LB21-0772-0

Field Developed Program

Query Facility For System/3

Program Number: 5798-ANB
Feature Number: 8248

Systems Guide

This "User's Manual" is for the Systems Analyst, Programmer and Operator during the Systems Test and daily operations of the programs. It also contains programmer notes to assist in making minor alterations.

IBM

PROGRAMMING SERVICES

During a specified number of months immediately following initial availability of each licensed program designated as the Programming Services Period, and referenced in the Supplement to Amended License Agreement for IBM Program Products, the Customer may submit documentation to a designated IBM location when he encounters a problem which his diagnosis indicates is caused by a licensed program error. During this period only, IBM through the program sponsor(s) will, without additional charge, respond to an error in the current unaltered release of the licensed program by issuing known error correction information to the Customer reporting the problem and/or issuing corrected code or notice of availability of corrected code. However, IBM does not guarantee service results or represent or warrant that all errors will be corrected. Any on-site programming services or assistance will be provided at a charge.

Program error correction requests and/or comments concerning the contents of this publication should be addressed to:

IBM Corporation
8712 West Dodge Road
Omaha, Nebraska 68114
Attn: Mr. Don Larimer

WARRANTY

EACH LICENSED PROGRAM IS DISTRIBUTED ON AN 'AS IS' BASIS WITHOUT WARRANTY OF ANY KIND EITHER EXPRESS OR IMPLIED.

Requests for copies of IBM publications should be made to your IBM Representative or to the IBM Branch Office serving your locality.

TABLE OF CONTENTS

INTRODUCTION

The purpose of this manual is to provide supplemental information on the design, logic, and coding of the System/3 Query Facility. The Program Description and Operations Manual should remain the prime information source for the Facility, but this Systems Guide will be of value in making extensive modifications and diagnosing unusual error conditions. Also included is a suggested approach for generating the facility to execute on a System/3 Model 6.

SYSTEM OVERVIEW

The System/3 Query Facility is a tool that allows the user, programmer, and non-programmer alike to selectively access disk data files for preparing reports or gathering statistics. By using the functions provided in four simple commands and the functions of the System/3 Disk Sort, the user can rapidly formulate queries to the system and receive responses in the form of printed output without expending time and effort for writing and testing programs.

System Description

The System/3 Query Facility is comprised of two programs written in RPG II. Input to the first program is the 'SELECT' and 'SORT' commands. This program translates these statements into control statements that are compatible with the System/3 disk sort.

Upon processing these control statements, the sort utility then produces an output file of the records the user has selected, placing them in the defined sequence. This file then becomes input to the second program of the facility which processes the data and produces a report according to the definitions established by the 'PRINT' and 'COUNT' commands. Figure 4 presents samples of the control statements; Figure 5 illustrates how a query might be written and the resulting report.

The commands are designed to give a great deal of function to the user and demand little concern for rigid formatting. They are written in a manner that conveys meaning to the writer of a query, making a request simple and direct, and provides results in a minimum of time.

Programming Systems

The programs were written and tested under RPG II (5702-RG1, Version 5, Modification Level 0) for the System/3 Model 10 (disk). Instructions are provided for adaptation to a System/3 Model 6. In addition to the RPG II Compiler, the Disk Sort Program (5702-SM1) is required.

System Configuration

The object programs require 9216 bytes of memory. With a normal size core-resident system control program (approximately 3K bytes) a 12K central processor is required. The programs will execute on an otherwise minimum System/3 Model 10 disk configuration. Sufficient disk space should be available for sort work areas and output areas.

PROGRAMMING TECHNIQUES

The purpose of this section is to provide a general overview of the programming philosophy used in creating the Query Facility and to briefly describe the uses and functions of key fields, arrays, and subroutines. Because of the nature of the programs and the generally low requirements for the user to perform extensive modification, a "microscopic" perspective will not be provided.

For both programs, the command statements are read into arrays that are comprised of 1-byte character-formatted elements. These arrays can then be scanned, keying on the primary delimiters (blanks, commas, parentheses), to extract parameters from the control statements. This scanning technique is necessary because of the freedom of format allowed in statement construction. When the parameters (e.g. data element names and logical operators) are isolated in the input array they are stored for later reference in operand-oriented arrays that correspond to their function. For example, each data element name extracted from the 'SORT' command (program ANB01) is placed in the array 'FLD' that is later referenced to produce the sort field specifications. The output print lines in the program ANB02 are also arrays in which each print line is constructed byte-by-byte. A close inspection of the scanning, data extraction, and output line construction logic in both programs reveals that the technique is not unlike that which would be used in Assembler Language coding. RPG II coding was implemented to avoid excluding the Model 6 user and those Model 10 users that do not have assemblers.

PROGRAM ANB01

Arrays

TABFLD -- This table contains the portion of the Data Element Dictionary that is the data element name.

TABPAR -- This table (alternates with TABPAR) contains the remainder of the information for the Data Element Dictionary including: location of the field in the record, format, number of decimals, print field size, and heading information.

IN -- Array that is the command statement input area.

SRT -- The array that the entire 'SORT' command is moved to (from 'IN') prior to scanning.

SEL -- This array contains the one or two cards comprising the 'SELECT' statement. Each card is moved from the input array 'IN' when it is read.

OPA -- This array contains the 'A' or first operand of each comparative Boolean relationship, always a data element name.

Arrays

OPB -- This array contains the 'B' or second operand of each comparative Boolean relationship, either a data element name or a literal field.

BO -- This array contains the comparative operator for each Boolean relationship (GT, LT, EQ, etc.).

LN -- This array contains the first 2 characters of the linkage operator ('AND' or 'OR') that connects each pair of comparative relationships.

LBF -- This is the array in which literals are constructed prior to output, termed the "literal buffer".

FLD -- This array contains the data element names extracted from the 'SORT' statement. The last byte contains the sequence by which that field is to be sorted (N-normal, D-descending).

Subroutines

SCANSR -- Scans the array containing the 'SORT' control statement ('SRT') extracting the data element names and the sequence codes and places them in the array 'FLD'.

SCANSL -- Scans the array containing the 'SELECT' control statement ('SEL') extracting the logical operators and operands (within the parentheses) and the linkage operators (connecting the comparative relationships) and placing them in the arrays 'OPA', 'OPB', 'BO', and 'LN'.

PARAMS -- Extracts the definitive parameters for a data element name from the Data Element Dictionary entry.

FLDFIL -- Called by the routines 'SCANSR' and 'SCANSL' to move the operand or operator they have encountered into a 20-byte field 'BYT20' left adjusted.

RCDCRD -- This routine takes the operands and operators from the 'SELECT' statement (found in arrays 'OPA', 'OPB', 'BO', 'LN') and punches the sort specifications that specify record selection.

SRTCRD -- Takes the fields that are to be sorted from the array 'FLD' and and produces disk sort specification cards.

LITRAL -- This routine analyzes literal fields found in the array 'OPB' and performs the following functions: right- or left-justification, decimal alignment, and padding for matching field lengths, sign extraction.

PROGRAM ANB02

Arrays

AIN -- Array in which the disk record for the queried file is stored for data extraction.

ALN -- Print-line array for detail lines.

HLN -- Print-line array for heading line.

ACN -- Input array for control statements.

TABFLD/
TABPAR -- See "Arrays", program ANB01.

FLD -- Array containing operands (data element names and numeric constants) of the 'PRINT' statement.

CNT -- Array containing the data element names of the 'COUNT' statement.

BF -- Buffer in which numeric fields are assembled, edited, zero suppressed, before placement in the print line.

TOT -- Array in which the totals for the 'COUNT' fields are accumulated.

Subroutines

ANALYZ -- Scans the 'PRINT' and 'COUNT' statements in the array 'ACN' and isolates the operands encountered, placing them in the arravs 'FLD' or 'CNT'.

PARAMS -- Extracts the definitive parameters for each field to be printed from the Data Element Dictionary. If the operand is numeric (for space control) it increments the index for the print line array and returns to the calling routine.

BLDLNX -- Creates the detail print line ('ALN') and the heading print line ('HLN').

MVNUMS -- Extracts a zoned-decimal field from the input disk record and places it in a numeric field 'NM' for later editing.

NUMRIC -- Edits the numeric field 'NM' and places it in the output line.

Subroutines

UNPACK -- Isolates packed-decimal fields in the input disk record and places them in the numeric field 'NM'.

TOTOUT -- Outputs the total fields for the 'COUNT' statement after it causes them to be edited.

MODIFICATION AIDS

The only modifications discussed here are those that would allow the system to function on a 12K System/3 Model 6. Any other modifications are discussed in the Program Description and Operations Manual. The system has not been formally tested on a System/3 Model 6 and the following information is suggested only as a guide to implementation in that environment.

To avoid the errors inherent in keying the source programs using Keyboard Source Entry it is more practical to load the source programs in card form using a Model 6 with a data recorder or a Model 10 disk system. When using a Model 10 to place the source decks in the source library use the following OCL for each program:

```
// READER MFCU1
// LOAD $MAINT, F1
// RUN
// COPY FROM-READER, TO-R1, LIBRARY-S, RETAIN-P, NAME-ANBXX



// CEND
// END
```

The modifications to allow execution on a Model 6 are mainly oriented to the File Description section of the programs. For program ANB01, the File Description specifications for the files 'SPECSIN' and 'SPECSOUT' must be changed. 'SPECSIN' must be altered in columns 40-46 from 'MFCU1' to 'CONSOLE'. 'SPECSOUT' must be modified in columns 40-46 from 'MFCU2' to 'TRACTR1' and in columns 22-23 from '96' to '01'. This will cause the program to look for input via the console-keyboard and to place the sort specifications (normally punched into cards on the MFCU) on the printer file. This requires that the specifications be keyed into the Disk Sort Program in Phase II of execution. In addition, the output specifications with '*PRINT' in columns 32-37 must be deleted. Finally, place the printer spacing controls in each 'E' and 'T' type output line, a '1' in column 18 of these lines will suffice.

Only the File Description specifications for program ANB02 should require change. For the file 'CONTROL' change columns 40-46 from 'MFCU1' to 'CONSOLE'; for the file 'LIST' change columns 40-46 to 'TRACTR1' and columns 22-23 to 01. The OCL required for execution of the facility is very compatible between Model 6 and Model 10 and therefore no discussion of OCL will be presented.

Page Numbering and Date Identification

A potentially valuable modification to make to the Facility is the addition of page numbering and system date to print on each page of the report. This would serve as reference information to the user of the report information. The following cards should be keyed and inserted in the source deck of program ANB02 using the statement number identifier (columns 1-5) to properly place these cards in the deck:

```
035540                          PAGE  Z    96
035550                                     93 'PAGE'
035560                          UDATE Y    88
035570         D  12     99N75N69
035580        OR         OF
```

User Report Headings

This modification to ANB02 would allow the user to include a heading card along with the 'PRINT' and 'COUNT' statements when executing the print program (Phase III). By creating a card with the characters '**' in columns 1 and 2, the remaining columns may contain any identifying information that the user wishes to appear at the top of his printed output. The following cards are required in the source deck of ANB02:

```
00208ICONTROL  NS  35   1 C*   2 C*
00209I                                        3   96 HEADG

00295C   35                  GOTO ENDCAL

035530                            HEADG     96
```

If the modification for date and page numbering has not been made, include statements 3547-8 as described in the previous section.

Tag Sorting

To decrease the time required to sort a larger master file and reduce the size of the sort work and output files, the "TAG" sorting technique could be implemented to produce an ADDROUT file for input to ANB02.

In program ANB01, statement number 4000 should have the literal changed to 'HSORTA', statement number 4030 should be changed to output a literal of '0' in position 27, and statements 4350-4380 should be removed.

Program ANB02 requires the addition of a file description specification for the Record Address file, modification of the file description specification for the file 'INPUT' to be processed via the RA file, and creation of an extension specification should be added to associate the RA file with the file 'INPUT'.

These modifications may require more than a 12K CPU to function. Additional information relating to TAG sorts and processing Record Address files can be found in the System/3 Disk Sort and RPG II reference manuals.

LB21-0772-0

Query Facility For System/3
Systems Guide

Reproduced in France Printed in U.S.A. LB21-0772-0

IBM

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)