

CYBER

记录管理

COBOL 用户手册

地质矿产部石油地质海洋地质局 编
北京数据库培训班

一九八三年三月

编者的话

为了进一步开发 COBOL 语言的应用,配合培训石油地质数据库专业人员,我们根据《CYBER RECORD MANAGER GUIDE FOR USERS OF COBOL》,以及《COBOL 4 REFERENCE MANUAL》的部分章节编译了《CYBER记录管理COBOL用户指南》一书。全书共分十章,第一、二章叙述CYBER记录管理的主要概念;第三到第八章,介绍用COBOL语言建立和使用六种文件的方法;第九、十章,介绍COBOL程序编译和执行时要用到的一些控制语句。

本书中有较多的程序实例,适合于具有中等水平的,并对CDC系统有所了解的COBOL程序员阅读。对COBOL程序员、信息处理和数据库专业人员有一定的参考价值。

本书由地质矿产部石油地质海洋地质局北京数据库培训班编印。由于水平所限,本书一定存在不少缺点和错误,尚望读者不吝批评指正。

地矿部北京计算中心提供了本书的原文资料,在组织编译、印刷、发行上都予以大力的支持,并承担了许多颇费心力的工作。因此,本书能够与大家见面,与他们的关心和努力是分不开的。对此谨致深切的谢意。

一九八三年三月

原书缺页

原书缺页

原书缺页

原书缺页

原书缺页

原书缺页

表 2—2 说明匹配各种文件组织的记录类型。

表 2—2 记录类型和文件组织的匹配情况

文件组织	记 录 类 型					
	D	F	R	T	W	Z
顺 序	×	×	×	×	×	×
相 对		×				
标 准					×	
直 接	×	×	×	×	×	×
索引顺序	×	×	×	×	×	×
实 际 键	×	×	×	×	×	×

6. 1 十进制字符计数记录 (D型)

D型记录是可变长度记录，其长度通过每个记录中的一个字段确定。采用如下形式的 RECORD CONTAINS子句来规定D型记录：

```
RECORD CONTAINS  整数—1
TO  整数—2  CHARACTERS
DEPENDING ON  数据名
```

其中整数—1是按字符计数的最小记录长度，整数—2是按字符计数的最大记录长度。

DEPENDING ON 指出的数据名就是01层描述体中基本项的名称。它一定是一个 DISPLAY或COMPUTATIONAL—1项。在图 2—2 这个例子中，RLE 便是 DEPENDING ON子句中的数据名，它被定为数据记录PAYREC的第二个基本项。

FD PAYROLL
LABEL RECORDS OMITTED
RECORD CONTAINS 11 TO 80
CHARACTERS DEPENDING ON RLE
DATA RECORD IS PAYREC
01 PAYREC.
02 SOC-SEC-NO PIC 9(9).
02 RLE PIC 9(2).
02 NAME-ADDRESS.
03 N-CHARACTER PIC X
OCCURS 69 TIMES.

图 2—2 D型记录

由DEPENDING ON 定出的数据名必须要放在记录的定长部分内,且其长度不得超过六个字符。在记录输出之前COBOL程序会把数据名变成合适的记录长度。如果它包含的整数超出了RECORD CONTAINS 子句规定的范围,则执行一个输入/输出语句,发出一诊断信息。并且中止程序。如果把上面例子中的DEPENDING ON RLE 去掉,编译程序将会根据对PAYREC的01层描述体来计算,认为它是F型记录,其固定长度为80个字符。

6.2 定长记录 (F型)

定长记录都包含同样数目的字符。当一个文件的数据部分描述体没有给出这个文件里的记录长度是可变的情况下,也就是说,把DEPENDING ON选择从文件描述体中的RECORD CONTAINS子句中去掉(或把整个子句都去掉),并把OCCURS.....DEPENDING ON选择从记录描述体中也去掉之后,COBOL将选择此种记录类型。

COBOL在进行编译时就计算记录文件中所有记录的长度,求出文件01层记录描述体中所有项目的和,就是记录长度。如果同一文件中不同01层描述体规定的记录格式长度不一,则使用最长记录作为文件中所有记录的长度。在这种情况下,文件中所有记录格式都使用同样输入记录区和输出记录区,最大组项的长度就是文件工作存储区的长度。

如果去掉RECORD CONTAINS 子句,或采用以下形式,COBOL便选择F型记录:
RECORD CONTAINS 整数 CHARACTERS

或

RECORD CONTAINS 整数—1 TO 整数—2
CHARACTERS

计算记录长度并不使用这里规定的整数值,但可以包括它们用于公文陈式。如果使用下面这个格式:

RECORD CONTAINS 整数 CHARACTERS

并且01层描述体有不同长度,则会导致出错。

在图2—3这个例子中,文件PAYFILE里的每个记录的固定长度被计算为80个字符,但并没有使用RECORD CONTAINS 规定。

```
FD PAYFILE
   LABEL RECORDS OMITTED
   RECORD CONTAINS 209 CHARACTERS
   DATA RECORD PAYREC.
01 PAYREC.
   02 SOC-SEC-NO PIC9(9).
   02 RATE PIC 9(3)V99.
   02 HRS PIC 99V9.
   02 FILLER PIC X(63).
```

图2—3 F型记录

6.3 记录标志型记录 (R型)

记录标志型记录可具有各种长度, 其结尾用一专门字符。如果FILE控制语句没有使用RMK参数(见第九章), 规定了其他标志的话, 则使用右中括号“)”字符作为记录标志。

要想规定记录标志, 程序员应使用象征常字RECORD-MARK, 格式如下:

```
RECRD CONTAINS 15 TO 80
CHARACTERS DEPENDING ON
RECORD-MARK
```

然后在这个记录的WRITE语句之前用一个MOVE语句把记录标志字符放到记录的最后字符位置上。

在图2—4这个例子中, 记录标志的位置由以下描述项规定:

```
03 N-CHARACTER PIC X
OCCURS 66 TIMES
```

设置记录标志, 应使用如下形式的语句:

```
MOVE RECORD MARK
TO N-CHARACTER (sub)
```

上述语句要用在写这个记录之前, 其中sub是一个标识符, 或是一个常数, 其值在1—66之间。

```
FD PAYROLL
  LABEL RECORDS OMITTED
  RECORD CONTAINS 15 TO 80
  CHARACTERS DEPENDING ON
  RECORD-MARK
  DATA RECORD IS PAYREC.
01 PAYREC.
  02 SOC-SEC-NO PIC 9(9).
  02 EMPLOYEE-NO PIC X(5).
  02 NAME-ADDRESS.
  03 N-CHARACTER PIC X
    OCCURS 66 TIMES.
```

图2—4 R型记录

6.4 尾部计数记录 (T型)

尾部计数记录包含一个定长头部, 随之是一些定长的尾部项, 这些定长尾部项的数目不定。当在01层记录描述体中规定了COBOL子句:

```
OCCURS 整数—1 TO 整数—2
```

TIMES DEPENDING ON 数据名
的时候,就产生了尾部项。它们的实例如图2—5所示。

```
01 WEEKLY-OVERTIME-HISTORY.  
    02 SOC-SEC-NO PIC 9(9).  
    02 EMP-NO PIC X(5).  
    02 WEEK-COUNT PIC 99.  
    02 WEEKLY-ENTRY OCCURS  
        1 TO 52 TIMES DEPENDING ON  
        WEEK-COUNT.  
    03 HRS-WORKED PIC 99V9.  
    03 RATE PIC 99V99.
```

图2—5 T型记录

OCCURS子句规定的数据项(例子中的WEEKLY-ENTRY)被称为尾部项,在实际记录中这些项目的出现数目不一,长度相等,但内容不同。在DEPENDING ON选择中提到的数据名(例子中的WEEK-COUNT)被称为尾部计数字段,它必须是一个DISPLAY或COMPUTATIONAL—1的基本项。在填写一个记录时,尾部计数字段中的值确定记录中尾部项目的数目。这个数目和头部长度(头部包括尾部项目之前的所有数据项)一起决定记录的长度。

6.5 控制字记录(W型)

控制字记录前部有一个系统生成的标题字,它指出这个记录的长度。这种记录可以是任意长度,其大小变化是在01层文件描述体里记录描述项中确定。控制字只能由CYBER记录管理来处理,它是在写记录的时候产生的。然后被放在记录的前部。当用户发出一个读请求时,这个控制字在记录返回用户之前被切除。

定义控制字记录要使用一个后缀——FW,接在ASSIGN子句的设备名后边,例:

```
SELECT PAYROLL-FILE  
ASSIGN TO PAYFILE-FW
```

当规定为控制字记录时,一定要在文件描述体中包括下面这个子句:

```
RECORDING MODE IS BINARY
```

具有W型记录的文件可以使用多种记录格式,这些格式可以是定长的或可变长度的。所写的每个记录的长度就是格式要求的长度。这些记录上能被装填到下一个字的边界为止。

6.6 零字节记录(Z型)

由COBOL程序员规定的零字节记录为定长记录,但以压缩形式存贮。CYBER记录管理用一个零字节结束符号(由12个位零组成,在一个中央存贮字中向右对齐)表示这种记录的结束,并能自动去掉任何尾部空格以便缩减大容量存贮设备的空间。在检索这种记录时,又能恢复空格,使记录原样返回用户程序。

当从索引顺序文件、直接存取文件或实际键文件中读出记录时,则不能恢复空格。如果被读出记录小于输入记录区,记录区剩余部分的内容将是不相干的。在这种情况下,程序将在每次读之前将输入记录区清零,或用一些其他方法来保证记录的完整性。

凡是由卡读机读出的文件,或是规定要到行打机或卡片穿孔机进行输出的文件,都必须使用零字节记录。分配给ASSIGN子句中INPUT、OUTPUT或PUNCH的文件不需要什么其他说明来表示它们是零字节记录。COBOL编译程序会自动地把它们作为Z型记录处理。除INPUT、OUTPUT和PUNCH外,其他文件如要规定为零字节记录,则可使用后缀—FZ,把它加在SELECT……ASSIGN里的设备名之后,例:

```
SELECT ZFILE
ASSIGN TO PAYFILE-FZ
```

任何文件只要其记录最初是通过卡读机读入的,而且它的设备名不是INPUT(比如是从INPUT文件复制来的),就有必要把—FZ添到设备名上。另外,ACCEPT或DISPLAY语句的UPON部分提到的任何文件也必须要有零字节记录,亦可通过使用—FZ后缀加以规定。

规定Z型记录的另外一种方法是使用后缀—P,它表示应安排该文件打印输出。当把—P添到设备名上时,编译程序将给文件具有零字节记录,并在每个打印行的开始保留一个特殊的字符,作为走纸控制用。对于这些文件应该使用带有ADVANCING选择的WRITE语句,以便控制走纸功能。

第七节 USE 过程

除系统在标号处理、错误校验和键码操作这些方面提供处理过程外,USE过程使用户得以规定其他一些处理。每当碰到USE语句规定的条件时,用户提供的过程被调用。

必须要在过程部分的DECLARATIVES(说明)节里规定USE过程。每个USE过程都包含一个节头,然后是一条USE语句。再下面是段,其中包含将执行的过程。

在USE过程中可以包括输入/输出语句,但一定不能引起由USE语句中提名的文件执行的任何输入/输出。在USE语句中提名的文件不能是分类文件。

USE过程一定不能引起执行自己或其他USE过程。

这里讨论USE语句的四种变形,其余的都在COBOL 4 参考手册中讨论:

USE AFTER ERROR PROCEDURE	在CYBER记录管理提供的输入/输出过程出错之后执行。
USE BEFORE AFTER LABEL PROCEDURES	用在CYBER记录管理提供的标准和标准标号写入和校验过程之前或其后。
USE FOR DUPLICATE KEY	在索引顺序文件上碰到重复基本键值的时候执行。
USE FOR HASHING	用于为直接文件计算随机键码。

7.1 USE AFTER ERROR

规定了USE AFTER ERROR PROEDURE,即指出了输入/输出错误之后执行一个

USE过程如果会引起执行INVALID KEY的话, 那么就不执行出错处理。赋给ERROR—CODE的数码是与CYBER记录管理程序诊断程序相应的数码, 这些数码列于记录管理程序参考手册中。

文件名一 1 [文件名一 2]

图 2—6 出错处理的USE语句

规定了USE BEFORE或AFTER LABEL PROCEDURE, 用户可以提供一些特殊处理, 以用在检验输入标号之前、后, 或用在准备输出标号之前后。关于这种特殊处理何时发生, 以及将要处理哪些文件都在USE语句中规定, 其格式如图2-7所示。

BEGINNING和ENDING规定要处理的标号的类型。如果没有规定BEGINNING,也没有规定ENDING,则对头标和尾标都执行USE过程。如果没有规定REEL (磁带卷),也没有规定FILE (文件),则对磁带卷和文件标号都执行这些过程。这里UNIT和REEL是同义词。

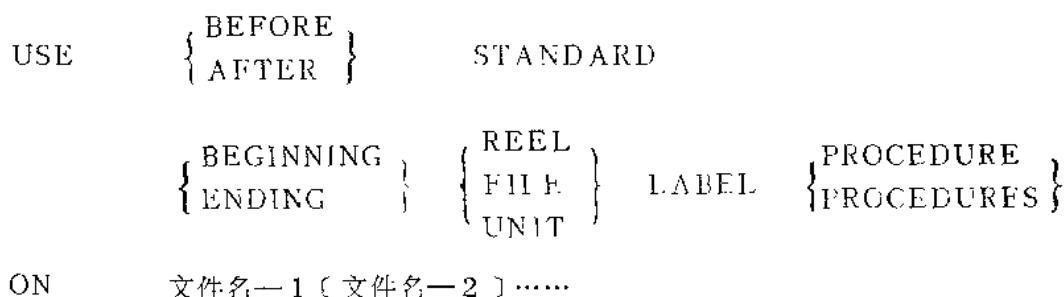


图 2-7 对于标号处理的USE语句

如果对一个索引顺序文件规定了USE FOR DUPLICATE KEY过程, 并且在文件建立或更新过程中碰到一个重复基本键值的话, 含有重复键的记录则不能写入文件, 而代之以执行USE过程, 以便由用户决定就重复键记录将采取什么动作。

• 12 •

```

USE FOR DUPLICATE KEY ON
    { ALL
      文件名—1 [文件名—2]
    }

```

图 2—8 重复键的USE语句

7.4 USE FOR HASHING

直接文件的记录是随机存放在有编号的同族数据块中的。每个记录的键都映射一个号码,而这个记录本身则存入对应的块里。这种映射处理称之为散列法,执行这种映射处理的子程序称为散列子程序。使用USE FOR HASHING过程,用户就能提供自己的散列子程序。COBOL并不会提供散列例程, 如果不规定USE FOR HASHING过程, 则使用 CYBER 记录管理程序的隐含散列例程。程序实例 5 (见第六章) 包含了一个 USE FOR HASHING过程的例子。

USE语句中或者列出将要使用这个过程的文件, 或者指出它作用于所有的直接文件 (见图 2—9)。

```

USE FOR HSHING ON
    { ALL
      文件名—1 [文件名—2] ...
    }

```

图 2—9 用于散列法的USE语句

散列过程从键码计算出一个号码, 系统就用这个号码找到正确的同族块。计算出这个值以后, 它被放入专门寄存器HASHING-VALUE (散列值), 系统规定它为COMPUTATIONAL—1项。这个值必须在零和文件中的块数减 1 之间。数据块数是在设备部分的NUMBER OF BLOCKS子句里给出的。如果这个值不在此界限内, 则执行INVALID KEY选项, 并给专门寄存器ERROR-CODE置位。

当把一个直接文件作为INPUT或I-O打开时(即这个文件已经存在, 并不是正在建立), CYBER记录管理程序要核实所供给的散列例程同写这文件时用过的散列例程是否一样。从这个文件中读出一个随机记录, 就调用所供给的散列例程去计算其键码的散列值。拿这个值与这键码的实际散列值进行比较, 如果它们不一致就发出一个诊断信息。所以, 散列例程必须可以独立于程序而能单独执行, 而且它必须只使用记录本身的数据。当打开这文件时, 不能从程序的任何其他工作区拾取数据, 除非那个区的内容保证不受到破坏。

第三章 顺序文件组织

在顺序组织的文件中, 定长记录或可变长度记录是按它们被写入时的物理顺序驻留在磁

带上或大容量存贮设备上的。每个WRITE语句都能使所指定的记录存放到紧挨最后写入记录的后面，文件关闭之后，在文件的最后记录后面要存放一个结束信息（EOI）。因为在顺序文件中总是按连续次序来读、写一个记录，因此不用关键字。

顺序文件组织最适用于从头读起的文件和只在文件末尾加进记录的文件。对加进记录并不予留空间，而是随着加新记录随时移动结束信息（EOI）。记录跳跃是不可能的，也不能重写一个记录。磁带文件、穿孔卡片文件、打印文件和某些大容量存贮文件都属于顺序文件，不驻留在大容量存贮设备上的所有文件必须具有顺序组织。

第一节 记录类型

用COBOL规定的六种记录类型都可适用于顺序文件：

- D 十进制计数
- F 定长
- R 记录标志
- T 尾部记数
- W 控制字
- Z 零字节

各种记录类型的说明见第二章。

第二节 设备类型

所有文件必驻留在下述两种设备之一：

磁带或大容量存贮设备（穿孔卡片文件和行打文件由COBOL程序处理时是驻留在大容量存贮设备上的）。另外，磁带文件被分为两类，其不同之处主要在规定物理记录大小的方式上：

X和I磁带（只适于NOS 1）和SI 磁带（适于NOS/BE 1 和NOS 1），它们和大容量存贮文件都具有这样的格式，其物理记录大小是由操作系统预先规定的。I 磁带是NOS 1 缺省情况，SI磁是NOS/BE 1 缺省情况。对于这类文件，其物理记录的大小就是设备的物理记录单位（PRU）的大小，如表 3—1 所示。

S (stranger) 和L (Long record stranger) 磁带有这样的格式：其物理记录大小是由用户规定的，S磁带上的物理记录必须小于或等于5120个字符。

这两类格式之间的差别对用户显得重要之处仅在于它们对记录块类型和长度的影响上（下面将会讨论）。

表 3—1 设备的 PRU 大小

设 备	PRU 大 小
大 容 量 存 贮 器	640个 字 符
编 码 的 SI 磁 带	1280个 字 符
二 进 制 SI 磁 带	5120个 字 符
x或I磁带（只适于NOS 1）	5120个 字 符

第三节 文件界限符

CYBER记录管理在顺序文件中能识别三种边界条件。文件的实际物理内容取决于设备类型和其他考虑（参见记录管理程序参考手册），但CYBER记录管理对各类边界的处理都是一致的。按使用范围讲，从大到小列出这三种边界如下：

信息结束（EOI），每个文件都只有一个信息结束标志，它紧跟在文件最后数据记录后面。磁带上的尾标在信息结束之后。在INPUT的文件上，信息结束相当于一张6/7/8/9卡。当文件是用反绕关闭时则在文件最末一个记录之后写这个信息结束标志。

区段结束（EOP），有时把这个边界视为文件结束。所有文件在输入时碰到EOP或EOI，都会引起执行READ语句中的AT END子句。当一个文件不是用反绕关闭时，则写区段结束。

节结束（EOS），这个边界对应于一个系统逻辑记录的结束。在INPUT的文件上，一个节结束相当于一张7/8/9卡，对于这个文件，输入时碰到一个EOS，将引起执行READ语句中的AT END子句。在其他文件上，COBOL对节结束是忽略不计的。

第四节 记录块类型

在顺序文件中把记录组合为较大单元，称这种单元为记录块，其目的是为了提高内存和大容量存贮设备之间的传输效率。CYBER记录管理和COBOL可支撑四种记录块，编译程是根据BLOCK CONTAINS子句或从ASSIGN于句的设备名上加的后缀得出数据块属何种类型。如果没有利用上面两种方法，则根据设备类型采用隐含的块类型。根据设备类型和BLOCK CONTAINS子句之规定，COBOL能选择的块类型总结在表3—2上。一般说来，C和I型块要比K和E型块效率更高，因为它们设计时专门利用适宜COBOL 4运行的硬件结构的优点。

表 3—2 记 录 块 类 型

BLOCK CONTAINS 子句格式	SI, I 和 X 磁带 大容量存贮器	S 和 L 带
省略该子句	C	K 每块1个记录
BLOCK CONTAINS 整数 CHARACTERS	C	E
BLOCK CONTAINS 整数 RECORDS	K	K
BLOCK CONTAINS 整数 1 TO 整数 2 RECORDS	C	E
BLOCK CONTAINS 整数 1 TO 整数 2 CHARACTERS	C	E