

DFSMS DFSORT: The ICETOOL Cometh - Getting Started Using ICETOOL

Samuel Ryan Smith
IBM

March 2nd 2015
Session Number : 17106

ssmith@us.ibm.com



SHARE is an independent volunteer-run information technology association that provides **education, professional networking and industry influence.**



Agenda

- What is ICETOOL?
- Generating Sample data
- Basic ICETOOL JCL
- ICETOOL Utility Operators
- ICETOOL Operator Syntax
- Cool Things you can do with ICETOOL
- Defining and Using Symbols
- Q & A

What is ICETOOL?

- ICETOOL is a batch front-end utility
- ICETOOL uses the capabilities of DFSORT
- ICETOOL can be called directly or from a program
- ICETOOL includes 17 operators
- Messages and return codes

Generating Sample data

- Before we get started, this presentation uses a chain of bookstores for examples
 - Colorado and California
- Corresponding info in Getting Started
- **ICEDATA JOB**
 - Creates all data used in examples
 - Located in **SYS1.SICESAMP**

Basic ICETOOL JCL

- ICETOOL required JCL statements
 - TOOLMSG DD
 - DFSMSG or SSMSG DD
 - TOOLIN DD
 - XXXXCNTL DD
 - ICETOOL statements indicate operations to be performed

Basic ICETOOL JCL (continued)

```
//EXAMP      JOB  A492,PROGRAMMER
```

```
//STEP0100 EXEC PGM=ICETOOL
```

```
//TOOLMSG    DD  SYSOUT=*
```

```
//DFSMSG     DD  SYSOUT=*
```

```
//TOOLIN     DD  *
```

```
<ICETOOL statements go here>
```

...

```
<Additional JCL statements go here, DD's, etc...>
```

...

```
//XXXXCNTL DD *
```

```
<Sort control statements go here>
```

...

ICETOOL Utility operators

- COPY
- COUNT
- DATASORT
- DEFAULTS
- DISPLAY
- MERGE
- MODE
- OCCUR
- RANGE

ICETOOL Utility operators (continued)

- RESIZE
- SELECT
- SORT
- SPLICE
- STATS
- SUBSET
- UNIQUE
- VERIFY

ICETOOL Operator Statement Syntax

- **operator operand ... operand**
- **Example:** COPY FROM(IN) TO(OUT1,OUT2)
 - Operator is one of the seventeen ICETOOL operator names.
 - Operand is keyword or keyword(parameter,...)
 - One or more blanks can be used before the operator and between operands.
 - Columns 1-72 are scanned; columns 73-80 are ignored.
 - Continuation can be indicated by a dash (-) after the operator or any operand. Each operand must be completely specified on one line.
- **Example:** SORT FROM(IN1) –
TO(OUT1,OUT2,OUT3) –
USING(ABCD)

Cool Things you can do with ICETOOL

- **Creating multiple identical copies**
- **Collecting statistics using ICETOOL**
- **The STATS Operator/statistical output**
- **Counting values in a range**
- **Creating Tailored Reports**
- **Edit Masks, Leading Zeros, Edit Patterns and Division**
- **Leading, Floating and Trailing Characters**
- **Printing Sectioned Reports**
- **How Many Times Fields Occur**
- **Records by Field Occurrences**
- **Create small records from large records and vice versa**

Creating multiple identical copies using ICETOOL

```

//STEP0100 EXEC PGM=ICETOOL
//TOOLMSG DD SYSOUT=*
//DFSMSG DD SYSOUT=*
//TOOLIN DD *
* BOOKS FROM VALD AND WETH
  SORT FROM(BKS) - ← BKS is DD of input
    TO (DAPUBS, PRPUBS) - ← DAPUBS/PRPUBS is DD of first/second output
    USING (SPUB) ← First four characters of control data set DD
/*
//BKS DD DSN=YOURHLQ.SORT.SAMPIN,DISP=SHR
// DD DSN=YOURHLQ.SORT.SAMPADD,DISP=SHR
//DAPUBS DD DSN=L2.SAMPLE.SORT.DAPUBS,
// DISP=(NEW,CATLG),SPACE=(CYL,(5,5),RLSE),UNIT=SYSDA
//PRPUBS DD DSN=L2.SAMPLE.SORT.PRPUBS,
// DISP=(NEW,CATLG),SPACE=(CYL,(5,5),RLSE),UNIT=SYSDA
//SPUBCNTL DD *
  INCLUDE COND=(106,4,EQ,C'VALD',OR,106,4,EQ,C'WETH'),FORMAT=CH
  SORT FIELDS=(106,4,A,1,75,A),FORMAT=CH
/*

```

Input data : YOURHLQ . SORT . SAMP IN

```

-----+-----1-----+-----2-----+-----3-----+ . . . +-----8-----+-----9-----+-----0-----+-----1-----+-----2
*****
GUNTHER'S GERMAN DICTIONARY          WILLIS          GUNTHER          WETH
COMPLETE SPANISH DICTIONARY          ROBERTS         ANGEL            VALD
ANOTHER ITALIAN DICTIONARY           UNDER          JOAN             COR
FRENCH TO ENGLISH DICTIONARY        JONES          JACK             FERN
GUIDE TO COLLEGE LIFE                LAMB           CHARLENE         WETH
THE ANIMAL KINGDOM                   YOUNG          KEVIN            COR BIOL 80522B
A SMALLER WORLD: MICROBES            BEESLY         GEORGE           FERNBIOL 80522B
DNA: BLUEPRINT FOR YOU               HAVERS         ILSA             FERNBIOL 80523I
CELLS AND HOW THEY WORK              JETTS          PETER            VALDBIOL 80523I
KNOW YOUR CONSUMER                   ZANE           JENNIFER         COR BUSIN70251M
ANTICIPATING THE MARKET              ALLEN          CLYDE            WETHBUSIN70124A
ZEN BUSINESS                          WILLIAMS       KATIE            VALDBUSIN70255B
THE ART OF TAKEOVERS                 HUNT           ROBERT           FERNBUSIN70255B
THE TOY STORE TEST                   LITTLE        MARIE            COR COMP 00205V
NOVEL IDEAS                           PETERS        SETH             VALDENGL 10054F
POLITICS AND HISTORY                 TOMPSOM        KEN              FERNHIST 50521W
CIVILIZATION SINCE ROME FELL         PIERCE        NICOLE           WETHHIST 50420W
REBIRTH FROM ITALY                   FISH          JOHN             WETHHIST 50632E
FREUD'S THEORIES                      GOOLE         APRIL            VALDPSYCH30975P
MAP OF THE HUMAN BRAIN               WINTER        POLLY            COR PSYCH30016P
*****

```

Output Data in datasets PRPUBS and DAPUBS

```

-----+-----1-----+-----2-----+-----3-----+
*****
CELLS AND HOW THEY WORK                                VALD
COMPLETE SPANISH DICTIONARY                            VALD
EDITING SOFTWARE MANUALS                              VALD
FREUD'S THEORIES                                       VALD
INTRODUCTION TO BIOLOGY                               VALD
NOVEL IDEAS                                            VALD
SHORT STORIES AND TALL TALES                          VALD
STRATEGIC MARKETING                                   VALD
VIDEO GAME DESIGN                                     VALD
ZEN BUSINESS                                           VALD
ANTICIPATING THE MARKET                               WETH
CIVILIZATION SINCE ROME FELL                           WETH
COMPUTERS: AN INTRODUCTION                             WETH
EIGHTEENTH CENTURY EUROPE                             WETH
GUIDE TO COLLEGE LIFE                                 WETH
GUNTHER'S GERMAN DICTIONARY                           WETH
REBIRTH FROM ITALY                                    WETH
SYSTEM PROGRAMMING                                    WETH
THE INDUSTRIAL REVOLUTION                             WETH
*****
0-----+-----1
*****

```

**** Only Portion of the data is shown to emphasize how the data is sorted**

Collecting Statistics using ICETOOL

```
//TOOLIN DD *
* BOOKS FROM VALD AND WETH
  SORT FROM(BKS) TO(DAPUBS,PRPUBS) USING(SPUB)
* STATISTICS FROM ALL BRANCHES
  STATS FROM(ALL) - DD name of the input dataset
    ON(18,4,ZD) - Employees
    ON(28,6,PD) - ← Profit
    ON(22,6,PD) ← Revenue
/*
//ALL DD DSN=YOURHLQ.SORT.BRANCH,DISP=SHR
//BKS DD DSN=YOURHLQ.SORT.SAMPIN,DISP=SHR
// DD DSN=YOURHLQ.SORT.SAMPADD,DISP=SHR
//DAPUBS DD DSN=L2.SAMPLE.SORT.DAPUBS2,
// DISP=(NEW,CATLG),SPACE=(CYL,(5,5),RLSE),UNIT=SYSDA
//PRPUBS DD DSN=L2.SAMPLE.SORT.PRPUBS2,
// DISP=(NEW,CATLG),SPACE=(CYL,(5,5),RLSE),UNIT=SYSDA
//SPUBCNTL DD *
  INCLUDE COND=(106,4,EQ,C'VALD',OR,106,4,EQ,C'WETH'),FORMAT=CH
  SORT FIELDS=(106,4,A,1,75,A),FORMAT=CH
/*
```

- Added 6 lines in **RED**

Sample Input Data for STATS Operator

```

-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5
***** Top of Data ****
Los Angeles      CA003B....ë.....ý
San Francisco    CA003E...âb.....c.
Fort Collins     CO002B.....f.
Sacramento      CA002I...âÊ%.....%
Sunnyvale       CA001H.....pý
Denver           CO003C....g%....ð
Boulder          CO003B....f%.....
Morgan Hill     CA001E.....
Vail             CO001I.....@
San Jose         CA002A.....*....<
San Diego       CA002B....m.....*
Aspen            CO002{....Ø.....
***** Bottom of Data **

```

•Note that the data in columns 22 through 34 is not readable since these values are currently packed decimals.

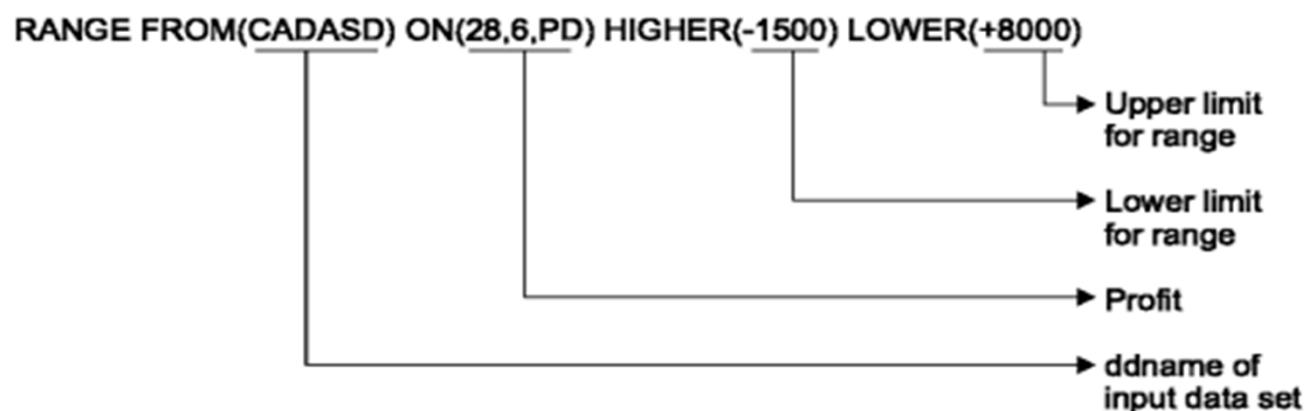
The Output from STATS Operator

```
* STATISTICS FROM ALL BRANCHES
  STATS FROM(ALL) ON(18,4,ZD) ON(28,6,PD) ON(22,6,PD)
DFSORT CALL 0001 FOR COPY FROM ALL      TO E35 EXIT COMPLETED
RECORD COUNT:  0000000000000012
STATISTICS FOR (18,4,ZD)      :
  MINIMUM:  +0000000000000015,  MAXIMUM:  +0000000000000035
  AVERAGE:  +0000000000000024,  TOTAL   :  +0000000000000298
STATISTICS FOR (28,6,PD)     :
  MINIMUM:  -0000000000004278,  MAXIMUM:  +0000000000008276
  AVERAGE:  +0000000000004222,  TOTAL   :  +00000000000050665
STATISTICS FOR (22,6,PD)     :
  MINIMUM:  +000000000012300,  MAXIMUM:  +0000000000042820
  AVERAGE:  +000000000027469,  TOTAL   :  +0000000000329637
OPERATION RETURN CODE:  00
```


Counting Values in a Range using ICETOOL

```
//TOOLIN DD *
* SEPARATE OUTPUT FOR CALIFORNIA
  SORT FROM(ALL) USING(CACO)
* STATISTICS FROM ALL BRANCHES
  STATS FROM(ALL) ON(18,4,ZD) ON(28,6,PD) ON(22,6,PD)
* CALIFORNIA BRANCHES PROFIT ANALYSIS
  RANGE FROM(CADASD) ON(28,6,PD) HIGHER(-1500) LOWER(+8000)
/*
//ALL DD DSN=YOURHLQ.SORT.BRANCH,DISP=SHR
//CADASD DD DSN=&&CA,DISP=(,PASS),SPACE=(CYL,(2,2),RLSE)
//CACOCNTL DD *
  SORT FIELDS=(1,15,CH,A)
  OUTFIL FNAMES=CADASD,INCLUDE=(16,2,CH,EQ,C'CA')
/*
```

Output for Counting Values in a Range



* CALIFORNIA BRANCHES PROFIT ANALYSIS

RANGE FROM(CADASD) ON(28,6,PD) HIGHER(-1500) LOWER(+8000)

ICE627I 0 DFSORT CALL 0002 FOR COPY FROM CADASD TO E35 EXITCOMPLETED

ICE628I 0 RECORD COUNT: 0000000000000007

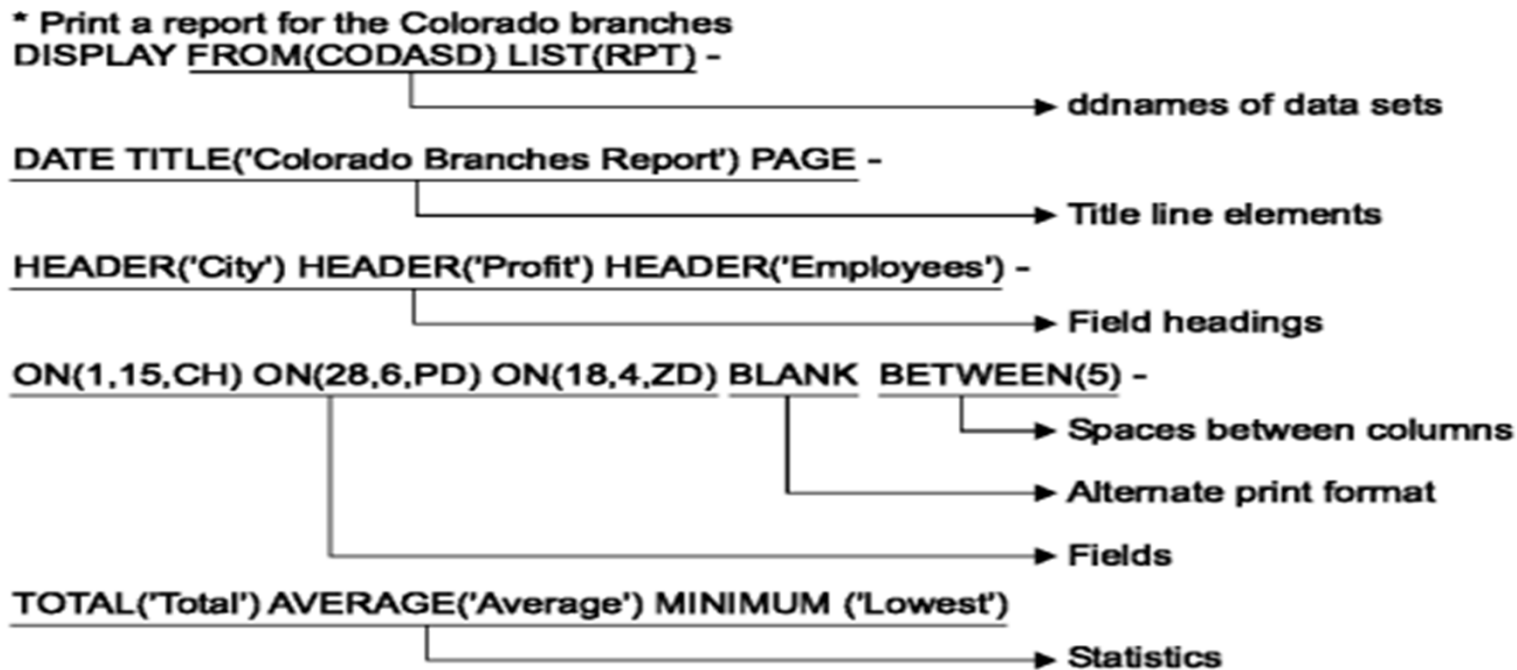
ICE631I 0 NUMBER OF VALUES IN RANGE FOR (28,6,PD) : 0000000000000003

ICE602I 0 OPERATION RETURN CODE: 00

Creating Tailored Reports

```
//TOOLIN DD *
* SEPARATE OUTPUT FOR CALIFORNIA AND COLORADO BRANCHES
  SORT FROM(ALL) USING(CACO)
* PRINT A REPORT FOR THE COLORADO BRANCHES
  DISPLAY FROM(CODASD) LIST(RPT) -
    DATE TITLE('COLORADO BRANCHES REPORT') PAGE -
    HEADER('CITY') HEADER('PROFIT') HEADER('EMPLOYEES') -
    ON(1,15,CH) ON(28,6,PD) ON(18,4,ZD) BLANK BETWEEN(5) -
    TOTAL('TOTAL') AVERAGE('AVERAGE') MINIMUM('LOWEST')
/*
//ALL DD DSN=YOURHLQ.SORT.BRANCH,DISP=SHR
/*
//CACOCNTL DD *
  SORT FIELDS=(1,15,CH,A)
  OUTFIL FNames=CODASD,INCLUDE=(16,2,CH,EQ,C'CO')
/*
//CODASD DD DSN=&&CO,DISP=(,PASS),SPACE=(CYL,(2,2)),UNIT=3390
//RPT DD SYSOUT=*
```

Creating Tailored Reports



•This is what we saw in the RED control statements from the previous slide

Tailored Report Output

01/14/13			COLORADO BRANCHES REPORT			- 1 -		
CITY			PROFIT			EMPLOYEES		
-----			-----			-----		
Aspen			5200			20		
Boulder			7351			32		
Denver			6288			33		
Fort Collins			-2863			22		
Vail			5027			19		
TOTAL			21003			126		
AVERAGE			4200			25		
LOWEST			-2863			19		

Edit Masks

- Thirty-nine pre-defined Edit Masks
 - **d** - decimal digit (0-9)
 - **w** - leading sign blank for + or – for negative
 - **x** - trailing sign blank for + or – for negative
 - **y** - leading sign blank for + or (for negative
 - **Z** - trailing sign blank for + or) for negative

 - Edit Pattern Mask E1 would look like
 - yd,ddd,ddd,ddd,ddd,ddd,ddd,ddd,ddd,dddz
 - 12345678
 - 12,345,678 ←results

Edit Masks Example

Add this Edit Pattern Mask
Changing ON(28,6,PD) to ON(28,6,PD,**E1**)

01/14/13 COLORADO BRANCHES REPORT - 1 -		
CITY	PROFIT	EMPLOYEES
-----	-----	-----
Aspen	5,200	20
Boulder	7,351	32
Denver	6,288	33
Fort Collins	(2,863)	22
Vail	5,027	19
TOTAL	21,003	126
AVERAGE	4,200	25
LOWEST	(2,863)	19

Leading Zeros

By default, leading zeros are not displayed when you use an edit mask, but you can change that by adding LZ

HEADER('No leading zeros', '(without LZ)') ON(28,6,PD,E1)

HEADER('Leading zeros', '(with LZ)') ON(28,6,PD,E1,LZ)

No leading zeros (without LZ)	Leading zeros (with LZ)
-----	-----
(4,278)	(00,000,004,278)
6,832	00,000,006,832
(2,863)	(00,000,002,863)
8,276	00,000,008,276
(978)	(00,000,000,978)
6,288	00,000,006,288
7,351	00,000,007,351
3,271	00,000,003,271
5,027	00,000,005,027
8,264	00,000,008,264
8,275	00,000,008,275
5,200	00,000,005,200

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

Edit Patterns

- The pattern (1 to 44 characters) must be enclosed in single apostrophes. Each **9** in the pattern (up to 31) is replaced by a corresponding digit from the numeric value. Characters other than 9 in the pattern appear as specified. To include a single apostrophe (') in the pattern, specify two single apostrophes (").
 - For Example:
 - 8-byte ZD date in the form *mmddyymm* cols. 41-48
 - *mm/dd/yyyy* using ON(41,8,ZD,E'99/99/9999')
 - 01122013 is displayed as **01/12/2013**
 - Or:
 - 10-byte ZD phone number in the form *aaapppnnnn* cols. 21-30
 - (aaa)-ppp-nnnn using ON(21,10,ZD, E'(999)-999-9999')
 - 0123456789 is displayed as **(012)-345-6789**

Division

- Ten division items
 - /**D** - divide by 10
 - /**C** - divide by 100
 - /**K** - divide by 1000
 - /**DK** - divide by 10000 (10*1000)
 - /**CK** - divide by 100000 (100*1000)
 - /**M** - divide by 1000000 (1000*1000)
 - /**G** - divide by 1000000000 (1000*1000*1000)
 - /**KB** - divide by 1024
 - /**MB** - divide by 1048576 (1024*1024)
 - /**GB** - divide by 1073741824 (1024*1024*1024)

Profit/(Loss) in M\$	

	(4)
	6
	(2)
	8
	0
	6
	7
	3
	5
	8
	8
	5

- Using HEADER('Profit/(Loss) in M\$') and ON(28,6,PD,E1,/M)

Leading, Floating and Trailing Characters

- Add floating, leading, and trailing characters to your numeric and character fields as follows:
 - **F'string'** - a floating string
 - Left of the first non-blank character
 - **L'string'** - a leading string
 - Beginning of the character or numeric data
 - **T'string'** - a trailing string
 - End of the character or numeric data

- Using `HEADER('Profit')` and `ON(28,6,PD,A1,F'$',T'**)'`

Profit	

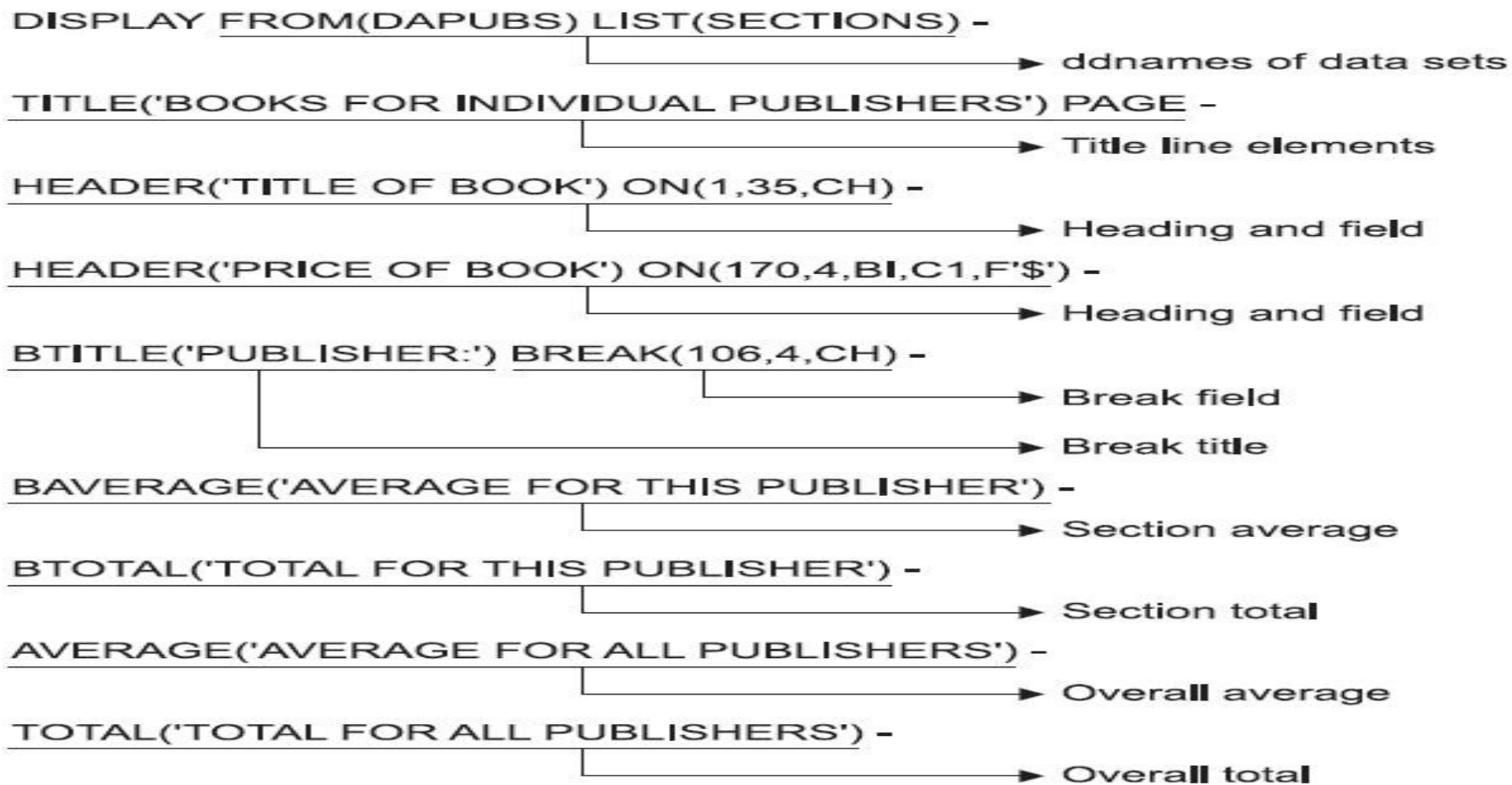
	\$-4,278**
	\$6,832**
	\$-2,863**
	\$8,276**
	\$-978**
	\$6,288**
	\$7,351**
	\$3,271**
	\$5,027**
	\$8,264**
	\$8,275**
	\$5,200**

Printing Sectioned Reports

- BREAK operand of DISPLAY
 - Create reports divided into sections
 - BREAK on a character or numeric
 - Previously sorted data
- **BREAK(p,m,f,formatting)**
 - Formatting is the same as with the ON operator
- **BTITLE**
 - String to be used for the break title
- **BTOTAL and BAVERAGE**
 - Section statistics at the end of each section

Printing Sectioned Reports

* Print a report of books for individual publishers



Complete your session evaluations online at www.SHARE.org/Seattle-Eval

Sectioned Report Output

BOOKS FOR INDIVIDUAL PUBLISHERS - 1 -	
PUBLISHER: VALD	
TITLE OF BOOK	PRICE OF BOOK
-----	-----
CELLS AND HOW THEY WORK	\$24.95
.
ZEN BUSINESS	\$12.00
AVERAGE FOR THIS PUBLISHER	\$17.91
TOTAL FOR THIS PUBLISHER	\$179.14

BOOKS FOR INDIVIDUAL PUBLISHERS - 2 -	
PUBLISHER: WETH	
TITLE OF BOOK	PRICE OF BOOK
-----	-----
ANTICIPATING THE MARKET	\$20.00
.
THE INDUSTRIAL REVOLUTION	\$7.95
AVERAGE FOR THIS PUBLISHER	\$18.53
TOTAL FOR THIS PUBLISHER	\$166.77

Sectioned Report Output (continued)

BOOKS FOR INDIVIDUAL PUBLISHERS	
TITLE OF BOOK	PRICE OF BOOK

AVERAGE FOR ALL PUBLISHERS	\$18.20
TOTAL FOR ALL PUBLISHERS	\$345.91

How Many Times Fields Occur

- **OCCUR** operator how many times ON fields occurs
 - **ALLDUPS** - Duplicate values
 - **NODUPS** - Non-duplicate values
 - **EQUAL** - Specified number of times
 - **HIGHER** - More than a specified number of times
 - **LOWER** - Less than a specified number of times
 - **ON(VALCNT)** - Each field value occurs
 - **ON(VLEN)** - Length of VLR records

How Many Times Fields Occur

* Print the count of books in use from each publisher

OCCUR FROM(BKIN) LIST(PUBCT) BLANK -

→ Alternate print format

→ ddnames of data sets

TITLE('Books from Publishers') DATE(DMY.) -

→ Title line elements

HEADER('Publisher') HEADER('Books Used') -

→ Field headings

ON(106,4,CH) ON(VALCNT,N05)

→ Publisher and Count

How Many Times Fields Occur Output

```
Books from Publishers 11.01.13
```

```
Publisher Books Used
```

```
-----
```

```
COR 7
```

```
FERN 4
```

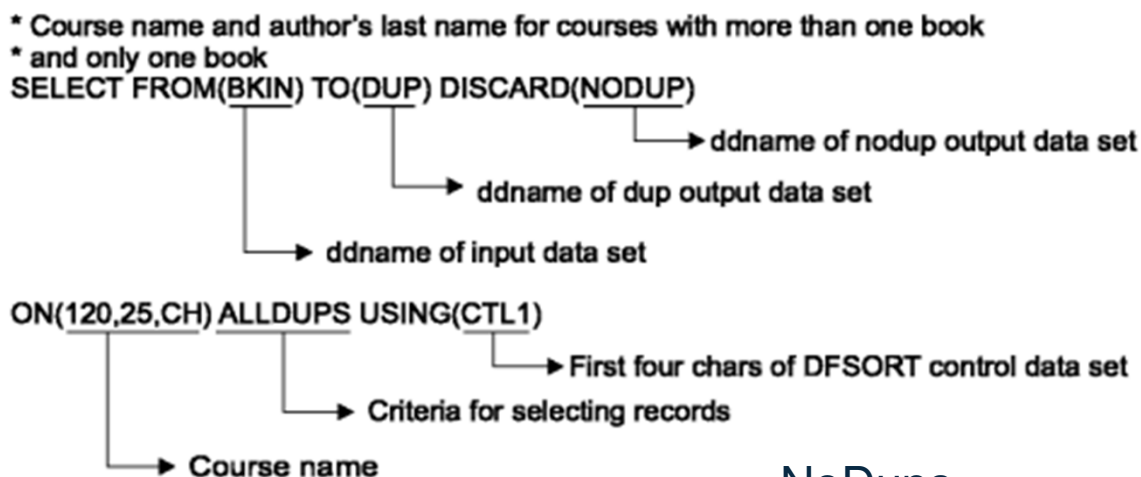
```
VALD 5
```

```
WETH 4
```

Records by Field Occurrences

- **SELECT** operand to create an output dataset with how many times different ON field values occur
 - **FIRST** - First record for each value
 - **LAST** - Last record for each value
 - **FIRSTDUP** - First record for duplicate values
 - **LASTDUP** - Last record for duplicate values
 - **ALLDUPS** - All records with duplicate values
 - **NODUPS** - Records with non-duplicate values
 - **EQUAL** - Specified number of times
 - **HIGHER** - More than a specified number of times
 - **LOWER** - Less than a specified number of times

Records by Field Occurrences



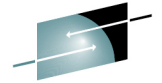
AllDups

INTRO TO COMPUTERS	CHATTERJEE
INTRO TO COMPUTERS	CHATTERJEE
INTRO TO COMPUTERS	CHATTERJEE
MODERN POETRY	FRIEDMAN
MODERN POETRY	FRIEDMAN
WORLD HISTORY	GOODGOLD
WORLD HISTORY	WILLERTON

NoDups

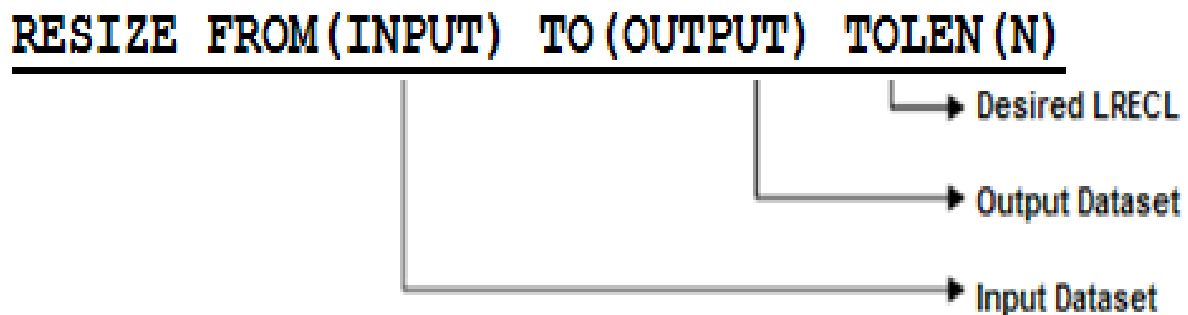
ADVANCED MARKETING	LORCH
BIOLOGY I	GREENBERG
DATA MANAGEMENT	SMITH
EUROPEAN HISTORY	BISCARDI
FICTION WRITING	BUCK
MARKETING	MAXWELL
PSYCHOANALYSIS	NAKATSU
PSYCHOLOGY I	ZABOSKI
TECHINCAL EDITING	MADRID

Complete your session evaluations online at www.SHARE.org/Seattle-Eval



Create small records from large records and vice versa

- **RESIZE** operator to create multiple smaller fixed-length records from a larger fixed-length record or to create one long fixed-length record from several shorter fixed-length records.



Creating Small records from Large record

Let's say we have an input file with RECFM=FB and LRECL=52 that has these records:

```
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5--  
*SECTION 001**SECTION 002**SECTION 003**SECTION 004*  
*SECTION 005**SECTION 006**SECTION 007**SECTION 008*
```

```
//TOOLIN DD *  
  RESIZE FROM(INPUT) TO(OUTPUT) TOLEN(13)  
//*
```

We want to split each 52-byte record into four 13-byte records. The output data set will have RECFM=FB and LRECL=13 and contain these records:

```
*SECTION 001*  
*SECTION 002*  
*SECTION 003*  
*SECTION 004*  
*SECTION 005*  
*SECTION 006*  
*SECTION 007*  
*SECTION 008*
```

Creating Large Records From Small Records

Lets say we have an input file with RECFM=FB and LRECL=13 that has these records:

```

-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5--
*SECTION 001*
*SECTION 002*
*SECTION 003*
*SECTION 004*
*SECTION 005*
*SECTION 006*
*SECTION 007*
*SECTION 008*

```

```
//TOOLIN DD *
```

```
  RESIZE FROM(INPUT) TO(OUTPUT) TOLEN(52)
```

```
//*
```

We want to combine 4 13-byte records into a single 52-byte record. The output data set will have RECFM=FB and LRECL=52 and contain these records:

```

-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5--
*SECTION 001**SECTION 002**SECTION 003**SECTION 004*
*SECTION 005**SECTION 006**SECTION 007**SECTION 008*

```

Defining and Using Symbols

- **SYMNAMES DD**
 - RECFM=FB and LRECL=80
- Used in any ICETOOL or DFSORT JOB

- **TITLE**
 - Name of the Symbol created
 - 1 – 50 characters
 - Letters, numbers, \$, @, _, and –
 - First character NOT a number
 - Provide length and format
 - SYMBOL, Symbol, and symbol
 - 3 different symbols

Defining and Using Symbols

- `Author_First_Name, *, 15, CH`
- `Publisher, *, 4, CH`
- `Instructor_Last_Name, *, 15, CH`
- `Instructor_Initials, *, 2, CH`
- `Price, *, 4, BI`
- ...

```
//IN          DD DSN=A123456.SORT.SAMPIN,DISP=SHR
//SYMNAMES   DD DSN=A123456.SORT.SYMBOLS,DISP=SHR
//OUT        DD DSN=A123456.SORT.SAMPOUT,DISP=OLD
//SYMNOUT    DD SYSOUT=*
//TOOLIN     DD *
    SORT FROM(IN) TO(OUT) USING(CTL1)
    RANGE FROM(OUT) ON(Price) LOWER(+700)
    RANGE FROM(OUT) ON(Price) HIGHER(+2000)
//CTL1CNTL DD *
    SORT FIELDS=(Instructor_Last_Name,A,Instructor_Initials,A,
    Price,D)
/*
```

Defining and Using Symbols

- Use Symbols to define constants
 - Decimal, character, hexadecimal constants or bit
- From an earlier slide
 - RANGE FROM(OUT) ON(Price) LOWER(+700)
 - RANGE FROM(OUT) ON(Price) HIGHER(+2000)
- If you define
 - Discount,+700
 - Premium,+2000
- You could use
 - RANGE FROM(OUT) ON(Price) LOWER(Discount)
 - RANGE FROM(OUT) ON(Price) HIGHER(Premium)

Questions?



Complete your session evaluations online at www.SHARE.org/Seattle-Eval

2/25/2015

REFERENCES.

- The DFSORT home page on the World Wide Web at URL: <http://www.ibm.com/storage/dfsor>
- z/OS DFSORT Application Programming Guide (SC23-6878-00)
- DFSORT: Getting Started (SC26-7527)
- z/OS DFSMSrmm Reporting (SC26-7406)
- z/OS DFSMSHsm Data Recovery Scenarios (GC35-0419)
- z/OS SecureWay Security Server RACF Auditor's Guide (SA22-7684)
- z/OS SecureWay Security Server RACF Security Administrator's Guide (SA22-7683)

REFERENCES.

- RACFICE2 describes a technique for analyzing RACF data using ICETOOL. You can obtain RACFICE2 at:
www.ibm.com/systems/z/os/zos/features/racf/downloads/racvice.html
- The DFSORT product tape contains a set of illustrative examples of interest to Storage Administrators and others who analyze data created by DFHSM, DFSMSrmm, DCOLLECT and SMF. The source for the following examples are available in sample job ICESTGEX:
 - DCOLEX1 - DCOLLECT Ex 1: VSAM report
 - DCOLEX2 - DCOLLECT Ex 2: Conversion reports

REFERENCES.

- DCOLEX3 - DCOLLECT Example 3: Capacity planning analysis and reports.
- DFHSMEX1 - DFHSM Example 1: Deciphering Activity Logs
- DFHSMEX2 - DFHSM Example 2: Recover a DFHSM CDS with a broken index.
- RMMEX1 - DFSMSrmm Example 1: SMF audit report.
- RMMEX2 - DFSMSrmm Example 2: Create ADDVOLUME commands.
- ICESTGEX is also available via anonymous FTP from:
[ftp.software.ibm.com/storage/dfsor/mvs/](ftp://software.ibm.com/storage/dfsor/mvs/)

System z Social Media Channels

- Top Facebook pages related to System z:
 - [IBM System z](#)
 - [IBM Academic Initiative System z](#)
 - [IBM Master the Mainframe Contest](#)
 - [IBM Destination z](#)
 - [Millennial Mainframer](#)
 - [IBM Smarter Computing](#)
- Top LinkedIn groups related to System z:
 - [System z Advocates](#)
 - [SAP on System z](#)
 - [IBM Mainframe- Unofficial Group](#)
 - [IBM System z Events](#)
 - [Mainframe Experts Network](#)
 - [System z Linux](#)
 - [Enterprise Systems](#)
 - [Mainframe Security Gurus](#)
- Twitter profiles related to System z:
 - [IBM System z](#)
 - [IBM System z Events](#)
 - [IBM DB2 on System z](#)
 - [Millennial Mainframer](#)
 - [Destination z](#)
 - [IBM Smarter Computing](#)
- YouTube accounts related to System z:
 - [IBM System z](#)
 - [Destination z](#)
 - [IBM Smarter Computing](#)
- Top System z blogs to check out:
 - [Mainframe Insights](#)
 - [Smarter Computing](#)
 - [Millennial Mainframer](#)
 - [Mainframe & Hybrid Computing](#)
 - [The Mainframe Blog](#)
 - [Mainframe Watch Belgium](#)
 - [Mainframe Update](#)
 - [Enterprise Systems Media Blog](#)
 - [Dancing Dinosaur](#)
 - [DB2 for z/OS](#)
 - [IBM Destination z](#)
 - [DB2utor](#)



Complete your session evaluations online at www.SHARE.org/Seattle-Eval

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.



AIX*	DB2*	DFSORT	IBM*	Language Environment*	Redbooks*	System Storage	System z10 Business Class	z10 EC
BladeCenter*	DFSMS	Domino*	IBM eServer	MVS	REXX	System x*	Tivoli*	zEnterprise*
BookManager*	DFSMSdss	DS6000	IBM logo*	Parallel Sysplex*	RMF	System z	WebSphere*	zSeries*
CICS*	DFSMSHsm	DS8000*	IMS	ProductPac*	ServerPac*	System z9	z10	
DataPower*	DFSMSrmm	FI CON*	InfinBand	RACF*	SYSREXX	System z10	z10 BC	

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Windows Server and the Windows logo are trademarks of the Microsoft group of countries.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

* Other product and service names might be trademarks of IBM or other companies.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice.

Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

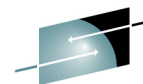
Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

This information provides only general descriptions of the types and portions of workloads that are eligible for execution on Specialty Engines (e.g. zIIPs, zAAPs, and IFLs) ("SEs"). IBM authorizes customers to use IBM SE only to execute the processing of Eligible Workloads of specific Programs expressly authorized by IBM as specified in the "Authorized Use Table for IBM Machines" provided at www.ibm.com/systems/support/machine_warranties/machine_code/aut.html ("AUT"). No other workload processing is authorized for execution on an SE. IBM offers SE at a lower price than General Processors/Central Processors because customers are authorized to use SEs only to process certain types and/or amounts of workloads as specified by IBM in the AUT.

Complete your session evaluations online at www.SHARE.org/Seattle-Eval





SHARE
Educate • Network • Influence



Complete your session evaluations online at www.SHARE.org/Seattle-Eval

2/25/2015

