

数据库交互式条件检索技术

盛福杰 (中科院金属所计算机应用开发部)

摘要:本文分析了目前几种常用的数据库检索方法,并指出了不足之处,然后介绍一种实用灵活的数据库交互式条件检索方法。

一、微机关系数据库检索方法概述

数据库检索方式大体分为两类:一类是带索引文件的数据按关键字快速查询,如 FoxBASE 中的 Find, Seek 命令与其它命令的联合使用;它在管理信息系统中是必要的,但不是充分的,因为它只能在索引关键字字段上检索与关键字相匹配的记录,检索路径单一死板,不能满足较复杂的检索功能。

另一类是对关系数据库按照一个算术逻辑关系表达式为条件进行检索,称做条件检索。在 FoxBASE 中可用命令 Locate for <条件表达式> 和 Continue 语句联合实现。或用 Set filter to <条件表达式> 对数据库全体记录进行限定,然后用适当方式输出这些满足条件的记录。条件检索是按照由属性名(字段名)、常量以及算术关系逻辑运算符等组成的条件表达式来检索满足条件的所有元组(记录),不需要建立或打开索引文件,免去了空间和时间的巨大开销,使得检索路径达到最多,所有字段均可做为检索条件,能满足“任意的”逻辑表达式的检索,当然这是我们所追求的理想状态了,这就是检索条件表达式是否能灵活组织的重要性之所在,也是本文的核心内容。由于检索条件表达式组织手段和方法的不同,使得检索方法和功能也大相径庭。总之,在条件检索中,条件表达式组织手段的合理性和灵活性将直接影响检索中的查全率、查准率乃至整个系统的效率和质量。

1. 固定式组合检索

日期_____凭单号_____经手人_____
摘要_____科目码_____课题码_____
借额_____贷 额_____余 额_____

图 1 固定式组合检索示意图

该方法是将关系数据库字段名显示于屏幕,由用户输入对应的样板值(见图 1),其中的样板值可有空缺项。

然后程序输入的样板值与对应的字段名转换成检索条件式表达式形式如下(实际的可能要短些):日期=样板值 1.AND.凭单号=样板值 2.AND.经手人=样板值 3.AND.摘要=样板值 4.AND.科目码=样板值 5.AND.课题码=样板值 6.AND.借额=样板值 7.AND.贷额=样板值 8.AND.余额=样板值 9

可见表达式中出现的表达式运算符只有逻辑运算符 .AND. 和关系运算符 =, 当然上面的程序也可演变出多个程序,使生成的条件表达式中含有逻辑运算符 .OR. 及 .NOT. 和关系运算符 >, <, >=, <=。但同时只能出现一种关系和逻辑运算符,这是由于通向用户的接口是固定的格式的。这就使得检索条件显得呆板,缺乏灵活性和适应性。

表 1 交互式组合检索属性数据词典

元组	属性号	连接符	逻辑符	属性条件	提示命令
001	01	1	.AND.	日期=变量 1	I,J SAY"日期"GET
002	01	2	.OR.	(同上)	变量 1 (同上)
003	02	1	.AND.	凭单号=变量	I,J SAY"凭单号"GET
004	02	2	.OR.	2 (同上)	变量 2 (同上)
:	:	:	:	:	:
:	:	:	:	:	:
017	09	1	.AND.	余 额 = 变 量	I,J SAY"余 额"GET
018	09	2	.OR.	9 (同上)	变量 9 (同上)

2. 交互式组合检索

该方法(参考文献 2)生成的表达式中可以同时出现逻辑运算符 .AND. 和 .OR. 但关系运算符还只能是 = 号。

首先应根据主数据库的属性手工构造出属性数据词典(见表 1),

为使用户输入检索条件方便,每个属性名对应一个属性号,逻辑运算符.AND.和.OR. 分别定义成连接符 1 和 2,同时每个属性名对应一个提示命令,供用户输入样板值。具体步骤是:

(1)由用户一次选择一个属性号和一个连接符,系统程序查一次表 1,得到一个属性名、运算符和提示命令,这一过程可重复多次,直到停止选择;

(2)系统根据步 1 得到的提示命令执行,由用户输入相应属性的样板值;

(3)系统程序将步 1 和步 2 得到的属性名, 样板值和逻辑运算符组合成一个 FoxBASE 可以识别的条件表达式。例如该方法可以检索如下条件式:日期=920102.AND. 凭单号=118.OR.余额=10000

与固定式组合相比,该方法引进了或(.OR.)运算,是一个改进,但改进不大。为此,有必要探求新的检索方法,使得条件式能包含所有运算符,可以是最简表达式,并且让用户输入输入的字符最少,操作最灵活,以下给出一个交互式条件检索的方法。

二、交互式条件检索

FoxBASE 的检索条件表达式可以由以下各项组成:

- 算术运算符集{+,-, *, /, * *, (,)};
- 逻辑运算符集{.and.,.or.,.not.,(,)};
- 关系运算符集{<,<=,>,>=,<>};
- 字符串运算符集{+,\$};
- 各种类型函数。
 - 被检索数据库属性(字段)名;
 - 各种类型的样板值(常量)。

交互式条件检索的基本思想是:允许以上各项在条件表达式中出现,使得关系数据库的优点得以充分体现;数据库属性名可对用户完全透明;尽量减轻用户输入条件式的负担,以上较复杂的项可以用简短的字符代替,由程序自动识别转换,并在屏幕固定位置显示;输入完的条件式还可由用户修改。具体方法如下。

1.构造数据词典

构造数据词典的目的是简化用户的输入,使字段名和复杂的运算对用户透明。

假定主数据库关系模式为:

日期	凭单号	经手人	摘要	科目码	课题码	借额	贷额	余额
----	-----	-----	----	-----	-----	----	----	----

不妨定义该库(Main.dbf)字段名为:

rg	pdh	jsr	by	kmm	ktm	je	de	ye
----	-----	-----	----	-----	-----	----	----	----

建立数据词典如表 2 所示。其中运算符栏可由字段名、复杂运算符和函数名组成;代码和提示符显示于屏幕供用户选择输入,用户只需输入代码,由程序将其转换成相应算符,组成检索条件式。该数据词典由一个数据库构成,建立和程序转换十分方便。词典内装入的函数名的数量可根据具体应用而定。系统也可提供让有经验的用户修改词典的功能模块。

表 2 交互式条件检索数据词典(部分)

代码	算符/字段/函数	提示符
^	.AND.	并且
@	.OR.	或
~	.NOT.	非
#1	rq	日期
#2	pdh	凭单号
#3	jsr	经手人
#4	zy	摘要
#5	kmm	科目码
#6	ktm	课题码
#7	je	借额
#8	de	贷额
#9	ye	余额
#10	abs	绝对值
#11	chr	取字符
#12	int	取整
#13	log	取对数
:	:	:

2.检索表达式的生成

具体过程是:

(1)显示数据词典记录的代码和提示符字段,供用户选择输入代码。

(2)用户可以输入任意字符串:字典中的代码、常量(如果是字符型应在两端加引号)、各种运算符、运算符与常量的组合、合法的表达式或一部分。

(3)程序对步 2 输入的字符串到词典库快速查询是

否为一个代码,是:转换成相应算符并连接到表达式尾部;
否:直接将字符串连接到表达式尾部;如果是结束符!将结束
表达式的输入,转 5

(4)重复步 2 和步 3,并显示当前表达式内容,显示内
容中包含字典算符的部分用相应提示符替代,使用户一
目了然,容易理解。

(5)允许用户修改已输入的条件表达式。

程序流程图见图 2。由以上还可看到:条件式的输
入是按从左往右的自然顺序一次输出;由于括号的使用
可以将表达式化为最简式后输入,从而逻辑关系明了,输
入简洁;随着用户使用经验的丰富,即使同样长的条件式,
上述步 2 到步 3 的循环次数会日趋减少。

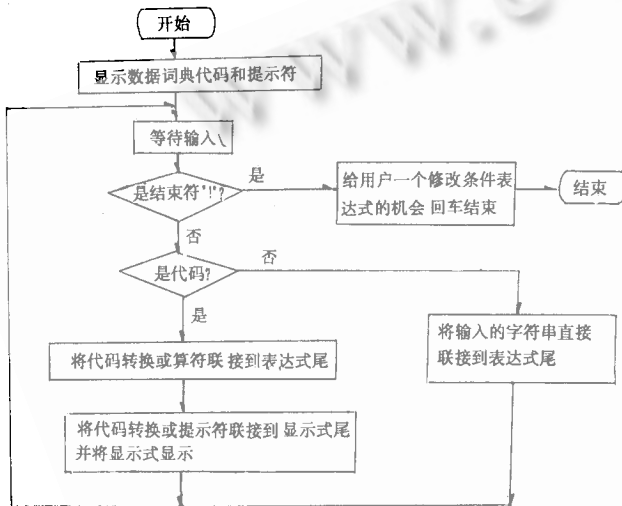


图 2

交互式条件检索条件表达式生成程序如下。

程序清单:

```
*.===== 人机交互生成检索条件表达式,存入变量,
*===== 供检索命令使用(可用宏替换方式).
clear
set talk off
set echo off
use dict inde dict &&打开数据词典库和索引
condi = "" &&存放待生成检索条件表达式变量
sub__cond = space(20) &&接收输入字符串变量
```

dis__cond = '' &&直观显示类表达式变量

```
@ 1,10 say "***** 请输入检索条件
*****"
@2,10 say "注:a.以下项目和算符可用代码代替输入"
@3,10 say " b.代码一次只能输入一个(即代跟回车)"
@4,10 say " c.除代码以外每次输入字符个数不限"
@5,10 say " d.样板值如果是字符型两端应加引号"
@6,10 say " e.结束输入敲!键"
@7,10 say "#1 -> 日期(字符型) #2 -> 凭单号"
@8,10 say "#3 -> 经手人(字符型)#4 -> 摘要(字符型)"
@9,10 say "#5 -> 科目码(字符型)#6 -> 课题码(字符型)"
@10,10 say "#7 -> 借额 #8 -> 贷额"
@11,10 say "#9 -> 余额 #10 -> 决对值"
@12,10 say "#11 -> 取字符 #11 -> 取整"
@13,10 say "#12 -> 取字数 (a -> 或"
@14,10 say "-> 并且 - -> 非"
@15,10 say "其它运算符或函数可直接输入,例如:"
@16,10 say "+ - * / ( ) < = > 等"
do while .t.
```

```
@18,1 say "请输入字符串:" get sub__cond
read
if sub__cond = trim(sub__cond)
find &sub__cond
if .not.eof()
condi = trim(condi)+name
dis__cond = trim(dis__cond)+mean
else
condi = trim(condi)+sub__cond
dis__cond = trim(dis__cond)+sub__cond
endif
sub__cond = space(20)
@19,1 say "已输入的检索条件是:"dis__cond
enddo
condi = condि+space(10)
@20,1 say "可以修改已输入的条件表达式(回车结束)"
@21,20 get condi
read
```

参考文献:

[1]周秀菊,盛福杰:提高数据库检索速度的方法,
<<计算机应用>>,Vol. 10, No. 5,1990
[2]陈永平,杨娇英:数据库交互式组合检索,<<计
算机世界月刊>>,No.8,1989