптимизаторами. Однако на практике часто приходится использовать относительно недорогие системы, не имеющие оптимизаторов, где пользователи нередко применяют семантически прозрачные запросы, но не оптимальные с точки зрения их эффективности. Более того, как показывает опыт, и в системах со встроенными оптимизаторами небесполезна предварительная внешняя оптимизация. Возникает потребность в разработке программной надстройки для формирования оптимальных запросов. Поэтому в данной

[1] *Работа финансируется Министерством образования РФ в форме гранта Г00-4.1-68*

статье предлагается формальный подход, который будет полезен для реализации такой надстройки.

1. Модель реляционного выражения

Реляционное выражение состоит из элементов, каждый из которых связан с одним или более другими элементами выражения. Известно, что элементами выражения являются элементы множеств отношений, логических выражений, атрибутов, реляционных операций. Введем обозначения:

B – множество элементов реляционного выражения.

$K \subseteq B$ – множество отношений, участвующих в выражении.

$S \subseteq B$ – множество логических выражений, участвующих в реляционном выражении.

$P \subseteq B$ – множество атрибутов, участвующих в реляционном выражении.

$O \subseteq B$ – множество реляционных операций, участвующих в выражении.

$\Sigma \subseteq O, \Pi \subseteq O, D \subseteq O, U \subseteq O, R \subseteq O$ – множества реляционных операций каждого типа: селекция, проекция, декартово произведение, объединение, разность.

$H = K \cup O$ – множество аргументов реляционных операций.

$G \subseteq B^2$ – множество связей выражения. Если операция не коммутативна (разность), порядок аргументов будем задавать однонаправленной связью между аргументами: $(c,a) \in G, (c,b) \in G, (a,b) \in G, a \in B, b \in B, c \in O$, что интерпретируется так: "реляционная операция c имеет минимум два упорядоченных аргумента a и b. Если операция коммутативна, то никаких однонаправленных связей между аргументами быть не должно: $(c,a) \in G, (c,b) \in G, (a,b) \notin G, a \in B, b \in B, c \in O$.

$J \subseteq S \times P$ – множество связей между логическими выражениями и атрибутами; это соответствие является постоянным в ходе любых преобразований реляционного выражения, а поэтому может считаться внешним по отношению к реляционному выражению.

$T \subseteq K \times P$ – множество связей между отношениями и атрибутами. Это соответствие также остается постоянным в ходе любых преобразований реляционного выражения. Множество атрибутов, с которыми связано отношение, называется типом отношения. Типом аргумента называется либо тип исходного отношения, если аргумент является отношением, либо тип отношения, которое будет получено путем применения некоторых операций над исходными отношениями, вплоть до операции, являющейся аргументом, тип которого ищется.

$q=(G,B,J,T)$ – реляционное выражение из множества всех реляционных выражений $\{\Theta \mid q\in\Theta\}$.

Создадим несколько вспомогательных операций над реляционным выражением.

Определим операцию получения множества всех атрибутов проекции $p\in\Pi$ в реляционном выражении q, используя аппарат исчисления предикатов: $P'_q(p)=\{x \mid (p,x)\in G\wedge x\in P\}$.

Определим операцию получения множества всех атрибутов, связанных с логическими выражениями селекции $s\in\Sigma$ в реляционном выражении q: $P''_q(s)=\{x \mid \exists y[(s,y)\in G\wedge y\in S\wedge (y,x)\in J]\}$.

Определим операцию получения аргумента для произвольной реляционной операции $a\in O$ реляционного выражения q, а также обратную ей операцию:

$$\arg_q a=\{x \mid (a,x)\in G\wedge x\in H\},\ \arg_q^{-1} a=\{x \mid (x,a)\in G\wedge x\in H\}.$$

Исходя из свойств реляционного выражения, результатом обеих операций будет либо единственный элемент $x\in H$, либо пустое множество $\varnothing$, что означает непопадание a в область определения операции.

Определим операцию получения типа аргумента $a\in H$ в реляционном выражении q:

$$T_q(a)=\left\{x\,\middle|\,\begin{array}{l}(a\in K\to(a,x)\in T)\wedge\\ \exists b\left[\begin{array}{l}(a,b)\in G\wedge b\in H\wedge(a\in\Pi\to x=T_q(b)\cap P'_q(a))\wedge\\ (a\in D\to\exists c[(a,c)\in G\wedge c\in H\wedge c\neq b\wedge x=T_q(c)\cup T_q(b)])\end{array}\right]\end{array}\right\}.$$

Не всякое множество связей q соответствует корректному реляционному выражению. Поэтому необходимо задать совокупность предикатов, справедливость которых относительно конкретного реляционного выражения будет говорить о его корректности.

Для каждого подмножества элементов реляционного выражения $q\in\Theta$ можно сформулировать следующие утверждения:

1. Для селекций Σ:

$$\forall a(a\in\Sigma\to[\exists b((a,b)\in G\wedge b\in H\wedge\forall c[(a,c)\in G\to(c=b\vee c\in S)])]).$$

Для любой селекции существует аргумент из множества отношений либо из множества операций, и все остальные аргументы принадлежат множеству логических выражений.

2. Для проекций Π:

$$\forall a(a\in\Pi\to[\exists b((a,b)\in G\wedge b\in H\wedge\forall c[(a,c)\in G\to(c=b\vee c\in P)])]).$$

3. Для декартовых произведений D:

$$\forall a\left(a\in D\to\left[\exists b\exists c\left(\begin{bmatrix}(a,b)\in G\wedge b\in H\wedge\\ (a,c)\in G\wedge c\in H\wedge b\neq c\end{bmatrix}\wedge\overline{\exists d\begin{bmatrix}(a,d)\in G\wedge\\ d\neq b\wedge d\neq c\end{bmatrix}}\right)\right]\right).$$

4. Для объединений U:

$$\forall a\left(a\in U\to\left[\exists b\exists c\left(\begin{bmatrix}(a,b)\in G\wedge b\in H\wedge\\ (a,c)\in G\wedge c\in H\wedge b\neq c\end{bmatrix}\wedge\overline{\exists d\begin{bmatrix}(a,d)\in G\wedge\\ d\neq b\wedge d\neq c\end{bmatrix}}\right)\right]\right).$$

5. Для разностей R:

$$\forall a\left(a\in R\to\left[\exists b\exists c\left(\begin{bmatrix}(a,b)\in G\wedge b\in H\wedge\\ (a,c)\in G\wedge c\in H\wedge\\ b\neq c\wedge[(b,c)\in G\oplus(c,b)\in G]\end{bmatrix}\wedge\overline{\exists d\begin{bmatrix}(a,d)\in G\wedge\\ d\neq b\wedge d\neq c\end{bmatrix}}\right)\right]\right).$$

6. Если $\tilde{G}$ – транзитивное замыкание для G, то справедливо следующее утверждение: $\exists a\forall b[a\neq b\to(a,b)\in\tilde{G}]$.

7. Любая операция может быть аргументом только одной другой операции

$$\forall a\forall b[a\in O\wedge b\in O\wedge(a,b)\in G\to\overline{\exists c[c\in O\wedge(c,b)\in G]}].$$

8. Множество атрибутов проекции должно быть подмножеством типа ее аргумента: $\forall a[a\in\Pi\to P'_q(a)\subseteq T_q(\arg_q a)]$.

9. Множество атрибутов, с которыми связаны логические операции селекции, должно быть подмножеством типа ее аргумента:

$$\forall a[a\in\Sigma\to P''_q(a)\subseteq T_q(\arg_q a)].$$

10. Типы аргументов объединения (разности) должны совпадать:

$$\forall a[a\in U\vee a\in R\to\arg_q a=\{b,c\}\wedge T_q(b)=T_q(c)].$$

Если для q выполняются все перечисленные утверждения, то q является реляционным выражением с корректной структурой.

2. Формализация операций эквивалентного преобразования реляционного выражения

Исходя из законов эквивалентности реляционных выражений, q может быть преобразовано в q' без изменения соответствия между входом и выходом реляционного выражения.

Введем формализованные операции эквивалентного преобразования реляционных выражений в рамках созданной модели. Для упрощения далее приводятся только описания типов этих операций, без раскрытия их содержания и области определения, а также без описания вспомогательных операций.

1. Закон коммутативности декартового произведения заложен непосредственно в саму модель реляционного выражения.
2. Операция преобразования реляционного выражения по закону ассоциативности декартового произведения: $C_1 : \Theta \times H \times H \to \Theta$.
3. Преобразование выражения по закону каскада проекций $C_2 : \Theta \times \Pi \times \Pi \to \Theta$.
4. Преобразование выражения по закону каскада селекций $C_3 : \Theta \times \Sigma \times \Sigma \to \Theta$.
5. Операцию, обратную C_3, обозначим C_3^{-1}. Она имеет тип $C_3^{-1} : \Theta \times \Sigma \to \Theta$.
6. Перестановка селекции с проекцией имеет тип $C_4 : \Theta \times \Sigma \times \Pi \times \overline{\Pi} \to \Theta$.
7. Перестановка проекции с селекцией: $C_4^{-1} : \Theta \times \Pi \times \Sigma \times \overline{\Pi} \to \Theta$
8. Удаление селекции $s \in \Sigma$ из реляционного выражения q, если она не связана ни с одним логическим выражением: $C_5 : \Theta \times \Sigma \to \Theta$.
9. Перестановка селекции с декартовым произведением: $C_6 : \Theta \times \Sigma \times D \times \overline{\Sigma} \times \overline{\Sigma} \to \Theta$.
10. Перестановка селекции с объединением: $C_7 : \Theta \times \Sigma \times U \times \overline{\Sigma} \times \overline{\Sigma} \to \Theta$.
11. Перестановка селекции с разностью: $C_8 : \Theta \times \Sigma \times R \times \overline{\Sigma} \times \overline{\Sigma} \to \Theta$.
12. Перестановка проекции с декартовым произведением $C_9 : \Theta \times \Pi \times D \times \overline{\Pi} \times \overline{\Pi} \to \Theta$
13. Перестановка проекции с объединением типа $C_{10} : \Theta \times \Pi \times U \times \overline{\Pi} \to \Theta$.

3. Формализация алгоритма оптимизации реляционного выражения

Приведем известный алгоритм преобразования реляционного выражения [4]. Каждый этап алгоритма основан на последовательном применении законов эквивалентности к реляционному выражению. Хотя законы эквивалентности формальны, порядок их применения задан вербально, а само применение требует участия человека. Формализация алгоритма оптимизации позволит значительно упростить его реализацию на ЭВМ.

Вербально алгоритм выглядит так:

1. Представляем каждую селекцию в виде каскада селекций.
2. Перемещаем каждую селекцию в сторону большей вложенности.
3. Перемещаем каждую проекцию в сторону большей вложенности, насколько это возможно. При этом проекции могут исчезать и расщепляться.
4. Комбинируем каждый каскад селекций и проекций в одиночную селекцию, одиночную проекцию или селекцию с последующей проекцией.

Формализуем этот алгоритм. На основе уже созданных операций над реляционным выражением построим совокупность операций одного типа: $\Theta \times O \to \Theta$. Каждая такая операция применяется к некоему элементу $o \in O$ из реляционного выражения $q \in \Theta$, и выполняет то же преобразование реляционного выражения, которое она выполняла ранее.

В алгоритме используется следующее множество операций типа $\Theta \times O \to \Theta$: $\{C_2, C_3, C_3^{-1}, C_4, C_4^{-1}, C_6, C_7, C_8, C_9, C_{10}\}$.

Зададим с их помощью укрупненные операции, которые будут представлять собой одну итерацию второго, третьего и четвертого этапов в вербальном описании алгоритма. Одна итерация первого этапа есть не что иное, как операция C_3^{-1}.

Определим операцию перемещения селекции o в сторону большей вложенности в реляционном выражении q :

$$C_{11}(q,o)=\begin{cases}C_4(q,o),\arg_q o\in\Pi;\\C_6(q,o),\arg_q o\in D;\\C_7(q,o),\arg_q o\in U;\\C_8(q,o),\arg_q o\in R.\end{cases}.$$

Ее область определения:

$$f(C_{11},q,o)=o\in\Sigma\wedge\arg_q o\in\Pi\cup D\cup U\cup R.$$

Определим операцию перемещения проекции o в сторону большей вложенности в реляционном выражении q :

$$C_{12}(q,o)=\begin{cases}C_4^{-1}(q,o),\arg_q o\in\Sigma;\\C_9(q,o),\arg_q o\in D;\\C_{10}(q,o),\arg_q o\in U.\end{cases}$$

Область определения:

$$f(C_{12},q,o)=o\in\Pi\wedge\arg_q o\in\Sigma\cup D\cup U.$$

Определим операцию агрегирования проекции или селекции o в реляционном выражении q : $C_{13}(q,o)=\begin{cases}C_2(q,o),\arg_q o\in\Pi;\\C_3(q,o),\arg_q o\in\Sigma.\end{cases}$.

Область определения:

$$f(C_{13},q,o)=(o\in\Pi\wedge\arg_q o\in\Pi)\vee(o\in\Sigma\wedge\arg_q o\in\Sigma).$$

Каждый этап алгоритма представляет собой многократное повторение операций из следующего множества: $V=\{C_3^{-1},C_{11},C_{12},C_{13}\}$.

Зададим общий предикат области определения для всех операций из V . Он имеет тип $V\times\Theta\times O\to B$ и записывается в функциональной нотации как $f(v,q,o)$. Этот предикат истинен, если операция $v\in V$ применима к реляционной операции o в выражении q .

В алгоритме многократно применяется одна операция к реляционному выражению до тех пор, пока оно находится в области определения этой операции. Причем для алгоритма не существенно, в каком порядке применять ту или иную операцию.

Определим операцию $\min_v q$ выбора такой операции o из выражения q , для которой применима операция $v\in V$:

$$\min_v q=\{x\mid x\in O\wedge f(v,q,x)\wedge\forall o[o\in O\wedge f(v,q,o)\to x<o]\}.$$

Определим операцию $\Re$ многократного применения к реляционному выражению другой операции V , пока в выражении остается хотя бы один элемент, к которому можно применить операцию v :

$$\Re_v q=\begin{cases}q,\overline{\exists o(f(v,q,o))};\\\Re_v v(q,\min_v q).\end{cases}$$

Рассматриваемый алгоритм представляет собой последовательность этапов. К реляционному выражению применяется некоторая операция до тех пор, пока это выражение не выйдет за пределы области определения операции, то есть $\Im(q)=\Re_{C_{13}}\Re_{C_{12}}\Re_{C_{11}}\Re_{C_3^{-1}}q$.

Таким образом, получено формализованное представление алгоритма оптимизации реляционного выражения, который может быть реализован в виде программного обеспечения для генерации оптимальных запросов в рамках информационных систем, функционирующих на предприятиях.

Литература

1. МУТУШЕВ Д. М., ФИЛИППОВ В. И. *Объектно-ориентированные базы данных* // Программирование. 1995. № 6. С. 59 – 76.
2. НОВИКОВ Б. А. *Индексирование во временных базах данных* // Программирование. 1995. № 2. С. 31 – 36.
3. ЗАДОРОЖНЫЙ В. И., НИКИТЧЕНКО Н. С. *Алгебраический подход к формализации дедуктивных языков запросов* // Программирование. 1992. №6. С. 29 – 47.
4. УЛЬМАН ДЖ. *Основы систем баз данных*. Пер. с англ. – М.: Финансы и статистика, 1983. – 334 с.
5. SI A., LEONG H. *Query optimization for broadcast database* // Data and Knowledge Engineering. Vol. 29. 1999. № 3. P. 351 – 380.