

## Appendix A. STARWARP Release 6.1.0

---

STARWARP 6.1.0 has the following enhancements over the prior STARWARP 5.4.0 release of May 31, 1999.  
This is modification 5 (6.1.0 - 2000.001 – January 1, 2000).

### MAP Subcommand

The REPLACE keyword can be used in conjunction with the RELINK keyword to indicate that all CSECTS are to be replaced before the normal INCLUDE statement as in the following example:

```
>----->map dsat relink replace

** MAP          DSAT
//LKED          EXEC  PGM=IEWL,
//              PARM= 'NCAL,MAP,LIST,LET,RENT,REUS,REFR '
//SYSUT1        DD   UNIT=SYSDA,SPACE=(2048,(200,20))
//SYSPRINT      DD   SYSOUT=*
//SYSLIB        DD   DISP=SHR,DSN=WSER07.LINK.LOAD
//SYSLMOD       DD   DISP=SHR,DSN=WSER07.LINK.LOAD
//SYSLIN        DD   *
REPLACE DSAT,DSATPDS,DSATPCL
INCLUDE SYSLIB(DSAT)
ORDER   DSAT,DSATPDS,DSATPCL
SETCODE AC(1)
ENTRY   DSAT
NAME    DSAT(R)
```

The REPLACE keyword can also be used with the MODULE keyword to select the CSECT names for replacement:

```
>----->map dsat relink replace module(dsatp*)

** MAP          DSAT
//LKED          EXEC  PGM=IEWL,
//              PARM= 'NCAL,MAP,LIST,LET,RENT,REUS,REFR '
//SYSUT1        DD   UNIT=SYSDA,SPACE=(2048,(200,20))
//SYSPRINT      DD   SYSOUT=*
//SYSLIB        DD   DISP=SHR,DSN=WSER07.LINK.LOAD
//SYSLMOD       DD   DISP=SHR,DSN=WSER07.LINK.LOAD
//SYSLIN        DD   *
REPLACE DSATPDS,DSATPCL
INCLUDE SYSLIB(DSAT)
ORDER   DSAT,DSATPDS,DSATPCL
SETCODE AC(1)
ENTRY   DSAT
NAME    DSAT(R)
```

## DISASM Subcommand

Support for keyword MVS370 was dropped; if this keyword is used, MVSXA is assumed instead.

DISASM was modified to create only DC instructions in the entry code of a routine if the routine begins with an unconditional branch.

Support for IBM's new relative and immediate instructions was added. This added the following new operation codes: AHI, BRAS, BRC, BRCT, BRXH, BRXLE, BSA, BSG, CHI, CKSM, CLCLE, CLST, CUSE, LHI, MHI, MS, MSR, MVCLE, MVPG, MVST, PLO, SACF, SQDR, SQER, SRST, TMH, and TML.

## COMBINE, COPY, CREATE, and SEPARATE Subcommands

The COMBINE, COPY, CREATE, and SEPARATE subcommands were changed to issue an error message for new fixed blocked data sets if the LRECL is not an even multiple of the BLKSIZE with messages:

```
PDS482W Blocksize should be evenly divisible by the DCB LRECL
PDS984E CREATE terminated due to error
```

## COMPDIR Subcommand

The COMPDIR subcommand was changed to support SEQ and NOSEQ keywords in conjunction with the CHANGED and NOCHANGED keywords.

If SEQ is coded, the comparison ignores standard sequence number fields for fixed or variable length records.

With the default NOSEQ keyword, each data record is compared and any sequence number field is considered part of the data.

## Binder PM3 Support

STARWARP was modified to support Program Management Level 3 or PM3 level program objects for systems with DFSMS 1.4 or above. For earlier DFSMS levels, STARWARP continues to support the first level or PM1 modules.

STARWARP subcommands FIND, HISTORY, MAP, LIST, REPLACE, and XREF were modified to provide this new support.

## CSECTS Larger than 16 Megabytes

STARWARP PM3 support includes new support in ALIAS, ATTRIB, FIND, IF, HISTORY, LIST, MAP, REPLACE, VERIFY, and XREF for CSECTS larger than 16 Megabytes and for modules larger than 16 Megabytes.

To support these large CSECTS, several messages were modified:

- PDS141I message from FIND, LIST, and REPLACE for CSECT offsets
- MAP messages for CSECT lengths in hexadecimal
- PDS103I message from MAP for the entry point address
- PDS104I message from MAP for the module length in hexadecimal

## LISTC, 3.4, and MASK Service

The STARWARP LISTC service was modified to use the system CSI instead of the SUPERLOCATE service. This provided performance improvements and permits STARWARP to use the same data set name syntax as ISPF 3.4 and ISMF as well as other system services.

In performance testing, using the CSI service gave STARWARP LISTC about 50 percent better performance than SUPERLOCATE.

## STARBAT Input Member Names

STARBAT supports MEMBER=(member1,member2, ...) as well as MEMBER=memname as in previous releases.

STARBAT also supports several new member name forms in the MEMBERS keyword to permit member name forms used in the rest of STARWARP. For additional information, see the Member Name Forms Appendix in the STARWARP Reference manual.

The following member name support was added for the MEMBERS keyword:

1. Member name range. For example, MEMBERS=ABC:DEF
2. Member name pattern. For example, MEMBERS=CUT/N2
3. Member name combination. For example, MEMBERS=COMB\*DF
4. Member name placeholder. For example, MEMBERS=CO?BINE,  
MEMBERS=C?T/?2 or MEMBERS=CO%%\*DF

## STARBAT Concatenated Inputs

If data sets are concatenated for STARBAT input, STARBAT reports on the number of records read from each data set in order as shown in the following sample:

```
    2 Records were input from VOL=SER=SER002,DSN=WSER07.LIB.CNTL
    4 Records were input from VOL=SER=SER002,DSN=WSER07.GDG.G0004V00
    3 Records were input from VOL=SER=SER002,DSN=WSER07.GDG.G0003V00
    5 Records were input from VOL=SER=SER002,DSN=WSER07.GDG.G0001V00
    60 Records were input from VOL=SER=SER002,DSN=WSER07.LIB.PDSE
    3 Records were input from VOL=SER=,DSN=CARDFILE
    4 Records were input from VOL=SER=SER001,DSN=WSER07.SDSF.LIST
*** End of input data set
PDS051I Data set was copied; input=148; output=148
```

## STARBAT BSAM Processing

STARBAT was changed to support PARM=BSAM to turn off EXCP processing for sequential and partitioned data sets. This will probably only be used for diagnostics and support.

## STARBAT Record Columns

STARBAT was changed to support record columns in the PRINT and PRINTHEX operations. This displays at the start of each record image for records longer than 100 characters.

## Appendix B. STARWARP Release 5.4.0

---

---

STARWARP 5.4.0 has the following enhancements over the prior STARWARP 5.3.1 release of October 16, 1998. This is modification 4 (5.4.0 - 1999.151 – May 31, 1999).

### Copybook Bit Arrays

The STARTOOL and STARWARP copybook parser was modified to support PL/1 aligned and unaligned bit arrays. This supports copybook elements that may not be on a byte boundary.

The STARTOOL and STARWARP copybook parser was also modified to support PL/1 multidimensional arrays and declarations with bounds. This is accomplished by converting each multidimensional array into an equivalent single dimension array to get the total array size.

With this support, the following example is valid:  
DCL LOSDEX(1:5, -5:1) CHAR(2);

### Copybook Slack Bytes

Copybook support was changed to support slack bytes between array elements as defined in COBOL.

For the following copybook:

```
2 INSTALLATION.  
3 FIRSTX          PIC XX.  
3 NAME OCCURS 4 TIMES.  
5 FIRST          PIC X.  
5 SECOND        PIC 9(6) COMPUTATIONAL SYNC.
```

We provide the following mapping:

LEVEL AND NAME	PICTURE COLUMN	TYP	LENGTH
01 UNKNOWN			
2 INSTALLATION			
3 FIRSTX	XX @ 1	CHR	2
3 NAME	OCCURS 4 TIMES		
5 FIRST	X @ 3	CHR	1
5 SECOND	4 BIN @ 5	BIN	4
3 NAME	OCCURRENCE # 2		
5 FIRST	X @ 11	CHR	1
5 SECOND	4 BIN @ 13	BIN	4

## STARBAT OPTIONS=JCL

STARBAT supports OPTIONS=JCL. This means that STARBAT can determine JCL control statements and continuations; and each statement can be processed logically. STARBAT recognizes DD, EXEC, JCLLIB, JOB, OUTPUT, PROC, SET JCL statements and their logical continuations.

If a line in the member is not one of these statements or its continuation, STARBAT will use normal change and update processing rules for that single line. Following lines can then be searched or changed in either JCL or normal mode.

JCL statements must begin with a // in column 1 and comment statements are considered a part of the concatenation. Thus, the following would be considered a single statement:

```
//STEPABC EXEC PGM=STARBAT,PARM=' BATCH ' ,  
// * THIS IS A COMMENT IN THE MIDDLE OF A CONTINUED STATEMENT  
// TIME=4,REGION=4096K
```

To search a JCL structure, use the IF or OR statements. You can also search and update JCL statements with the CHANGE, CHANGEALL, OVERLAY and OVERALL statements. Be sure to use a length of zero for the search length of these parameters if you wish to search the entire JCL statement.

The OVERLAY and OVERALL parameters update data but do not shift data columns so they can be used without problems by any of the COPY operations, MULTICOPY or the UPDATE operations.

CHANGE and CHANGEALL can cause data shifting if the search and replacement strings differ in length. If the string is shortened, STARBAT follows normal JCL rules and is able to update any of the COPY, MULTICOPY or UPDATE operations. If the string is expanded, STARBAT adjusts the JCL statement to fit the statement in the new string using normal JCL rules.

JCL expansion is performed using the following logic to minimize changes in current JCL statements:

1. If the string will fit, the JCL text is expanded to the right on the changed JCL line.
2. If this expansion fails and the string will fit, the JCL text is added after shifting the changed JCL line left and right.
3. If both of the above expansions fail, the JCL line to be changed is split at the previous comma and continued on another line and the added line is expanded as needed. The new JCL line is indented to match the split JCL line.
4. If the above expansion fails because there is not a previous parameter, STARBAT scans for the next comma to the right and splits the JCL statement after that parameter. Again, the new JCL line is indented to match the split JCL line.

STARBAT can not split a JCL line for UPDATE operations. In this case, an expansion error will be indicated and the return code for the job will be set to eight.

Restrictions:

1. A JCL statement may contain up to 50 JCL lines.
2. OPTIONS=JCL and MOVE from input to output is not supported.
3. The JCL is assumed to be syntactically correct; however, STARBAT does not issue syntax messages for incorrect JCL.
4. To be considered JCL, the statements must reside in a PDS member and the data set must have a fixed record format with 80 character records.
5. Only data between columns 1 through 71 are considered for determining control statement boundaries and continuations.

## STARBAT Multiple Groups

STARBAT was modified to support leaving the input and output data sets open until the DDNAME is changed.

Normally, data set positioning is maintained between groups of STARBAT control statement so that each group begins where the previous group left off. However, if the previous control statements reached the end of the input data set, the action taken depends on the value of the OPTIONS parameter.

If OPTIONS=MULTI is specified, the data set will be repositioned and reread; otherwise, the default action (OPTIONS=NOMULTI) is assumed and another end of data set condition will be indicated.

Note that OPTIONS=MULTI is only effective after an end of file.

## STARBAT Reverse Processing

STARBAT reverse operations for sequential and VSAM data sets were implemented.

New operation codes:

<b>COPYREV</b>	copy starting from the end of the data set
<b>SKIPREV</b>	skip records in backwards order
<b>PRINTREV</b>	print backwards in list format
<b>PRINTCHRREV</b>	print backwards in character format
<b>PRINTHEXREV</b>	print backwards in hexadecimal format

Note that you may not follow a reverse operation like **COPYREV** with a forward operation like **COPYALL**; in the same way, you can not follow a forward operation with a reverse operation.

## STARBAT Syntax Conversion

SERENA provides four different methods to convert STARBAT syntax from the previous format to the new format defined for the 5.4.0 level:

1. **PDSCVMAC** is an edit macro provided to convert control statements. From edit, just enter **PDSCVMAC**.
2. **PDSCONVJ** is a guided dialog to prepare JCL and control statements for a conversion. From StarWarp, enter TSO %PDSCONVJ.
3. **PDSCVJOB** is a sample STARBAT job that can be used to convert STARBAT control statements to the new control statement syntax.
4. **@ STARBAT** is a sample source stub that can be used as the basis of a StarBat alias table. With this approach, StarBat can utilize your previous StarBat syntax and convert keywords dynamically.

## STARBAT T Strings

STARBAT was modified to support T'abc' literal strings. This means that `IF=(1,0,T'abc')` would locate "abc", "ABC" and "AbC" similar to ISPF. Note that T'abc' strings are also supported for CHANGE and OVERLAY as shown in the example below:

```
DD01 COPYREC CHANGE=(1,0,T'someword',C'Sword'),
OVERLAY=(1,0,T'abc',C'ABC'),
```

## STARBAT Relative Numbers

STARBAT was modified to not change relative numbers with each reference. If you would like to customize STARBAT to leave this processing as in the previous release, please contact SERENA.

For example:

IF=(1,0,C'A'), IF=(+2,EQ,C'C'), IF=(+3,EQ,C'D') would match ABCD but not ABCXXD.

## Assembler Copybooks

Copybook support was changed to support assembler as well as COBOL and PL/I copybooks.

The assembler support has several restrictions:

1. Only DC and DS statements are supported
2. Expressions are not supported:  
FIRST4K DS CL(WORKAREA+4096-\*)
3. ORG statements are not supported:  
ORG PREVSYMB
4. EQU statements are ignored:  
SAVLEN EQU MSYMBOL

## STARBAT Visual Age MLE Support

The STAR TOOL and STAR WARP copybook parser added support for VISUAL AGE MLE "DATE FORMAT IS YYXXX" COBOL data declaratives to define dates. Each MLE item is marked with \*DATE\* in the DATA/MSG field of the copybook table so that the programmer can position to these fields with an F \*DATE\* command.

## SMPGEN Subcommand

The SMPGEN subcommand was modified to generate ALIAS parameters instead of MALIAS parameters for any alias members if the MAC and TYPE(xx) keywords are used.

## VERIFY Message Changes

VERIFY was modified to show the cumulative character count for sequential data sets as in the following example:  
PDS117I Cumulative size is 223,192 characters

Message PDS860E was modified to add the member name for VERIFY (PDS860E MEMBERNAME is an alias but no main member exists). Also, message PDS867E was modified to add "of XXXX" for the member name (PDS867E ISPF current member size (33) is mismatched with actual size).

VERIFY for partitioned members and sequential data sets was changed to provide a summary message showing the largest LRECL or BLKSIZE encountered if PDS806E, PDS812E or PDS813E messages are issued during execution as in these examples:

PDS806E Block length of 8,992 exceeds the MAXBLK value  
PDS812E Block length of 10,512 exceeds the BLKSIZE value  
PDS813E Record length of 254 exceeds the maximum DCB LRECL

## AMODE and RMODE Support

The **MAP** command was modified to support changing AMODE and RMODE at a CSECT level. If the **MODULE** keyword is not used, all CSECTS in the member are modified. If the RMODE or AMODE is modified, the **MAPMOD** keyword is validated with the security system.

The following syntax is used:

```
MAP member MODULE(xyz) RMODE24/RMODEANY
                        AMODE24/AMODE31/AMODEANY
```

The **CSECT** function also supports changing AMODE and RMODE at a CSECT level. The **MAP** line command for a CSECT supports an operand of **RMODExx** or **AMODExx** in the **DATA/MSG** field.

## FINDMOD Dynamic Changes

The **FINDMOD** subcommand was changed to support dynamic LPA library searches. This involves searching the in-storage queue of LPA modules added dynamically by the operating system.

The **FINDMOD** subcommand was also changed to support searching libraries in the dynamic linklist. **FINDMOD** now supports both static and dynamic linklists.

## LISTC and WORKPAD Dynamic Linklist

The **LISTC** and **WORKPAD** function **LOAD LNKLIST** command was changed to support loading the names of libraries in the dynamic linklist. **LOAD LNKLIST** now supports both static and dynamic linklists.

## STARWARP ACTION Change

**STARWARP ACTION=ADJUST** was modified to perform correction actions similar to **ALIGN**. Now, **ADJUST** will move a resulting date to the end of the month if it is invalid (as before) or if the original date started at the end of a month (like **ALIGN**).

## STARWARP WORKDAYS Parameter

The **STARWARP WARP** parameter was changed to support **WORKDAYS=0123456** (where 0 is Sunday and 6 is Saturday).

This parameter may be used to define which days of the week are working days; **WORKDAYS=12345** is the default. Please contact **SERENA** if you would like to use this parameter.



## Appendix C. STARWARP Release 5.3.1

---

STARWARP 5.3.1 has the following enhancements over the prior STARWARP 5.3.0 release of June 1, 1998. This is modification **3 (5.3.1 - 1998.289 – October 16, 1998)**.

### Externalized WARP Functions

STARWARP was changed to externalize WARP functions in a called routine.

With this change, STARWARP date and monetary conversions can be invoked by a COBOL, PL/1, FORTRAN or assembler routine. SERENA provides sample routines written in COBOL, PL/1 and assembler called SAMPDATC, SAMPDATP and SAMPDATA respectively in our "assemble" distribution library.

To use this facility, the programmer adds calls in their routine to perform functions supported by the STARWARP WARP function.

As a short example, you could code the following COBOL segment to add 5 months to a date and move it to the next business day:

```
MOVE THE-DATE TO DATEX1.
MOVE 'C ' TO CONVERT1.
MOVE 'CCYY/MM/DD ' TO PICTURE1.
MOVE '5M ' TO ADD1.
MOVE 'PDS#HOLI ' TO HOLIDAY1.
MOVE 'NEXTBDAY ' TO ACTION1.
CALL 'PDSFDATE' USING RETURN1, ERROR-MSG, ANCHOR, DATEX1,
    CONVERT1, PICTURE1, DATE1, ADD1, SUB1, IGNORE1,
    FISCAL1, PIVOTYR1, BASEYR1, ACTION1, VALID1,
    HOLIDAY1, OUTFORM1, OUTPIC1, DIV1,
    MULT1, CONV1.
IF RETURN1 = 0 THEN MOVE DATEX1 TO THE-DATE.
```

### STARWARP Date Exits

STARWARP date exits were changed to support ignoring dates as with the IGNORE parameter. This is done with date exit return codes; you set the return code to 0 for a good conversion, 4 for an error and 8 if you wish to ignore a date.

## Day of Week Actions

Several new actions were added to support moving dates to the nth day of week in a month. These actions also support business rules.

The syntax is **ACTION=MON/TUE/WED/THU/FRI/SAT/SUN 1/2/3/4/LAST {B/P}** where the number or LAST is used for nth and **B** means move a date to the next business day of the month and **P** means the previous business day.

Valid actions would be SUN1, SUN1B, THU4, MONLASTB, FRILAST, WED4 and FRI3P.

## Day of Month Actions

Some additional actions were added to support moving a date to a specific day of a month. These actions also support business rules.

The syntax is **ACTION=DAY # {B/P}** where # is 1 (or 01) through 31 and **B** means move a date to the next business day of the month and **P** means the previous business day.

Valid actions would be DAY1, DAY1B, DAY1P, DAY01, DAY10B and DAY31P. Note that if the day is set after the end of the month, it will be adjusted to the last day of the month before taking any B or P action.

## GDG support

STARTOOL and STARWARP were changed to support GDG data sets better.

- CREATE and MODEL support **GDG(+1)** creation
- COPY, DUP and COMBINE can output to a **GDG(0)** or **GDG(-n)** data set.
- COPY, DUP and COMBINE can output to a **GDG(+1)** data set if **NEW** is specified.

## Mixed Release Level Support

STARTOOL and STARWARP were modified to support fixed alias entry points regardless of release level. Mixed levels of STARTOOL or STARWARP (for example, 5.2.1 and 5.3.0 together) previously caused ABEND S0C6 and other problems. Starting with this release, STARBAT has entry address X'000050' and all other STARTOOL and STARWARP aliases have entry address X'000040'.

## Variable Spanned Records

Support has been added to read VS and VBS records with QSAM; PBROWSE has not been changed to support spanned records yet.

## STARBAT Multiple VSAM Outputs

STARBAT now supports VSAM output data sets for MULTICOPY operations. In support of this modification, we changed the STRB05I message as shown in the following example:

```
STRB05I DDNAME=DD051 DSN=VSAM.DATA.SET.NAME opened for VSAM output
STRB05I DCB=(VS-KSDS ,LRECL=850 ,BLKSIZE=1514) ,VOL=SER=SER001
```

The VS-type field will be KSDS, ESDS, RRDS or LIN depending on the VSAM data set type. The LRECL field is the average record length and the BLKSIZE field is the maximum record length.

## Copybook Parsing

Copybook parsing was changed to continue until a lower level name is encountered. Previously, if you started with a 04 level, copybook parsing would terminate at the first 04 or 03 level. With this change, copybook parsing continues until a 03 or lower level is encountered.

## Compare VSAM Data Sets

The COMPARE subcommand was changed to support comparing two VSAM data sets with the COMPAREX utility from SERENA. The COMPAREW front end supplied with STARTOOL and STARWARP was also modified for VSAM support.

## KEYLISTS Added

PEDIT, MEMLIST source, ML load and LISTC were changed to support their own KEYLISTS and provide unique PF keys for different functions.

STARTOOL and STARWARP use the following KEYLIST names:

<b>PDSKEYL</b>	normal KEYLIST name which is used for most panels
<b>PDSKEYL2</b>	KEYLIST name used for scrollable tutorials
<b>PDSKEYL3</b>	KEYLIST name for PBROWSE
<b>PDSKEYL4</b>	KEYLIST name for PEDIT
<b>PDSKEYL5</b>	KEYLIST name for LISTC/LISTF
<b>PDSKEYL6</b>	KEYLIST name for MEMLIST source
<b>PDSKEYL7</b>	KEYLIST name for MEMLIST load



## Appendix D. STARWARP Release 5.3.0

---

STARWARP 5.3.0 has the following enhancements over the prior STARWARP 5.2.1 release of October 15, 1997. This is modification 2 (5.3.0 - 1998.150 - June 1, 1998).

### Cursor Selection

A new feature called "cursor selection" has been added. This feature is off by default but may be turned on in menu item 0.0 (SETALL) under "Set Environmental Options" on the line called "Cursor Selection".

If cursor selection is turned on, you may place the cursor on any of the yellow highlighted fields in a table column header and press ENTER to sort the table elements on that column. Note that even if cursor selection is turned off, you can still sort a column in the same way by pressing PF6 (RCHANGE) instead of ENTER.

Similarly, you can cause an action to be taken on a table entry by placing the cursor in the OPT field of the line and pressing ENTER (or PF6 if cursor selection is not enabled). The default option for this entry is MENU (a selection prompt is provided) but you can change the defaults in the SETSEL panel for each individual table. For example, if you change the option to S for MEMLIST, you can edit a member by placing the cursor on its OPT field and pressing ENTER.

The DATA/MSG field can also be cursor selected. The default action is to TAG the entry, and the default action can be changed in the SETSEL panel.

### STARBAT Improvements

- The new **COPY SOME** operation can be used in STARBAT to apply all changes to each record like COPYALL but only copy selected records like the COPY operation.
- STARBAT inputs can be entered in upper and lower case.

### STARBAT PARM='TSO'

STARBAT has been modified to support PARM='TSO'. This support changes the SYSLIST output such that page breaks are converted to triple spaces in the output to save paper; otherwise, each member output in the SYSLIST output begins on its own page.

This would be coded in your STARBAT JCL as follows:

```
//STEPNAME EXEC PGM=STARBAT, PARM='TSO'
```

Also, if no SYSLIST DD statement is present and STARBAT is executing under TSO, STARBAT will direct all SYSLIST output to SYSPRINT.

## STARBAT EXPAND

EXPAND=(column,type,size,newsize) was added as a STARBAT parameter to simplify date and data field expansions. **EXPAND** is similar to MOVE but it preserves the original record to the extent possible. Note that an **EXP** line command was added to the STARBAT dialog to specify **EXPAND** parameters.

Type can be **B** for Binary, **P** for Packed, **C** for Character or **N** for Numeric. Character type pads the original item on the right with blanks and the other formats add numeric padding to the left of each expanded field.

Note that character type is usually used for date expansions because STARWARP can read the date using the original format and rewrite it using the new date format. Other types of field expansion may use binary, packed or numeric character expansion rules if desired.

The column number referred to by the EXPAND parameter is the original column number in the data; similarly, references to column numbers by other parameters should also use the original data column number because STARBAT will adjust these columns dynamically.

## STARBAT TOTAL Operation

The TOTAL operation is now supported in STARBAT as well as the SUM parameter with the following syntax:

```
DD01  TOTAL  SUM=(LOC,LEN,B/C{,'DOC'}) / (LOC{,'DOC'})
```

TOTAL is a read-only operation for input files. SUM may be used with optional IF/OR/AND testing logic to accumulate the contents of numeric character, packed or binary fields. Note that the SUM parameter can also be used in other operations as in the following example:

```
DD01  COPYALL  IF=(22,EQ,C'COUNT'),  
        SUM=(27,4,C,'Total of the COUNT parameters')
```

If no length field is coded, the data is assumed packed and STARBAT can dynamically determine the length of the field. A field to be added may contain up to 15 decimal digits.

Binary fields can be one to four characters in length. Halfword and fullword binary items are assumed to contain a high-order sign bit. Character fields may contain one to 15 numeric digits and the last character may contain a signed digit (X'C0' for plus, X'D0' for minus).

## STARBAT Record Selection

Record selection by valid numeric or packed data support was added in the IF, AND and OR parameters with the following syntax: IF=(LOC,LEN,{DUPL}operator-type)

The location field is the first position to check and the length field indicates the number of characters to check. If 0 is used for packed data, STARBAT will scan for the end of the number. The DUPL number may be used to indicate a number of repeat checks.

Character fields may contain one to 255 numeric digits and the last character may contain a signed digit (X'C0' for plus, X'D0' for minus). Operator-type may be **EQP** to check for valid packed numbers, **EQN** to check for valid character numerics, **NEP** for invalid packed numbers or **NEN** to check for invalid character numerics.

For example:

```
IF=( 20 , 5 , EQN)      -- selects if 5 numeric characters
IF=( 20 , 5 , 2EQN)     -- selects if 2 consecutive numbers
IF=( 20 , 0 , NEP)      -- selects if non-numeric packed number
IF=( 20 , 5 , EQP)      -- selects if a packed number 5 characters long
IF=( 20 , 0 , 10EQP)    -- selects if 10 consecutive packed numbers
```

## STARBAT QSAM Output Message

The STARBAT STRB03I QSAM output is in use and STRB04I QSAM output is in use messages were replaced by the STRB05 message shown in the examples below:

```
STRB05I DDNAME=DD01O DSN=WSER07.LIB.CLISTV opened for BSAM/QSAM output
STRB05I DCB=(RECFM=VB,LRECL=254,BLKSIZE=11440),VOL=SER=SER002
or
STRB05I DDNAME=DD01OB DSN=WSER07.LIB.CLISTV opened for UPDATE
STRB05I DCB=(RECFM=VB,LRECL=254,BLKSIZE=11440),VOL=SER=SER002
```

## STARBAT Execution Messages

Many STARBAT execution message were enhanced to include an image of the failing parameter in the text of the error message as in the following example:

```
--- STRB75E WARP= Invalid character number at column 0012;
                                hex=X'F1F9F9F860F0F3'; char=C'1998-03'; at record 342
Text:                            WARP=(12,C,CCYYDDD,OUTPIC=CCYYDDD,OUTLOC=44,
```

## STARBAT Actions Taken

STARBAT was modified to output an "Actions Taken" section after each operation is completed. A competitor reports only on number of times each EDIT, MOVE and REPL parameter is executed but STARBAT provides more information as shown in the following example:

```
Actions taken for:      IF=(1,EQ,C'AA',C'AC'),          -----42 -----19
```

STARBAT reports on the AND, OR, SUM, EDIT, MOVE, REPL, STOP, WARP and WRITE parameters. The first number on an Actions taken line is the number of times the parameter was executed and the second number is a success count. This would be the number of times the parameter updated a record or the number of times the condition coded was true.

## STARBAT Binary Data

STARBAT was modified to better support binary data. In those parameters (like IF, OR, MOVE, ...) which accept character, hexadecimal and packed values. you may also specify halfword or fullword integer values as shown in the example below:

```
IF=( 22 , EQ , F'1234' ) , OR=( 55 , GT , H'12' ) , AND=( 63 , EQ , C'ABC' )
```

Constants beginning with an F will be saved as a binary fullword and constants beginning with an H will be saved as binary halfwords.

## STARBAT Bit Data

STARBAT was modified to better support bit operations. In those parameters (like IF, OR, EDIT, ...) which accept relational checks (EQ, NE, GT, GE, LT, LE), you may also specify bit value tests with several new test names: AO, AZ, NO and MX. STARBAT uses the assembler TM (test under mask) instruction to check the binary value of the masked bits as follows:

- AO** Tests for all one bits (A competitor calls this EQ)
- AZ** Tests for all zero bits (A competitor calls this NE)
- NO** Tests for not ones bits
- MX** Tests for mixed ones and zeroes

Examples:

To test for odd EBCDIC numbers IF=( 23 , AO , X' F1 ' )  
To test for even numbers IF=( 23 , NO , X' 01 ' )  
To test for EBCDIC 0, 4 and 8 IF=( 23 , AZ , X' 03 ' )  
To test for EBCDIC 1, 2, 5, 6, 9 IF=( 23 , MX , X' 03 ' )

Also, the STARBAT REPL and REPLALL parameters were modified to support bit operations. STARBAT can perform OR operations indicated by the letter **O**, NOT-AND (or binary minus) indicated by the letter **M**, AND operations indicated by the letter **A** and exclusive OR operations indicated by the letter **E**.

Examples:

To make four EBCDIC characters odd REPL=( 23 , OX' 01010101 ' )  
To clear the zones where ABCD is found REPL=( 23 , 20 , C' ABCD ' , MX' F0F0F0F0 ' )  
To clear the zones where ABCD is found REPL=( 23 , 20 , C' ABCD ' , AX' 0F0F0F0F ' )  
To flip several bits in an EBCDIC byte REPL=( 23 , EX' C3 ' )

## REMOVE Command Replaces EXCLUDE

The EXCLUDE command has been renamed to REMOVE in all ISPMODE tables to coordinate with the REMOVE command introduced for MEMLIST in the previous release. EXCLUDE can still be used as an alternate name for EXCLUDE at this time. Note that the EXCLUDE subcommand is also still available; it is based on the IF subcommand in STARWARP baseline processing.

## Migrated Data Set name

If STARWARP is invoked with no data set name, it will reuse the data set from the previous session. STARWARP now checks if that data set has been migrated before changing to it to avoid startup delays; if it is a migrated data set, STARWARP will use your ISPF profile data set instead.

## HISTORY Command Improvements

COBOL compiler support was enhanced in the HISTORY subcommand.

1. References to the COBOL 370 compiler were changed to COBOL for MVS and VM to conform with terminology currently used by IBM.
2. Modules translated with COBOL for MVS and VM version 11 were incorrectly marked as compiled by COBOL II. This error is corrected by indicating the correct compiler in the HISTORY PDS060I message.
3. Support was added for the COBOL for OS/390 and VM compiler.
4. For the COBOL for MVS and VM and the COBOL for OS/390 and VM compilers, additional support was added in the COBOL keyword to test for several additional compilation options: COMPCLASS, COMPPROGRAM, CURRENCY, NOCURRENCY, DATEPROC, NODATEPROC, INTDATELIL, INTDATEANSI, NUMCLSALT, NUMCLSPRIM, OPTFULL, NOOPTFULL, PGMLONGMIX, NOPGMLONGMIX, PGMLONGUPP, NOPGMLONGUPP, RMODEANY, RMODE24, TESTBLOCK, NOTESTBLOCK, TESTPATH, NOTESTPATH, TESTSTMT and NOTESTSTMT.

## STARWARP Panel

The STARWARP primary panel was modified to add STARWARP features; these options were previously available on the STARWARP option 4 panel.

## STARWARP Copybook Dialog

The STARWARP copybook dialog or the WARP function was changed to an internal dialog with many improvements:

1. The WARP function now operates in parallel with other STARWARP functions. This allows switching back and forth to other functions like MEMLIST, PEDIT and LISTC while leaving the WARP function active.
2. Multiple copybooks can be input at one time using extended copybook support
3. Panvalet and Librarian copybooks are supported
4. STARWARP supports warping a COBOL group item
5. A **D** (or Display) line command was added to display definitions
6. STARWARP supports multiple copybook table display formats
7. A STARWARP table can be saved to or retrieved from the GROUP data set or an named OTHER data set
8. The MAP primary command was added to display extended copybooks
9. OCCURS DEPENDING ON support has been improved
  - a. STARWARP keeps track of how wide an OCCURS group is
  - b. STARWARP is aware of where the dependent variable is
  - c. STARWARP supports warping dates after an OCCURS unit
10. The WARP dialog table may be displayed in any of five different formats. To display the table in a different format, press the LEFT or RIGHT PF key and you will switch to another table view.
11. The WARP function can create a batch job and submit it as before or execute in the foreground.
12. A **SHRINK** primary command was added; it deletes all table lines except for control statements. This can be used just before a **BUILD** command to reduce the size of the copybook table and speed up the build process.
13. A **REFRESH** primary command was added; it deletes all table lines except for control statements and rebuilds the copybook table. This could be used in some cases to apply multiple copybooks against one data set.
14. The **EXP** (EXPAND) line command or a WARP statement can be used to expand date or data fields.
15. If the output data set does not exist, it can be modeled after the input data set. If you use this option, STARWARP is executed separately for a MODEL command; just change any desired parameters and press ENTER to create the data set. You may then enter MODEL to correct any problems or to create additional data sets and press END to return to the WARP function prompting.

## Euro Currency Conversion

STARWARP has been modified to support currency conversions between European member states.

This support required several modifications to STARWARP numeric support (this is for numeric pictures made up of an S character and one or more 9 digits):

1. The WARP ADD= subparameter was enhanced to support decimal points
2. The WARP SUB= subparameter was enhanced to support decimal points
3. The MULT= subparameter was added to perform currency conversions.
4. The DIV= subparameter was added to perform currency conversions.
5. The CONV= subparameter was added to look up conversion factors.
6. Input pictures and OUTPIC= were modified to also support V (implied decimal) picture characters.

To convert from the currency of a member state to equivalent Euro units, a division is normally required. A multiplication is normally required to convert from euros to the currency of a member state. In both cases, the result must be rounded after the calculation. To convert between two currencies, you must first convert to euros and then convert to the final currency. In this case, a division and a multiplication are required before rounding the result.

To assist in these conversions, you may use the new CONV= subparameter. This subparameter will invoke an exit named PDS#CONV (supplied in source), pass it a description of the conversion to be performed and obtain the division and/or multiplication factors that are required for use. On January 1, 1999 the conversion constants for all initial 11 member states will be frozen at a fixed rate in respect to the euro and the constants in PDS#CONV can be updated permanently.

Examples:

- For a conversion like French francs to euros, you would enter CONV=FRFEUR as a token and the exit would return: DIV=6.45863. 100 French Francs would convert to 15.48 Euros
- For a conversion like euros to French francs, you would enter CONV=EURFRF as a token and the exit would return: MULT=6.45863. 100 Euros would convert to 645.86 French Francs
- For a conversion like Deutsche Marks to French Francs, you would enter CONV=DEMFRF as a token. The exit will first look up DEMEUR and then EURFRF to return: DIV=1.92573,MULT=6.45863. 100 Deutsche Marks would convert to 335.39 French Francs after rounding

This type of lookup requires six character names made up of two three character pieces which use the defined euro names shown below.

<b>BEF</b>	Belgian Franc (smallest currency unit is 1.00)
<b>LUF</b>	Luxembourg Franc (smallest currency unit is 1.00)
<b>DKK</b>	Danish Krone (choose not to join initially)
<b>DEM</b>	Deutsche Mark
<b>GRD</b>	Greek Drachma (not actually eligible)
<b>ESP</b>	Spanish Peseta (smallest currency unit is 1.00)
<b>FRF</b>	French Franc
<b>IEP</b>	Irish Punt
<b>ITL</b>	Italian Lira (smallest currency unit is 1.00)
<b>NLG</b>	Netherlands Guilder
<b>ATS</b>	Austrian Schilling
<b>PTE</b>	Portuguese Escudo (smallest currency unit is 0.10)
<b>FIM</b>	Finish Markka
<b>SEK</b>	Swedish Krona (choose not to join initially)
<b>GBP</b>	United Kingdom Pound Sterling (choose not to join initially)

## Euro Calculator

Option 4E (EUROCALC) has been added to perform currency calculations for European member states. This calculator is provided so that sample currency calculations can be made in testing the euro support of STARWARP.

To use this calculator, note the following:

1. "Input euro picture" is the picture to be used for the beginning value.
2. "Monetary value" must be entered with leading zeroes and left adjusted; for convenience, you may enter a decimal point in the position indicated by a V character in the input picture; but it is ignored.
3. "Output euro picture" can be different from the input picture to specify formatting of the calculated value.
4. "Type of action" should be ADD, SUB, MULT, DIV or CONV to select an operation for the monetary value (this will usually be MULT, DIV or CONV for monetary conversions).
5. "Data adjustment" will be the factor or CONV name to be applied against the input monetary value.
6. "Exit library name" may be specified if the PDS#CONV load member is normally referenced with an ISPF LIBDEF command.

## Miscellaneous

- Extended copybook support has been enhanced to support Panvalet or Librarian copybook members.
- The REPLACE subcommand for FIND/REPLACE was enabled for STARWARP users.

## Holiday Tables

Option 4G (HOLIDAYC) was added to support building a STARWARP holiday table from date inputs representing business holidays in CCYYMMDD format.

When you select this option, STARWARP prompts for the name of the data set (and member) containing the holiday date input and for the name of an output data set and member to save the resulting holiday table.

After prompting for these data set names, STARWARP analyzes the input and builds a source holiday table which is compatible with the PDS#HOLx holiday tables defined for STARWARP in the previous release. You should submit the job set if you want to assemble and link the holiday table.

This permits you to maintain holiday dates in an alternate date format; this input holiday data set is also compatible with TransCentury calendar inputs.

This option builds a source member similar to SAMPHOLI in the STARWARP source library without requiring that you modify assembler instructions.

Several changes were made to support holiday processing through exit routines:

1. A sample exit for the U.K. was added with calendar dates between 1990 and 2030.
2. Program entry and exit code was modified to avoid excessive GETMAIN and FREEMAIN activity
3. The sample exits were changed to not issue an error message if a date is not found in the calendar table. Instructions in the sample programs explain how to turn on messages for program testing.

## New STARBAT Warp Parameters

The **ADD** and **SUB** parameters were modified to support business day logic. If no holiday calendar is in use, STARWARP will add week days (but not weekend days) until the correct week day has been reached; if a holiday calendar is in use, STARWARP will also screen out holidays to obtain the correct business day.

Examples:

To add 10 days to a date	WARP=( 15 , C , YYDDD , ADD=10D )
To add 10 week days to a date	WARP=( 15 , C , YYDDD , ADD=10B )
To add 10 business days to a date	WARP=( 15 , C , YYDDD , ADD=10B , HOLIDAY=PDS#HOLI )

**IGNORE=B/L/H/Z/9/P/I** with letters in any order may be entered to bypass changing certain invalid or special value dates. The letters have the following meanings:

<b>B</b>	ignores blanks
<b>L</b>	ignores low values
<b>H</b>	ignores high values
<b>Z</b>	ignores all 0 dates
<b>9</b>	ignores all 9 dates (including 9/9/99)
<b>P</b>	ignores permanent retention dates like 99365 for YYDDD
<b>I</b>	ignores invalid input dates like February 30, 1998

**FISCAL** defaults to 1 and may be set from 1 to 12 to indicate the start of a fiscal year for ACTION processing.

Several new **ACTION** parameters were added:

<b>FYEND</b>	Moves a date to the end of the fiscal year
<b>FYLBD</b>	Moves a date to the last business day of the fiscal year
<b>FYFBD</b>	Moves a date to the first business day of the fiscal year
<b>FQEND</b>	Moves a date to the end of the fiscal quarter
<b>FQLBD</b>	Moves a date to the last business day of the fiscal quarter
<b>FQFBD</b>	Moves a date to the first business day of the fiscal quarter
<b>NEXTSDAY</b>	Moves a date to the next same day of week as starting if needed
<b>PREVSDAY</b>	Moves a date to the last same day of week as starting if needed
<b>BALIGN</b>	Moves a date to the last business day of the month if started

**ERRWRITE=ddname** was added to permit selecting records which contain invalid date or data values. If the ERRWRITE subparameter is present, a record will be written to this data set if the date is considered invalid by STARWARP (a STRB75E error message will also be output) or a date exit routine indicates that a date is invalid.

**PIVOTYR** defaults to 70 and is a pivot year for windowing two digit dates. PIVOTYR is still defined as a parameter; however, it should be used with a WARP or WARPDEF parameters; it may be entered as 0 through 99.

**BASEYR** defaults to 1900 and it provides CC digits for windowing two digit dates. BASEYR may be entered as 1500 through 9800 and it should always be a multiple of 100.

BASEYR and PIVOTYR are used to window dates as follows:

```
CCYYDATE=date
  IF CCYYDATE<100 THEN
    IF CCYYDATE<=PIVOTYR THEN CCYYDATE=CCYYDATE+100
    CCYYDATE=CCYYDATE+BASEYR
  END
```

## WARPDEF Parameter

The WARPDEF parameter was added to support default parameters for the WARP parameters. Parameters defined with WARPDEF stay in effect until they are redefined or nullified and a WARP parameter may also override the WARPDEF default. Currently, the **ADD**, **SUB**, **ACTION**, **VALID**, **HOLIDAY**, **IGNORE**, **FISCAL**, **BASEYR** and **PIVOTYR** subparameters are supported as shown in the following example.

```
DD01 COPYALL WARPDEF=ADD=5Y,
           WARPDEF=( ACTION=NEXTBDAY, HOLIDAY=PDS#HOLI ),
           WARP=( 15, C, CCYY/MM/DD ),
           WARP=( 29, C, CCYY-DDD ),
           WARP=( 44, C, YY/MM/DD, ADD=66M )
           WARPDEF=( ACTION=, HOLIDAY=, ADD=1Y ),
           WARP=( 6, C, CCYY/DDD )
```

## STARWARP Date Pictures

STARWARP date pictures for WARP and OUTPIC were changed to support a wider variety of date formats.

A date may be composed of CCYY, CYY, HYY, YY, MM, DDD or DD components in any order with any of the following combinations:

```
CCYY/CYY/HYY/YY
CCYY/CYY/HYY/YY DDD
CCYY/CYY/HYY/YY MM
CCYY/CYY/HYY/YY MMDD
```

For dates which contain all numeric characters, CC99, C99, H99, or 99 may be used instead of a year value to indicate nine's compliment dates. For character dates, **MON** may be used interchangeably with **MM**.

For character dates, up to ten separator characters may be added between individual date components. Slash, period or dash characters may be used as separators; in addition, a question mark may be used to indicate that any character is acceptable at a location.

In addition to the above formats, the following new date pictures are supported:

<b>MM</b>	where YY is taken from the MVS system date and DD is assumed 01
<b>MMDD</b>	where YY is taken from the MVS system date
<b>YDDD</b>	where this is always considered a future date from next decade
<b>YMMDD</b>	where this is always considered a future date from next decade
<b>PDS...</b>	is used as a date exit name instead of DYNDATEA meaning PDS#EXTA

In the previous STARWARP release, only DYNDATEA and DYNDATEB could be used as exit names to map to exit modules PDS#EXTA and PDS#EXTB. In this release, a STARWARP date exit can have any name that begins with 'PDS'.

As an aid in testing exit routines, you may enter a date value as shown in the following examples:

```
WARP=( 10, B, PDS#EXTF, DATE=19991231, . . . (for picture CCYYMMDD)
WARP=( 10, P, PDS#EXTG, DATE=19991231, . . . (for picture CCYYMMDD)
WARP=( 10, C, PDS#EXTH, DATE=1999/12/31, . . . (for picture CCYY/MM/DD)
```

STARWARP will convert a binary or packed date and store the date value into the record before calling the input date exit.



## Appendix E. STARWARP Release 5.2.1

---

---

STARWARP was introduced as a subset of STARWARP in this release. STARWARP includes all facilities needed to warp dates and manage a year 2000 conversion. STARWARP does not include PEDIT, PVIEW or several other data management subcommands such as COPY or FIXPDS.

STARWARP 5.2.1 has the following enhancements. This is modification **1 (5.2.1 - 1997.288 - October 15, 1997)**.

### Warping Dates

STARBAT was modified to support "warping" dates in data files and updated for several other types of support.

The following STARBAT changes were made for this release:

1. Concatenation of unlike data sets is permitted (disk and tape).
2. Packed decimal data items are supported.
3. Literals like C'AB,CD' are interpreted as C'AB' or C'CD' as for a competitor.
4. If STARBAT SYSIN input is missing or contains no control statements, STARBAT copies all matching //DDnn input data sets to //DDnnO output data sets.
5. **MAXDATERR** was added to specify the maximum number of date errors.
6. **PIVOTYR** was added to specify a pivot year for windowing two digit date fields (for example, PIVOTYR=70 translates 69 to 2069 while 71 is translated to 1971).
7. **WARP** was added to support modifying data (numeric values) and dates in multiple different formats. Dates can be moved forward or backward or converted to different formats.

The STARWARP dialog (**WARP** primary command or options 5.7 and 5.8) was added to create a ISPF table of a copybook map for use with STARBAT setup for date manipulations. The **BUILD** command may be entered to build STARBAT JCL for submission to JES.

The **WARP** line command was added in MEMLIST to permit entering the STARWARP dialog easily.

See option **5.7** for copybook support for warping dates and option **5.8** for modifying a previously saved copybook table.

For STARWARP users, option **4** automatically switches to the **WARP** option for easy entry to the STARWARP dialog.

These copybook tables can be saved and then managed by the **PROFMAN** application. To activate a saved STARWARP table, use the PROFMAN application and position to the **SW** tables; enter an **S** command to activate the STARWARP dialog from the point the table was saved.

The **DATEDIFF** command was added to calculate the number of days between any two dates in any of the supported STARWARP formats.

The **DATECALC** command was added to warp a date in any supported STARWARP format and display the final day of the week and the date in any desired format.

## GO Sessions

When a GO session is used to select a copybook member as described above, the normal MEMLIST statistics outputs are suppressed.

The GO command was changed to add the **ADD** keyword which always creates a new GO session regardless of the current environment. If a GO session can not be created, an error message is issued.

When you have found the copybook that you want, enter **S** (select) and that member name will be substituted in the copybook member selection panel.

## LISTC/LISTF Tables

LC **REFRESH** now updates HSM migration level indicators in **DO** field and the **Device type** field if changed.

The **F** (find) and **EXCLUDE** commands were changed in LISTC/LISTF to permit searching "On VTOC" and "Cataloged" fields instead of SEC and EXTENT fields.

## WORKPAD, LISTC and LISTA

WORKPAD line commands for data sets will not change to a data set if it is already current; this keeps current functions active and SUBLISTs related to the current data set are maintained.

WP/LA/LC **Browse** and **Edit** line commands now support a single member without creating a MEMLIST.

The **ALT** line command for LISTA, LISTC/LISTF and WORKPAD supports dynamic line commands. You may define a dynamic line command using ALT as the first keyword and STARWARP will change to the data set and perform the following function.

For example, if you define MMM as ALT %CLISTX and you enter the MMM line command for a data set, STARWARP will change to the data set and execute CLISTX.

MMM can also be used in a primary command such as APPLY MMM. In this case, the MMM line command will be executed for all data sets in the table. This provides another way to perform global data set processing.

## HISTORY Subcommand

1. The **NOSYSTEM** keyword was added to filter out compiler routines (DFH, DFS, DSN, IBM, IEY, IGY, ILB, ISP or PLI) like the HISTORY GENERATE option currently does.
2. The **ALL** keyword was added to mean TRANS, USERDATA, ZAP and LKED. You may use this with another option to turn on all other options as in the following: HISTORY name NOTRANS ALL
3. The PDS25II and PDS260I messages were modified to add an **I** character (for **IMS**) after the two version characters if the load module contains CSECTS beginning with DFS.

## MAP Subcommand

The **NOSYSTEM** keyword was added to the MAP subcommand to filter out compiler routines (DFH, DFS, DSN, IBM, IEY, IGY, ILB, ISP or PLI) like the HISTORY GENERATE option currently does.

## REMOVE Command

The **REMOVE** command was added to support table exclusion processing for MEMLIST source and load member lists. This command is based on the MEMLIST F (or find) command.

The syntax for source member lists is as follows:

```
REMOVE string NOT PREFIX/SUFFIX/WORD  
MEMBER/VERMOD/CREATED/MODIFIED/SIZE/INIT/ID/TTR/ALIASOF/DATA/MSG
```

The syntax for load member lists is as follows:

```
REMOVE string NOT PREFIX/SUFFIX/WORD  
MEMBER/DATE/LENGTH/ENTRY/MAIN/MATCH/SSI/TTR/ALIASOF/DATA/MSG
```

## ISPF Support

- Option **3.2.D** supports deleting the current data set by switching to the profile data set if necessary
- Option **3.2.R** supports renaming the current data set by switching to the profile data set if necessary
- **SELCMD** was added to pass a command through ISPF "SELECT CMD".
- **ISPEXEC** is supported for direct invocation of ISPF service from STARWARP CLISTS.
- **DGETVAR PDSRC** was added to get the return code from ISPEXEC services.

## Miscellaneous

- **SEPARATE** was changed to support PDS/E 2.1.0 through 2.1.2 COMBINE formats.
- **COMBINE** was modified to output five character Julian dates as used by PDSTOOLS and several public domain programs like REVIEW, OFFLOAD, OFFLOADW and LISTPDS when the new COMPATIBLE keyword is used.
- **ATTRIB, IF** and **MEMLIST** were changed to support ISPF dates in the range of 1900 through 2899 for a year value. Previously, a date like 1940 would be assumed to be 2040 and a date like 2695 would be assumed to be 1965 because the date windowing technique utilized ignored any century digits.
- **LANGTYP** (option 10.17) was added to analyze source member types for members in an existing or new member list.
- A batch version of **COMPCHK** and **CSECTCHK** was added to in the STARWARP CNTL library as member COMPBAT. This version of these routines is not dependent on ISPF processing or tables and should be able to process very large load libraries.