

基于 ORACLE 的三维报表研究

车文富 (北京系统工程研究所 100101)

摘要: 本文根据二维报表制作原理对三维报表的制作进行分析与研究, 给出了三维报表通用程序设计的方法和技巧; 对实现中遇到的几个实际问题进行剖析并最终加以解决。最后列举实例证明该种方法的可行性。

关键词: ORACLE 数据库 报表生成 电算化 制作

一、问题的提出

报表制作的通用程序设计一直是人们倍加关注, 投其精力以期彻底解决的问题, 也是开发 MIS 系统研究和争论的一个热点。目前对二维报表的研究已取得了一定的成果。但对三维报表, 研究的很少, 更不用说如何解决了。

1. 何谓“三维”报表?

所谓三维报表是相对二维报表而言的。对一般的二维报表, 只有一个字段是不确定实例字段简称不确定字段。其特点是: 不确定字段描述的往往是客观世界的实体, 其实例全集是确定的, 然而由某时某刻的实际情况而定的实例子集是不确定的, 此子集又恰恰构成报表数据处理的作用域, 并决定报表纵向的长度及风格。例如: 对库存日报表, “品名”字段是不确定字段。仓库允许存放的物品是确定的, 而某时某刻仓库存放的具体物品又是

不确定的, 它取决于当日库存的实际情况; 而其他字段, 如“昨日结存”、“今日入库”、“今日出库”和“今日结存”是对“品名”字段各个实例具体行为的描述。抽象上, 可将不确定字段(品名)定义为 X 轴, 确定字段定义为 Y 轴, 构成二维空间表格形式。

三维报表是二维报表的扩充。它有两个不确定字段, 第一个不确定字段定义为 X 轴, 第二个定义为 Y 轴, 确定字段定义为 Z 轴, 便构成一个三维报表。例如: “车间、班次产量统计日报表”即是一三维报表, 其中不确定字段“车间”和“班次”分别为 X 轴和 Y 轴, 确定字段“产量”为 Z 轴。

2. 三维报表制作的难点和关键问题

三维报表是建立在二维报表制作的基础上, 二维报表制作存在的问题在三维报表制作中都存在。此外, 必须解决如下问题



- (1) 不确定字段的循环嵌套控制;
- (2) 数据产生公式的保留及格式文件的重构;
- (3) 报表数据与表格的拼接。

二、二维报表制作

三维报表的制作建立在二维报表制作的基础上。在研究三维报表制作之前,不妨先看看二维报表的制作方法。

1. 二维报表的设计目标

(1) 制作的通用性。二维报表的制作不是按某一企业、某一部门固定模式设计的,它是在对企业各类报表从形式到内容经过深入分析、抽象、归纳和总结产生的,它的实现能满足企业各类报表和单据(如发票)制作的需要。

(2) 使用的灵活性。制表软件既可单独使用,又可作为过程为其他工具调用。如:sql * forms、pro * c,体现出简单方便、多样化的使用特点。

(3) 功能齐全且有较强的扩充能力。本软件可使用 sql * plus 中的任何形式的 select 语句作为数据查询公式,及 ORACLE 中的任何函数对数据进行统计和运算。此外,该软件留有库连接接口,用户可以连接自己写的 C 语言函数库,针对特定的功能需求进行扩充。

(4) 软件运行快速、稳定、可靠。

2. 二维报表实现的基本原理

二维报表的制作是采用原型法实现的。对用户而言,只需按规则提供实际报表的格式原型文件即可。原型文件有两部分:第一部分描述数据的生成公式;第二部分描述表格结构以及数据在报表中的存放位置。报表格式原型文件可用目前流行的任一编辑软件生成。

对报表制作软件,首先读取报表格式原型文件,对其中的各部分内容进行截取和分析,确定各数据的长度及在报表中的存在位置,并提炼出数据产生公式,即 select 语句。然后执行 select 语句,并对求得的数据进行运算,产生报表所需的数据全部。最后,将结果数据与报表结构相拼接,形成最终可打印的报表文件。

二维报表的制作原理详见本人的另一篇论文[1]。

3. 二维报表制作软件实现的特点

(1) 数据处理能力强、使用简单方便。

(2) 报表数据可存放在不同的表、数据库或不同的地点。select 语句具有多表查询功能和高级查询功能。此外,借助网络支持(如 TCP/IP 或 Novell),可实现不同结点的数据查询和统计操作。

(3) 软件运行速度快、报表调试效率高。该软件可直接读入报表格式原型文件,不需重新编译和连接,且经处

理可立刻产生报表结果,其风格由用户在格式文件中指定。

(4) 软件生成的报表结果存入中间文件,用户可对报表结构或数据进行调整,满意后再打印输出。

(5) 软件即可单独使用,也可嵌入其他工具中使用。

4. 软件的使用

二维报表制作软件自实现以来已在企业中广泛使用。如山东省烟草公司的青岛卷烟销售公司,其下属所有部门的报表及单据(如发票)都使用本软件来制作。从使用情况看,该软件功能齐全、使用方便、适应性强、快速稳定,得到使用者的充分肯定和好评。

三、三维报表的实现原理

1. 三维报表分析

三维报表与二维报表一样,从结构上可分四部分:

标题——报表名称或题头,以及附带的装饰线。

表头——标题下面的说明信息。如编制单位、日期及报表字号等;

表体——报表的栏目结构及所包含的数据内容。

表尾——表体下面的附加说明及顺序页号。

其中,表体是报表制作的核心。三维报表的表体与二维报表类似,也分为单记录、多记录和混合记录三种形式[1]。多记录制作是解决问题的关键,但它比二维报表包含更多的不确定字段,因而更复杂。

2. 三维报表的实现策略

三维报表的制作要解决不确定字段嵌套的二重循环,比二维报表更复杂。在实现上要借助二维报表的制作原理和方法并对循环字段变量施以一定的控制技巧。

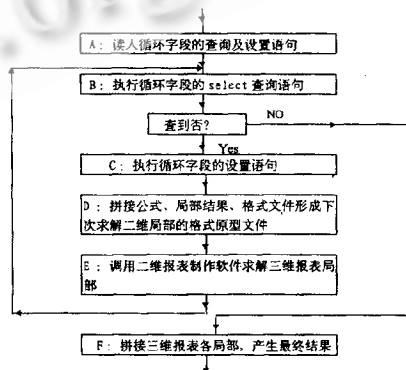


图1 三维报表制作的控制流程图

三维报表含有两个不确定字段,但将其中之一代之以实例,则三维就转化为对其余二维的求解,利用二维报

表制作方法,可求得三维报表的局部。在循环中,不断地替换外层不确定字段的值即实例,可依次求得三维报表的各个局部,拼接各局部,则形成三维报表的整个结果。

为了实现上述策略,软件设置另一入口,要求用户提供第一个不确定字段,即外部循环字段的查询语句及存放语句。存放位置由软件事先规定。如此读入并执行这两个语句,即可构成二重循环的外层循环控制。

3. 三维报表制作的控制过程(如图 1)

四、几个关键技术问题的解决

1. 外部循环变量(字段)的设置

三维报表制作的关键是如何解决外部循环变量,即不确定字段的设置与控制。本软件设置固定的表存放用户提供的循环控制变量的值。格式如下:

表名 d3d-tmp:

a1	a2	a3	a4	a5
char(10)	char(10)	char(10)	char(10)	char(10)

设置五个字段是因为有些不确定字段需带有编码来控制表字段的排序。相应地,传递循环字段的实例到固定表中的主变量依自然顺序设为 v1、v2、v3、v4 和 v5。

前例“车间、班次产量统计日报表”的入口文件 3d-from 如下:

```
select distinct(cjmc)
from cl
order by cjmc
#####
update d3d-tmp
set a1 = :v1
where a2 = 'cjcl-key'
#####
这里车间产量统计日报表的键值 a2 设为 cjcl-key。
```

软件读入此文件并执行,由主变量 v1 将查询到的车间名称存放到表 d3d-tmp 的 a1 字段。

2. 格式原型文件各“部件”的分割、存放及拼接

三维报表与二维不同,它必须多次执行循环内的 select 语句,且每次执行都涉及 select 语句、当前查询结果及格式文件部分内容的存放及修改以及如何合理准确地拼接形成下一次求解所需的格式原型文件。这是循环内部需要解决的一个首要问题。

为此,软件设置一个字符串数组 chs-array[100],并设变量指针分段存放格式原型文件的各部分内容:

chs-array[0..select-t]:数据查询公式部分;
chs-array[select-t+1..mark-h-1]:表头部分;

chs-array[mark-h..mark-t]:数据存放区域部分;

chs-array[mark-t+1..file-t]:报表结尾部分。

同时在循环处完成:

(1)在进入循环之前(图 1 中 A 处)

·复制 chs-array[select-t+1..mark-h-1]到文件 to;

(2)查询外部循环变量成功后(图 1 中 D 处)

·连接 chs-array[0..select-t]到格式文件 from;

·连接 out-tmp 到格式文件 from;

·连接 chs-array[mark-h..mark-t]到 from。

(3)调用二维制表软件,制作结果存入文件 to。(图 1 中 E 处)

(4)查询外部循环变量失败后(图 1 中 F 处)

·复制临时结果文件 from 到结果文件 to;

·连接 chs-array[mark-t+1..file-t]到结果文件 to。

3. 二维报表制作软件命令文件序列的构成及调用

三维报表制作是利用二维报表制作软件完成的。执行中调用二维制表软件的命令如下:system("2d.bat");

此处二维报表制作是以批处理的命令形式存在的,它由处理各种表体数据的处理模块及相应的操作系统命令构成。详见后例。

五、三维报表制作举例

以前述的“车间、班次产量统计日报表”为例,说明三维报表的制作步骤:

设产量 cl 的库结构如下:

车间名称	班次	品名	生产数量	生产日期
cjmc(char)	bc(char)	pm(char)	cpsl(number)	rq(char)

该报表初始定义如下:

Input = (from);格式原型文件为 from

Output = (to);结果文件为 to

3input(3d-from);入口文件为 3d-from

(1)制作“车间、班次产量统计日报表”95 年 7 月 5 日的格式原型文件 1d-from 如下:(日期可设置成参数由系统赋值,此处简化)

```
select 1, to-char(sum(cpsl), '999999')
from cl
where rq = '95.07.05'
#####
select 1, a1
from d3d-tmp
where a2 = 'cjcl-key'
#####
```

```
select 1, to-char(sum(cpsl), '99999999')
from cl
where rq = '95.07.05'
and cjmc = (select a1 from d3d-tmp where a2 = 'cpcl-
key')
#####
select bc, to-char(sum(cpsl), '99999999')
from cl
where rq = '95.07.05'
and cjmc = (select a1 from d3d-tmp where a2 = 'cpcl-
key')
group by bc order by bc
#####
```

车间、班次产量统计日报表

日期: \$ date 单位: 台

车间名称	车间班次	产品数量
! s2		
	&s1	&s2
车 间	小 计	@s2
总 计		\$ s2

页号: \$ pg 制表人: 赵值

(2)构造三维报表制作批命令文件 sample.bat 如下:

```
copy 1d-from from
cl-djl $
copy to from
ccl-dddjl ! @ &
```

(3)构造二维报表制作批命令文件 2d.bat, 该文件由三维报表制作软件 cl-dddjl 调用。

```
copy 2d-from from
cl-djl !
copy to from
cl-hjil @
copy to from
cl-ddjl & 0
```

(4)构造入口格式文件 3d-from。内容见 4.1。

(5)执行 sample.bat 产生报表最终结果如下:

车间、班次产量统计日报表

日期:1995.07.05 单位:台

车间名称	车间班次	产品数量
一车间		
	甲班	333
	乙班	679
	丙班	1099
车 间	小 计	2111
二车间		
	甲班	456
	乙班	632
车 间	小 计	1088
三车间		
	甲班	124
	乙班	126
	丙班	11
	丁班	12
车 间	小 计	273
	总 计	3472

页号:1 制表人: 赵值

六、结束语

本文给出的三维报表的制作方法是建立在二维报表制作的基础上实现的,因此也继承了二维报表制作软件通用性、灵活性及使用方便等优点。这在实践中得到了证明。同时,利用本文给出的思想并借助已实现的解决各种报表数据类型的程序处理模块可以研究和实现“多维”报表的制作,从根本上解决报表的电算化制作问题。

参考文献

[1] 车文富“基于 ORACLE 数据库的通用报表自动生成系统的研制和实现”。国防科工委指挥技术学院学报, 1995 年第一期。

[2] 杜小勇, 刘志斌著. 《ORACLE 5.10 版关系数据库教程》中科院希望高级电脑公司, 1990。

[3] Microsoft C5.0 技术丛书. 北京联想计算机集团, 1990。