Thenon/ONE

Compare and Merge Manager for the AS/400

for users of Thenon/SEE version 4.1

User and Reference Manual

Information in this document is subject to change without notice and does not represent a commitment on the part of Thenon Software Pty Limited.

Thenon Software Pty Limited provides no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

Thenon Software shall not be liable for errors contained herein or for any direct, incidental, or consequential damages resulting from the use of this material.

Contents

Introduction

What is Thenon/ONE

Thenon/ONE is a source code integrator tool for the IBM AS/400.

It allows you to compare any two source members, and identify the differences between them. It also allows you to merge any two different source members which have evolved from a common base version, produce a report showing the differences between the two versions and the base version, and automatically generate a composite source member.

Aims of Thenon/ONE

Thenon/ONE should:

- Enable you to determine the code changes in any particular source version against any previous version.
- Assist you in the integration of vendor changes to packaged software with your own in-house modifications, and significantly reduce the programming effort involved in the process.
- Assist you in managing the on-going software maintenance activities where an urgent fix is required for source code which is already being further developed for long term change (like a new release or a new version).
- Enable increased productivity by allowing more than one person to concurrently change the same source member.
- Enable increased development efficiency by allowing specialisation. The specialists can perform the tasks they do best on a timely basis; technologists can build the system interfaces; users can work with analysts on user interfaces; and normal daily maintenance can take place, all at the same time.

For whom is Thenon/ONE intended ?

Thenon/ONE is intended for use by Programmers and Project Leaders.

The product should also be of interest to Information Systems Managers who are involved in structuring and streamlining development activities.

Thenon/ONE is a generic source integrator, and can be used to compare and merge any two source members regardless of their actual content or statement syntax. Therefore it can also be used by technical writers who create and maintain text in AS/400 source files.

What this manual assumes you already know

It is assumed you have a working knowledge of the AS/400 and are familiar with it's operating system OS/400, and in particular the following features:

- libraries
- source files
- source members
- source types
- library lists and current library
- message handling
- jobs and job logs

Please refer to the appropriate OS/400 manuals for further information about the above topics.

Brief tour and basics

Thenon/ONE is command oriented.

The various menu options prompt for relevant command parameters. Once all parameters are specified, a batch submit window is shown, allowing you to execute the command either interactively or in batch. (For certain interactive-only functions the submit window is not shown).

All commands are provided with help text, which can be obtained in the normal way by pressing F1 or the HELP key.

Following are terms you should be familiar with before using the product.

Members for compare operations

For a compare operation you must specify two input members and one output member:

- **Base Version** which is the original source member against which the Changed Version member is analysed. You must specify the member name, source file name and library. The specified member must contain one or more records at run time. This is an input only member, and is not changed by the compare operation.

- **Changed Version** which is the source member containing the changes made to the Base Version. You must specify the source file name and library. The member name, if not changed, is defaulted to the base member name (*BASEMBR). The specified member must contain one or more records at run time. This is an input only member, and is not changed by the compare operation.

- **Compare Report** which is the report generated by the compare operation and is stored in a source file member. The source file name, if not changed, is defaulted to *CURLIB/CMPREPORT; the member name, if not changed, is defaulted to the base member name (*BASEMBR). If the report member or file do not exist at run time, they are created by this operation. If the report member exists, it is cleared by the compare operation before the new report is generated.

Members for merge operations

For a merge operation you must specify three input members and one or two output members:

- **Base Version** which is the original source member against which the 1st Version and 2nd Version members are analysed. You must specify the member name, source file name and library. The specified member must contain one or more records at run time. This is an input only member, and is not changed by the merge operation.

- **1st Version** which is the source member containing the first set of changes made to the Base Version. You must specify the source file name and library. The member name, if not changed, is defaulted to the base member name (*BASEMBR). The specified member must contain one or more records at run time. This is an input only member, and is not changed by the merge operation.

- **2nd Version** which is the source member containing the second set of changes made to the Base Version. You must specify the source file name and library. The member name, if not changed, is defaulted to the base member name (*BASEMBR). The specified member must contain one or more records at run time. This is an input only member, and is not changed by the merge operation.

- **Merge Report**which is the report generated by the merge operation and is stored in a source file member. The source file name, if not changed, is defaulted to *CURLIB/MRGREPORT; the member name, if not changed, is defaulted to the base member name (*BASEMBR). If the report member or file do not exist at run time, they are created by this operation. If the report member exists, it is cleared by the merge operation before the new report is generated.

- **Merge Composite** which is the composite source member generated by the merge operation, incorporating the changes to the Base Version found in the 1st Version and the 2nd Version source members. The source file name, if not changed, is defaulted to *CURLIB/MRGOUTPUT; the member name, if not changed, is defaulted to the base member name (*BASEMBR). The generation of the composite member is optional. If requested, and if the member or file do not exist at run time, they are created by this operation; you can specify whether existing member is replaced by the current merge operation.

Execution requests

Whenever you execute compare or merge operation(s), you can specify an Execution Request Name under which your compare or merge specifications are stored.

Each execution request can hold up to 999 'lines'. Each line is referred to as Execution Request Sequence.

By using this facility you can:

- repeat the same operation(s) at a later time without having to specify any parameters - all previously used parameters will be re-used

- access the compare or merge output members without having to specify their names or location

- create links between various source members which are being concurrently developed, and which will require a merge operation at a later time

- maintain a log of compare and merge activities

The default execution request name in all Thenon/ONE commands is *USER, which is substituted at run time with the current job user profile name. When used within Thenon/SEE, the default name CRxxxxxxxx is used instead, where 'xxxxxxxx' is the CR number from which the operation is requested.

Integration with Thenon/SEE

Thenon/ONE can be used as a stand alone product, or as an add-on product to Thenon's existing suite of products known as Software Environment Engineering (SEE).

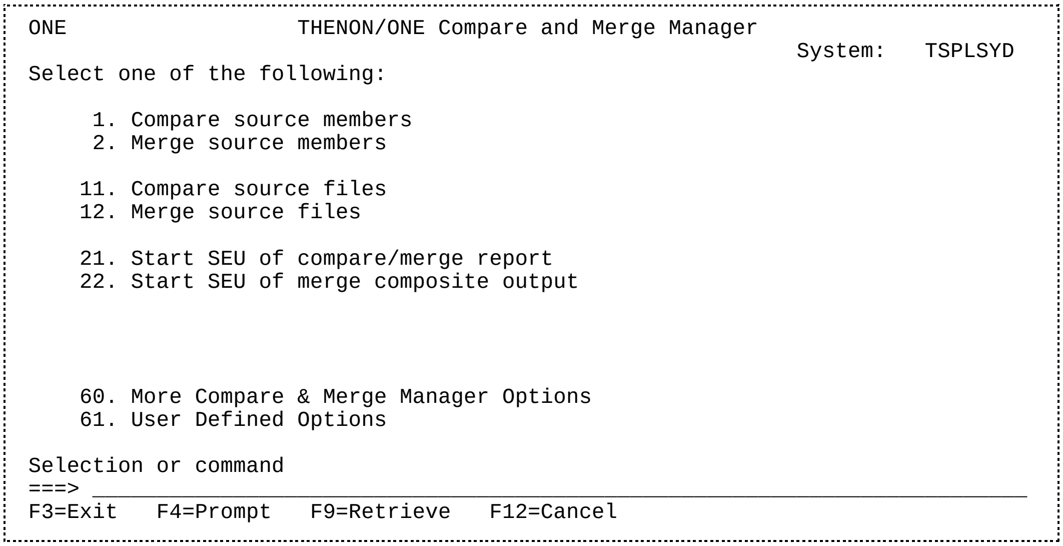
When used in conjunction with Thenon/SEE, most Thenon/ONE options are available from within the main development functions of Thenon/SEE. Refer to *Interface with Thenon/SEE* on page 51. You can also go directly to Thenon/ONE menus and select operations outside the context of Thenon/SEE.

Accessing Thenon/ONE in stand alone mode

In stand alone mode you can access Thenon/ONE menu by using command THENON from command entry. Type: `THENON MENU(ONE)`

Thenon/ONE menus

There are two menus associated with Thenon/ONE. Menu ONE is the primary menu and includes most commonly used options; menu ONE2 is the secondary menu. The following shows the menu options, and briefly describes each option:



```
ONE                  THENON/ONE Compare and Merge Manager
                                                            System:   TSPLSYD
Select one of the following:

     1. Compare source members
     2. Merge source members

    11. Compare source files
    12. Merge source files

    21. Start SEU of compare/merge report
    22. Start SEU of merge composite output




    60. More Compare & Merge Manager Options
    61. User Defined Options

Selection or command
===> _________________________________________________________________________
F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

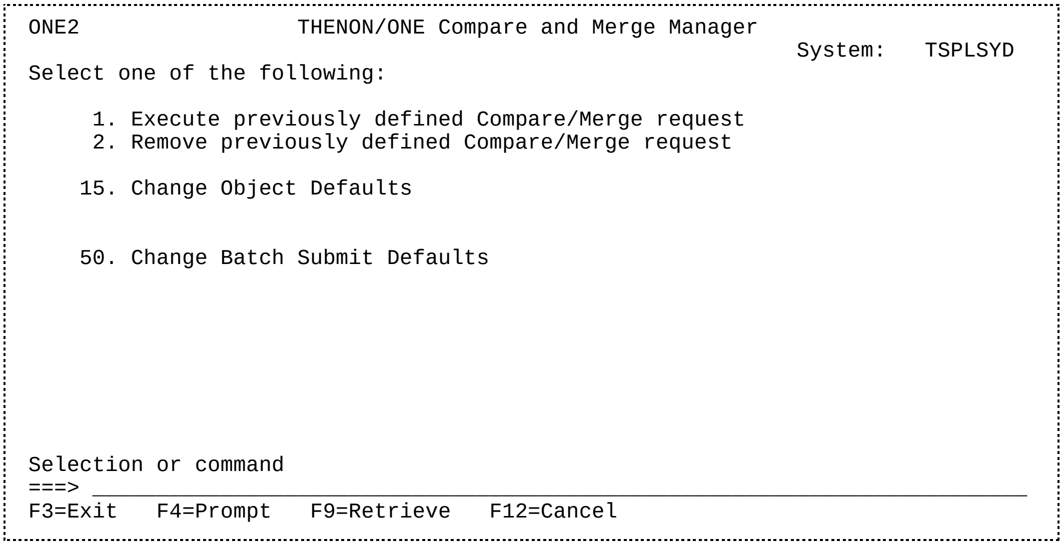
- Use **options 1 & 2** to execute a single ad-hoc compare or merge operation.

- Use **options 11 & 12** to execute bulk compare or merge operations, using all or a subset of members in the source files.

- Use **options 21 & 22** to access report and composite members which were generated by previous compare or merge operations.

- Use **option 60** to access the secondary menu.

- Use **option 61** to access a user-defined AS/400 menu. You must first have created menu name USRONE and included it in your library list.



```
ONE2                 THENON/ONE Compare and Merge Manager
                                                            System:   TSPLSYD
Select one of the following:

     1. Execute previously defined Compare/Merge request
     2. Remove previously defined Compare/Merge request

    15. Change Object Defaults


    50. Change Batch Submit Defaults








Selection or command
===> _________________________________________________________________________
F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

- Use **option 1** to repeat one or more compare and/or merge operations which were previously executed.

- Use **option 2** to remove the specifications of one or more compare and/or merge operations which were previously executed. Once you remove the specifications you cannot re-execute them.

- Use **option 15** to alter the product's internal defaults which are permanently stored for each object type.

- Use **option 50** to alter the job description name and attributes used for submitted jobs.

Comparing source members

You can compare any two source members, or execute the compare operation in bulk, using generic member names across whole source files.

Specifying and executing a simple compare operation

Use **option 1** from menu ONE, or prompt command CMPSRCMBR.

You have to specify:

- Base Version member, file and library names
- Changed Version member, file and library names
- Execution request name and sequence
- Execution request text

You can use F10 to prompt for additional (optional) parameters:

- Report member, file and library names
- Report format
- Whether blank compression is used
- Column delimiters for comparison
- The prefix and suffix character strings which are attached to the warning comments inserted in the report

For a detailed description of all parameters, refer to *Command CMPSRCMBR* on page 71.

If the compare operation completed normally, message id MRG0004 will indicate the name and location of the generated report source member.

Specifying and executing a bulk compare operation

Use **option 11** from menu ONE, or prompt command CMPSRCF.

The bulk compare allows you to execute a series of individual compare operations. All compares are logged under the same execution request name, and each compare is recorded under the execution request name with a unique sequence number.

You have to specify:

- A single member name, *ALL for all members, or *generic* for a subset of member names. The command scans all members in both input source files and executes a compare operation for each name which is included in the subset and is found in both Base Version source file and Changed Version source file
- Base Version source file and library names
- Changed Version source file and library names
- The execution request name and sequence number for the first compare operation; subsequent compare operations within the subset automatically increment the sequence number by 1
- The execution request text

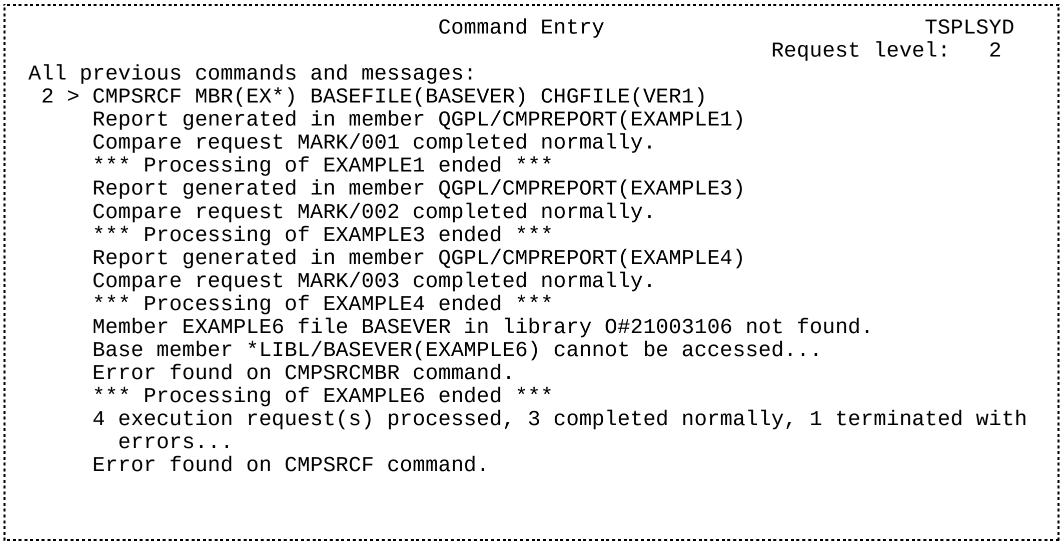
You can use F10 to prompt for additional (optional) parameters:

- Report file and library names; report member is created for each compare operation using the base version member name in the source file specified here
- Report format
- Whether blank compression is used
- Column delimiters for comparison
- The prefix and suffix character strings which are attached to the warning comments inserted in the report

For a detailed description of all parameters, refer to *Command CMPSRCF* on page 65.

To assist you in analysing the job log and determining which of the specified members exist only in the Base or Changed source file, but not in both, an information message is inserted in the job log after processing has completed for each member, thereby improving readability of the job log spool output. If the member does not exist in either the Base or Changed source files, you will see diagnostic message MRG0015 indicating the missing member, followed by an abnormal termination message for that member name.

After all members have been processed, diagnostic message MRG0110 will indicate the number of members which have been processed, the number of compare operations which have completed successfully and the number that terminated with errors.



```
                                Command Entry                         TSPLSYD
                                                          Request level:   2
All previous commands and messages:
 2 > CMPSRCF MBR(EX*) BASEFILE(BASEVER) CHGFILE(VER1)
     Report generated in member QGPL/CMPREPORT(EXAMPLE1)
     Compare request MARK/001 completed normally.
     *** Processing of EXAMPLE1 ended ***
     Report generated in member QGPL/CMPREPORT(EXAMPLE3)
     Compare request MARK/002 completed normally.
     *** Processing of EXAMPLE3 ended ***
     Report generated in member QGPL/CMPREPORT(EXAMPLE4)
     Compare request MARK/003 completed normally.
     *** Processing of EXAMPLE4 ended ***
     Member EXAMPLE6 file BASEVER in library O#21003106 not found.
     Base member *LIBL/BASEVER(EXAMPLE6) cannot be accessed...
     Error found on CMPSRCMBR command.
     *** Processing of EXAMPLE6 ended ***
     4 execution request(s) processed, 3 completed normally, 1 terminated with
       errors...
     Error found on CMPSRCF command.
```

Accessing the generated compare report

Regardless of whether you have executed a single compare or a bulk compare operation, you can access each of the generated report members by identifying the relevant operation under the execution request name, and request to display or edit the report member associated with the operation sequence number.

Use **option 21** from menu ONE, or prompt command STRMRGSEU.

You have to specify the execution request name you have used for the compare operation:

- The default is *USER
- You can use F4 to show a list of all existing execution request names found in the system

Once you have specified the execution request name, you have to select the required sequence number:

- The default is *LAST, i.e; the last sequence in the specified execution request name is used.
- You can use F4 to show a list of all existing sequence numbers under the specified execution request name, and select the sequence number you want. The list shows the name of the Base Version members used against each sequence number. Since the same request can contain both compare and merge operation logs, the Base Version member name is prefixed with C: to designate the sequence as a compare log, or M: to designate the sequence as a merge log.
- You can also specify keyword *SEARCH, to search for the last report generated for a compare (or merge) operation involving a specific member name. Then you have to specify the member name to search.

The SEU function being evoked is defaulted to the browse display. You can select the edit display or print functions instead.

For a detailed description of all parameters, refer to *Command STRMRGSEU* on page 97.

Interpreting the compare report

The compare report is organised as follows (refer to *Appendix C* on page 103 for a report sample):

Report Headers

The headers show the details of the compare operation, including:

- The execution request name, sequence and text
- The job (number/user/name) of the compare operation
- The Base Version member (library/file(member))
- The Changed Version member (library/file(member))
- The report type (*FULL or *SUMM)
- Whether blank compression was used
- The column delimiters used

Report detail lines

Following the headers, the detail lines showing the actual source statements appear in 3 distinct areas:

Area 1: Statement Line Numbers

Two columns of 6 digit line numbers. The first column shows the line number of the Base Version; the second column shows the line number of the Changed Version. These allow you to associate each source statement with its respective line number in the Base and Changed versions. If the line number does not appear under one of the columns, it indicates that the associated source statement is not included in the respective version, i.e; it was either deleted or inserted in the Changed Version.

Area 2: Action Codes

A column showing the action effected in the Changed Version on the associated source statement. The program identifies three main types of activities:

- **Delete (DEL)**
- **Insert (INS)**
- **Move (MOV)**

Therefore:

- a statement which has been changed is identified as a delete of the original statement which is flagged with action DEL, and an insert of a new statement which is flagged with action INS.

- a statement which has been copied from another location in the member is identified as an insert of a new statement which is flagged with action INS.

- a non-unique statement (or a group of non-unique statements) which has been moved from one location to another in the member is identified as a delete of the original statement which is flagged with action DEL, and an insert of a new statement which is flagged with INS.

- a unique statement (or a group of unique statements) which have been moved from one location to another in the member is identified as a move action. The statement flagged with action MOV is the originating statement, and identifies the original location. The statement flagged with action TO is the current statement, and identifies the location where the line was moved-to.

Possible action codes are:

INS This statement was inserted in the Changed Version.

DEL This statement (of the Base Version) was deleted in the Changed Version.

MOV This statement was moved to another location in the Changed Version. The Changed Version statement number (2nd column in Area 1) indicates the current location in the Changed Version. Every MOV can be coupled with a TO action code elsewhere using the statement line number.

TO This statement was moved to this location in the Changed Version. The Base Version statement number (1st column in Area 1) indicates the original location in the Base Version. Every TO can be coupled with a MOV action code elsewhere using the statement line number.

Area 3: Source Statements

The rest of the detail line shows the source statement itself.

Merging source members

You can merge any two source members including different sets of changes which have evolved from a common base version. You can also execute the merge operation in bulk, using generic member names across whole source files.

Specifying and executing a simple merge operation

Use **option 2** from menu ONE, or prompt command MRGSRCMBR.

You have to specify:

- Base Version member, file and library names
- The 1st Version member, file and library names
- The 2nd Version member, file and library names
- The execution request name and sequence
- The execution request text

You can use F10 to prompt for additional (optional) parameters:

- Report member, file and library names
- Composite member, file and library names; you can execute a merge operation without generating a composite source, by specifying *NONE for the composite source file name
- Whether the generated composite member is allowed to replace existing member
- Report format
- Whether blank compression is used
- Column delimiters for comparison
- The prefix and suffix character strings which attached to the warning comments inserted in the report and composite members

For a detailed description of all parameters, refer to *Command MRGSRCMBR* on page 87.

If the merge operation completed normally, message id MRG0004 will indicate the name and location of the generated report source member, and message id MRG0005 will indicate the name and location of the generated composite source member.

Specifying and executing a bulk merge operation

Use **option 12** from menu ONE, or prompt command MRGSRCF.

The bulk merge allows you to execute a series of individual merge operations. All operations are logged under the same execution request name, and each merge operation is recorded under the execution request name with a unique sequence number.

You have to specify:

- A single member name, *ALL for all members, or *generic* for a subset of member names. The command scans all members in all input source files and executes a merge operation for each name which is included in the subset and is found in the Base Version, 1st Version and 2nd Version
- Base Version source file and library names
- 1st Version source file and library names
- 2nd Version source file and library names
- The execution request name and sequence number for the first merge operation; subsequent merge operations within the subset automatically increment the sequence number by 1
- The execution request text

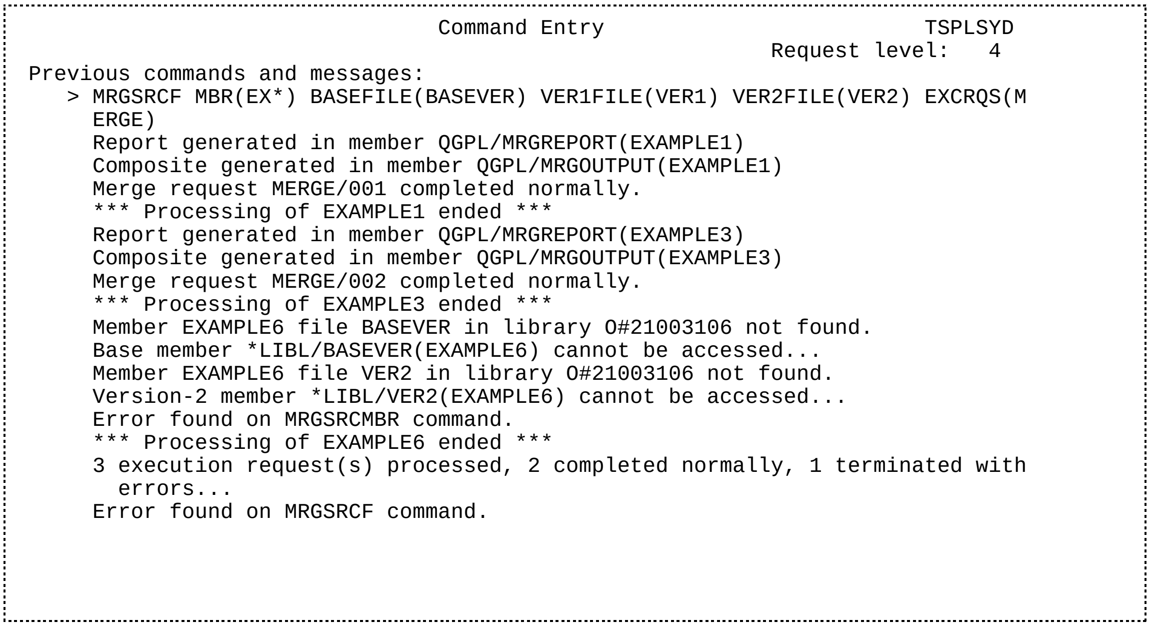
You can use F10 to prompt for additional (optional) parameters:

- Report file and library names; report member is created for each merge operation using the base version member name in the source file specified here
- Composite file and library names; composite member is created for each merge operation using the base version member name in the source file specified here. You can execute a merge operation without generating a composite source, by specifying *NONE for the composite source file name
- Whether each of the generated composite members is allowed to replace existing member
- Report format
- Whether blank compression is used
- Column delimiters for comparison
- The prefix and suffix character strings which are attached to the warning comments inserted in the report and composite members

For a detailed description of all parameters, refer to *Command MRGSRCF* on page 79.

To assist you in analysing the job log and determining which of the specified members exist only in one, but not in all source files, an information message is inserted in the job log after processing has completed for each member, thereby improving readability of the job log spool output. If the member does not exist in either the Base, 1st Version or 2nd Version source files, you will see diagnostic message MRG0015 indicating the missing member, followed by an abnormal termination message for that member name.

After all members have been processed, diagnostic message MRG0110 will indicate the number of members which have been processed, the number of merge operations which have completed successfully and the number that terminated with errors.



```
                                Command Entry                         TSPLSYD
                                                          Request level:   4
Previous commands and messages:
   > MRGSRCF MBR(EX*) BASEFILE(BASEVER) VER1FILE(VER1) VER2FILE(VER2) EXCRQS(M
     ERGE)
     Report generated in member QGPL/MRGREPORT(EXAMPLE1)
     Composite generated in member QGPL/MRGOUTPUT(EXAMPLE1)
     Merge request MERGE/001 completed normally.
     *** Processing of EXAMPLE1 ended ***
     Report generated in member QGPL/MRGREPORT(EXAMPLE3)
     Composite generated in member QGPL/MRGOUTPUT(EXAMPLE3)
     Merge request MERGE/002 completed normally.
     *** Processing of EXAMPLE3 ended ***
     Member EXAMPLE6 file BASEVER in library O#21003106 not found.
     Base member *LIBL/BASEVER(EXAMPLE6) cannot be accessed...
     Member EXAMPLE6 file VER2 in library O#21003106 not found.
     Version-2 member *LIBL/VER2(EXAMPLE6) cannot be accessed...
     Error found on MRGSRCMBR command.
     *** Processing of EXAMPLE6 ended ***
     3 execution request(s) processed, 2 completed normally, 1 terminated with
       errors...
     Error found on MRGSRCF command.
```

Accessing the generated merge report and composite members

Regardless of whether you have executed a single merge or a bulk merge operation, you can access each of the generated report and composite members by identifying the relevant operation under the execution request name, and request to display or edit the report or composite member associated with the operation sequence number.

Use **option 21** from menu ONE for the report member, or **option 22** for the composite member, or prompt command STRMRGSEU which can be used to access both report and composite.

You have to specify the execution request name you have used for the merge operation:

- The default is *USER
- You can use F4 to show a list of all existing execution request names found in the system

Once you have specified the execution request name, you have to select the required sequence number:

- The default is *LAST, i.e; the last sequence in the specified execution request name is used
- You can use F4 to show a list of all existing sequence numbers under the specified execution request name, and select the sequence number you want. The list shows against each sequence number, the name of the Base Version member used. Since the same request can contain both compare and merge operation logs, the Base Version member name is prefixed with C: to designate the sequence as a compare log, or M: to designate the sequence as a merge log.
- You can also specify keyword *SEARCH, to search for the last report or composite members which were generated for an operation involving a specific member name. Then you have to specify the member name to search.

The SEU function being evoked is defaulted to the browse display for the report, and the edit display for the composite member. You can select the browse, edit or print functions instead.

For a detailed description of all parameters, refer to *Command STRMRGSEU* on page 97.

The merge report

The merge report is organised as follows (refer to *Appendix D* on page 113 for a report sample):

Report Headers

The headers show the details of the merge operation, and include:

- The execution request name, sequence and text
- The job (number/user/name) of the compare operation
- The Base Version member (library/file(member))
- The 1st Version member (library/file(member))
- The 2nd Version member (library/file(member))
- The Composite member (library/file(member))
- Whether the Composite member replaces existing member
- The report type (*FULL or *SUMM)
- Whether blank compression was used
- The column delimiters used

Report detail lines

Following the headers, the detail lines showing the actual source statements appear in 3 distinct areas:

Area 1: Statement Line Numbers

Three columns of 6 digit line numbers. The first column shows the line number of the Base Version; the second column shows the line number of the 1st Version; the third column shows the line number of the 2nd Version. These allow you to associate each source statement with it's respective line number in the Base, 1st and 2nd versions. If the line number does not appear under one of the columns, it indicates that the associated source statement is not included in the respective version, i.e; it was either deleted or inserted by 1st and/or 2nd Versions.

Area 2: Action Codes

Two columns showing the action codes. The first column shows actions codes effected on the associated source statement in the 1st version in comparison to the base version; the second column shows action codes effected in the 2nd version in comparison to the base version. The program identifies three main types of activities:

- **Delete (DEL)**
- **Insert (INS)**
- **Move (MOV)**

Therefore:

- a statement which has been changed is identified as a delete of the original statement which is flagged with action DEL, and an insert of a new statement which is flagged with action INS.

- a statement which has been copied from another location in the member is identified as an insert of a new statement which is flagged with action INS.

- a non-unique statement (or a group of non-unique statements) which has been moved from one location to another in the member is identified as a delete of the original statement which is flagged with action DEL, and an insert of a new statement which is flagged with INS.

- a unique statement (or a group of unique statements) which have been moved from one location to another in the member is identified as a move action. The statement flagged with action MOV is the originating statement, and identifies the original location. The statement flagged with action TO is the current statement, and identifies the location where the line was moved-to.

Area 3: Source Statements

The rest of the detail line shows the source statement itself.

Activities and action codes

The action codes which appear on the report depend on the type of activity effected in both changed versions. Activities are internally classified as either:

- Simple
- Intermediate
- Complex

The following pages show examples of the possible action codes associated with various activities. Below is the base version used in all these examples:

Base version (for all activity examples)

```
0001.00      C           @PRMC     IFEQ '$DA1'
0002.00      C                     MOVE @SSRCF    LPMV03
0003.00      C                     MOVE @SOMSN    LPMV07
0004.00      C                     MOVE @SXXX1    LPMV04
0005.00      C                     MOVE @SXXX2    LPMV05
0006.00      C                     MOVE @SELTY    LPMV08
0007.00      C                     MOVE @SELAT    LPMV09
0008.00      C                     MOVE @SELNM    LPMV01
0009.00      C                     MOVE *BLANKS   LPMV16
0010.00      C                     MOVEL@SSRCT    LPMV16
0011.00      C                     MOVE *BLANKS   LPMV17
0012.00      C                     MOVEL@SSRCQ    LPMV17
0013.00      C                     END
0014.00       **
0015.00      C                     MOVE IFMSGE    LPMSGE
0016.00      C                     MOVE IFSBMT    LPSBMT
0017.00      C                     MOVE @SELNM    LPSBMN
0018.00      C           @ITMFL    IFEQ 'N'
0019.00      C                     MOVE IFSBMN    LPSBMN
0020.00      C                     END
```

Example of simple activities

- **A statement was not affected by either version**

Actions		
V1	V2	
===	===	No action was effected on this statement in either 1st or 2nd version.

- **A statement is deleted in both versions**

Actions		
V1	V2	
DEL	DEL	This same statement was deleted in both 1st and 2nd versions.

- **A statement is inserted at the same location in both versions**

Actions		
V1	V2	
INS	INS	This same statement was inserted in both 1st and 2nd versions.

- **A statement is deleted in one version, and not affected in the other**

Actions		
V1	V2	
DEL		This statement was deleted in the 1st version.
	DEL	This statement was deleted in the 2nd version.

- **A statement is inserted in one version, and not affected in the other**

Actions		
V1	V2	
INS		This statement was inserted in the 1st version.
	INS	This statement was inserted in the 2nd version.

1st version - simple activities

```
0001.00      C           @PRMC     IFEQ '$DA1'
0002.00      C                     MOVE @SSRCF    LPMV03
0003.00      C                     MOVE @SXXX2    LPMV05
0004.00      C                     MOVE @SELTY    LPMV08
0005.00      C                     MOVE @SELAT    LPMV09
0006.00      C                     MOVE @SELNM    LPMV01
0007.00      C                     MOVE *BLANKS   LPMV16
0008.00      C                     MOVEL@SSRCT    LPMV16
0009.00      C                     MOVE *BLANKS   LPMV17
0010.00      C                     MOVE *BLANKS   LPMV18
0011.00      C                     MOVE *BLANKS   LPMV19
0012.00      C                     MOVE *BLANKS   LPMV20
0013.00      C                     MOVEL@SSRCQ    LPMV17
0014.00      C                     END
0015.00       **
0016.00      C                     MOVE IFMSGE    LPMSGE
0017.00      C                     MOVE IFSBMT    LPSBMT
0018.00      C                     MOVE @SELNM    LPSBMN
0019.00      C           @ITMFL    IFEQ 'N'
0020.00      C                     MOVE IFSBMX    LPSBMN
0021.00      C                     END
```

2nd version - simple activities

```
0001.00      C           @PRMC     IFEQ '$DA1'
0002.00      C                     MOVE @SSRCF    LPMV03
0003.00      C                     MOVE @SXXX1    LPMV04
0004.00      C                     MOVE @SXXX2    LPMV05
0005.00      C                     MOVE @SELTY    LPMV08
0006.00      C                     MOVE @SELAT    LPMV09
0007.00      C                     MOVE @SELNM    LPMV01
0008.00      C                     MOVE *BLANKS   LPMV16
0009.00      C                     MOVEL@SSRCT    LPMV16
0010.00      C                     MOVE *BLANKS   LPMV17
0011.00      C                     MOVE *BLANKS   LPMV18
0012.00      C                     MOVE *BLANKS   LPMV19
0013.00      C                     MOVE *BLANKS   LPMV20
0014.00      C                     MOVEL@SSRCQ    LPMV17
0015.00      C                     END
0016.00       **
0017.00      C                     MOVE IFMSGE    LPMSGE
0018.00      C                     MOVE IFSBMT    LPSBMT
0019.00      C                     MOVE @SELNM    LPSBMN
0020.00      C           @ITMFL    IFEQ 'N'
0021.00      C                     MOVE IFSBMN    LPSBMX
0022.00      C                     END
```

Merge report - simple activities

```
Base   Ver-1  Ver-2  Actions .................Source statement..................
000100 000100 000100 === ===       C           @PRMC     IFEQ '$DA1'
000200 000200 000200 === ===       C                     MOVE @SSRCF    LPMV03
000300               DEL DEL       C                     MOVE @SOMSN    LPMV07
000400        000300 DEL           C                     MOVE @SXXX1    LPMV04
000500 000300 000400 === ===       C                     MOVE @SXXX2    LPMV05
000600 000400 000500 === ===       C                     MOVE @SELTY    LPMV08
000700 000500 000600 === ===       C                     MOVE @SELAT    LPMV09
000800 000600 000700 === ===       C                     MOVE @SELNM    LPMV01
000900 000700 000800 === ===       C                     MOVE *BLANKS   LPMV16
001000 000800 000900 === ===       C                     MOVEL@SSRCT    LPMV16
001100 000900 001000 === ===       C                     MOVE *BLANKS   LPMV17
       001000 001100 INS INS       C                     MOVE *BLANKS   LPMV18
       001100 001200 INS INS       C                     MOVE *BLANKS   LPMV19
       001200 001300 INS INS       C                     MOVE *BLANKS   LPMV20
001200 001300 001400 === ===       C                     MOVEL@SSRCQ    LPMV17
001300 001400 001500 === ===       C                     END
001400 001500 001600 === ===        **
001500 001600 001700 === ===       C                     MOVE IFMSGE    LPMSGE
001600 001700 001800 === ===       C                     MOVE IFSBMT    LPSBMT
001700 001800 001900 === ===       C                     MOVE @SELNM    LPSBMN
001800 001900 002000 === ===       C           @ITMFL    IFEQ 'N'
                                   /* ! Review Required: Begin !*/
001900               DEL DEL *     C                     MOVE IFSBMN    LPSBMN
       002000        INS     *     C                     MOVE IFSBMX    LPSBMN
              002100     INS *     C                     MOVE IFSBMN    LPSBMX
                                   /* ! Review Required: End   !*/
002000 002100 002200 === ===       C                     END
```

Composite output - simple activities

```
0000.01      C           @PRMC     IFEQ '$DA1'
0000.02      C                     MOVE @SSRCF    LPMV03
0000.03      C                     MOVE @SXXX2    LPMV05
0000.04      C                     MOVE @SELTY    LPMV08
0000.05      C                     MOVE @SELAT    LPMV09
0000.06      C                     MOVE @SELNM    LPMV01
0000.07      C                     MOVE *BLANKS   LPMV16
0000.08      C                     MOVEL@SSRCT    LPMV16
0000.09      C                     MOVE *BLANKS   LPMV17
0000.10      C                     MOVE *BLANKS   LPMV18
0000.11      C                     MOVE *BLANKS   LPMV19
0000.12      C                     MOVE *BLANKS   LPMV20
0000.13      C                     MOVEL@SSRCQ    LPMV17
0000.14      C                     END
0000.15       **
0000.16      C                     MOVE IFMSGE    LPMSGE
0000.17      C                     MOVE IFSBMT    LPSBMT
0000.18      C                     MOVE @SELNM    LPSBMN
0000.19      C           @ITMFL    IFEQ 'N'
0000.20      /* ! Review Required: Begin !*/
0000.21      C                     MOVE IFSBMN    LPSBMN
0000.22      C                     MOVE IFSBMX    LPSBMN
0000.23      C                     MOVE IFSBMN    LPSBMX
0000.24      /* ! Review Required: End   !*/
0000.25      C                     END
```

Interpreting the report and composite member

The following summarises the relationship between what appears in the report and what is included in the composite output:

Deletes

- When a statement is deleted in the 1st version, and no changes are effected on the statement in the 2nd version, the deleted statement will be flagged with action DEL on the report and will not be included in the composite member.
- When a statement is deleted in the 2nd version, and no changes are effected on the statement in the 1st version, the deleted statement will be flagged with action DEL on the report and will not be included in the composite member.
- When a statement is deleted in both the 1st and 2nd versions, the deleted statement will be flagged with action DEL on the report and will not be included in the composite member.

Inserts

- When a statement is inserted in the 1st version, and no changes are effected in the same location in the 2nd version, the inserted line will be flagged with action INS on the report and will be included in the composite member.
- When a statement is inserted in the 2nd version, and no changes are effected in the same location in the 1st version, the inserted line will be flagged with action INS on the report and will be included in the composite member.
- When the same statement is inserted in the same location in both the 1st and 2nd versions, the inserted line will be flagged with action INS on the report and will be included in the composite member.
- When a statement is inserted in one version and a different statement is inserted at the same location in the other version, the inserted statements are identified as a conflict and will appear between **Review Required** comment lines in both the report and composite member.

  Typically, this is the result when different modifications are affected in the two versions on the same base version statement, i.e; both versions have deleted the existing statement and have replaced it with different statements.

Example of intermediate activities

- **A statement was moved to the same location in both versions**

Actions
V1 V2

MOV MOV This statement was moved to another location in both the 1st and 2nd versions. In both versions the statement was moved in the same relative direction, i.e; either downwards or upwards in the source member.

TO TO The statement was moved to this location in both the 1st and 2nd versions.

- **A statement was moved in one version, and not affected in the other**

Actions
V1 V2

MOV This statement was moved to another location in the 1st version.

TO The statement was moved to this location in the 1st version.

MOV This statement was moved to another location in the 2nd version.

TO The statement was moved to this location in the 2nd version.

1st version - intermediate activities

```
0001.00      C           @PRMC     IFEQ '$DA1'
0002.00      C                     MOVE @SXXX1    LPMV04
0003.00      C                     MOVE @SXXX2    LPMV05
0004.00      C                     MOVE @SELTY    LPMV08
0005.00      C                     MOVE @SELAT    LPMV09
0006.00      C                     MOVE @SELNM    LPMV01
0007.00      C                     MOVE @SSRCF    LPMV03
0008.00      C                     MOVE @SOMSN    LPMV07
0009.00      C                     MOVE *BLANKS   LPMV16
0010.00      C                     MOVEL@SSRCT    LPMV16
0011.00      C                     MOVE *BLANKS   LPMV17
0012.00      C                     MOVEL@SSRCQ    LPMV17
0013.00      C                     END
0014.00       **
0015.00      C                     MOVE @SELNM    LPSBMN
0016.00      C           @ITMFL    IFEQ 'N'
0017.00      C                     MOVE IFSBMN    LPSBMN
0018.00      C                     MOVE IFMSGE    LPMSGE
0019.00      C                     MOVE IFSBMT    LPSBMT
0020.00      C                     END
```

2nd version - intermediate activities

```
0001.00      C           @PRMC     IFEQ '$DA1'
0002.00      C                     MOVE @SXXX1    LPMV04
0003.00      C                     MOVE @SXXX2    LPMV05
0004.00      C                     MOVE @SELTY    LPMV08
0005.00      C                     MOVE @SELAT    LPMV09
0006.00      C                     MOVE @SELNM    LPMV01
0007.00      C                     MOVE @SSRCF    LPMV03
0008.00      C                     MOVE @SOMSN    LPMV07
0009.00      C                     MOVE *BLANKS   LPMV17
0010.00      C                     MOVEL@SSRCQ    LPMV17
0011.00      C                     MOVE *BLANKS   LPMV16
0012.00      C                     MOVEL@SSRCT    LPMV16
0013.00      C                     END
0014.00       **
0015.00      C                     MOVE IFMSGE    LPMSGE
0016.00      C                     MOVE IFSBMT    LPSBMT
0017.00      C                     MOVE @SELNM    LPSBMN
0018.00      C           @ITMFL    IFEQ 'N'
0019.00      C                     MOVE IFSBMN    LPSBMN
0020.00      C                     END
```

Merge report - intermediate activities

```
Base   Ver-1  Ver-2  Actions .................Source statement..................
000100 000100 000100 === ===        C           @PRMC     IFEQ '$DA1'
000200 000700 000700 MOV MOV        C                     MOVE @SSRCF    LPMV03
000300 000800 000800 MOV MOV        C                     MOVE @SOMSN    LPMV07
000400 000200 000200 === ===        C                     MOVE @SXXX1    LPMV04
000500 000300 000300 === ===        C                     MOVE @SXXX2    LPMV05
000600 000400 000400 === ===        C                     MOVE @SELTY    LPMV08
000700 000500 000500 === ===        C                     MOVE @SELAT    LPMV09
000800 000600 000600 === ===        C                     MOVE @SELNM    LPMV01
000200 000700 000700 TO  TO         C                     MOVE @SSRCF    LPMV03
000300 000800 000800 TO  TO         C                     MOVE @SOMSN    LPMV07
000900 000900 001100     MOV        C                     MOVE *BLANKS   LPMV16
001000 001000 001200     MOV        C                     MOVEL@SSRCT    LPMV16
001100 001100 000900 === ===        C                     MOVE *BLANKS   LPMV17
001200 001200 001000 === ===        C                     MOVEL@SSRCQ    LPMV17
000900 000900 001100     TO         C                     MOVE *BLANKS   LPMV16
001000 001000 001200     TO         C                     MOVEL@SSRCT    LPMV16
001300 001300 001300 === ===        C                     END
001400 001400 001400 === ===         **
001500 001800 001500 MOV            C                     MOVE IFMSGE    LPMSGE
001600 001900 001600 MOV            C                     MOVE IFSBMT    LPSBMT
001700 001500 001700 === ===        C                     MOVE @SELNM    LPSBMN
001800 001600 001800 === ===        C           @ITMFL    IFEQ 'N'
001900 001700 001900 === ===        C                     MOVE IFSBMN    LPSBMN
001500 001800 001500 TO             C                     MOVE IFMSGE    LPMSGE
001600 001900 001600 TO             C                     MOVE IFSBMT    LPSBMT
002000 002000 002000 === ===        C                     END
```

Composite output - intermediate activities

```
0000.01      C           @PRMC     IFEQ '$DA1'
0000.02      C                     MOVE @SXXX1    LPMV04
0000.03      C                     MOVE @SXXX2    LPMV05
0000.04      C                     MOVE @SELTY    LPMV08
0000.05      C                     MOVE @SELAT    LPMV09
0000.06      C                     MOVE @SELNM    LPMV01
0000.07      C                     MOVE @SSRCF    LPMV03
0000.08      C                     MOVE @SOMSN    LPMV07
0000.09      C                     MOVE *BLANKS   LPMV17
0000.10      C                     MOVEL@SSRCQ    LPMV17
0000.11      C                     MOVE *BLANKS   LPMV16
0000.12      C                     MOVEL@SSRCT    LPMV16
0000.13      C                     END
0000.14       **
0000.15      C                     MOVE @SELNM    LPSBMN
0000.16      C           @ITMFL    IFEQ 'N'
0000.17      C                     MOVE IFSBMN    LPSBMN
0000.18      C                     MOVE IFMSGE    LPMSGE
0000.19      C                     MOVE IFSBMT    LPSBMT
0000.20      C                     END
```

Interpreting the report and composite member

The following summarises the relationship between what appears in the report and what is included in the composite output:

- When a statement is moved in the 1st version, and no changes are effected on the statement in the 2nd version, the statement in the original location will be flagged with action MOV on the report and will not be included in the composite member; the statement in the target location will be flagged with action TO and will be included in the composite member.

- When a statement is moved in the 2nd version, and no changes are effected on the statement in the 1st version, the statement in the original location will be flagged with action MOV on the report and will not be included in the composite member; the statement in the target location will be flagged with action TO and will be included in the composite member.

- When a statement is moved in both versions to the same target location, the statement in the original location will be flagged with action MOV on the report and will not be included in the composite member; the statement in the target location will be flagged with action TO and will be included in the composite member.

- Note that a MOV is always matched with a TO in the same activity column. You can associate a MOV with a TO by matching the statement line number in the appropriate column in area 1.

Example of complex activities type 1

- **A statement was moved to another location in one version, and deleted in the other version**

Actions		
V1	V2	
DEL	MOV	This statement was moved to another location in the 2nd version, and deleted in the 1st version.
DEL	TO	The statement was moved to this location in the 2nd version, and deleted in the 1st version.
MOV	DEL	This statement was moved to another location in the 1st version, and deleted in the 2nd version.
TO	DEL	The statement was moved to this location in the 1st version, and deleted in the 2nd version.

1st version - complex activities type 1

```
0001.00      C           @PRMC     IFEQ '$DA1'
0002.00      C                     MOVE @SXXX1    LPMV04
0003.00      C                     MOVE @SXXX2    LPMV05
0004.00      C                     MOVE @SELTY    LPMV08
0005.00      C                     MOVE @SELAT    LPMV09
0006.00      C                     MOVE @SELNM    LPMV01
0007.00      C                     MOVE *BLANKS   LPMV16
0008.00      C                     MOVEL@SSRCT    LPMV16
0009.00      C                     MOVE *BLANKS   LPMV17
0010.00      C                     MOVEL@SSRCQ    LPMV17
0011.00      C                     END
0012.00       **
0013.00      C                     MOVE IFSBMT    LPSBMT
0014.00      C                     MOVE @SELNM    LPSBMN
0015.00      C           @ITMFL    IFEQ 'N'
0016.00      C                     MOVE IFSBMN    LPSBMN
0017.00      C                     MOVE IFMSGE    LPMSGE
0018.00      C                     END
```

2nd version - complex activities type 1

```
0001.00      C           @PRMC     IFEQ '$DA1'
0002.00      C                     MOVE @SXXX1    LPMV04
0003.00      C                     MOVE @SXXX2    LPMV05
0004.00      C                     MOVE @SELTY    LPMV08
0005.00      C                     MOVE @SELAT    LPMV09
0006.00      C                     MOVE @SELNM    LPMV01
0007.00      C                     MOVE *BLANKS   LPMV16
0008.00      C                     MOVE @SSRCF    LPMV03
0009.00      C                     MOVE @SOMSN    LPMV07
0010.00      C                     MOVEL@SSRCT    LPMV16
0011.00      C                     MOVE *BLANKS   LPMV17
0012.00      C                     MOVEL@SSRCQ    LPMV17
0013.00      C                     END
0014.00       **
0015.00      C                     MOVE IFSBMT    LPSBMT
0016.00      C                     MOVE @SELNM    LPSBMN
0017.00      C           @ITMFL    IFEQ 'N'
0018.00      C                     MOVE IFSBMN    LPSBMN
0019.00      C                     END
```

Merge report - complex activities type 1

```
Base   Ver-1  Ver-2  Actions .................Source statement..................
000100 000100 000100 === ===        C           @PRMC     IFEQ '$DA1'
000200        000800 DEL MOV        C                     MOVE @SSRCF    LPMV03
000300        000900 DEL MOV        C                     MOVE @SOMSN    LPMV07
000400 000200 000200 === ===        C                     MOVE @SXXX1    LPMV04
000500 000300 000300 === ===        C                     MOVE @SXXX2    LPMV05
000600 000400 000400 === ===        C                     MOVE @SELTY    LPMV08
000700 000500 000500 === ===        C                     MOVE @SELAT    LPMV09
000800 000600 000600 === ===        C                     MOVE @SELNM    LPMV01
000900 000700 000700 === ===        C                     MOVE *BLANKS   LPMV16
000200        000800 DEL TO         C                     MOVE @SSRCF    LPMV03
000300        000900 DEL TO         C                     MOVE @SOMSN    LPMV07
001000 000800 001000 === ===        C                     MOVEL@SSRCT    LPMV16
001100 000900 001100 === ===        C                     MOVE *BLANKS   LPMV17
001200 001000 001200 === ===        C                     MOVEL@SSRCQ    LPMV17
001300 001100 001300 === ===        C                     END
001400 001200 001400 === ===         **
001500 001700        MOV DEL        C                     MOVE IFMSGE    LPMSGE
001600 001300 001500 === ===        C                     MOVE IFSBMT    LPSBMT
001700 001400 001600 === ===        C                     MOVE @SELNM    LPSBMN
001800 001500 001700 === ===        C           @ITMFL    IFEQ 'N'
001900 001600 001800 === ===        C                     MOVE IFSBMN    LPSBMN
                                    /* ! Review Suggested: Begin !*/
001500 001700        TO  DEL *      C                     MOVE IFMSGE    LPMSGE
                                    /* ! Review Suggested: End   !*/
002000 001800 001900 === ===        C                     END
```

Composite output - complex activities type 1

```
0000.01      C           @PRMC     IFEQ '$DA1'
0000.02      C                     MOVE @SXXX1    LPMV04
0000.03      C                     MOVE @SXXX2    LPMV05
0000.04      C                     MOVE @SELTY    LPMV08
0000.05      C                     MOVE @SELAT    LPMV09
0000.06      C                     MOVE @SELNM    LPMV01
0000.07      C                     MOVE *BLANKS   LPMV16
0000.08      C                     MOVEL@SSRCT    LPMV16
0000.09      C                     MOVE *BLANKS   LPMV17
0000.10      C                     MOVEL@SSRCQ    LPMV17
0000.11      C                     END
0000.12       **
0000.13      C                     MOVE IFSBMT    LPSBMT
0000.14      C                     MOVE @SELNM    LPSBMN
0000.15      C           @ITMFL    IFEQ 'N'
0000.16      C                     MOVE IFSBMN    LPSBMN
0000.17      /* ! Review Suggested: Begin !*/
0000.18      C                     MOVE IFMSGE    LPMSGE
0000.19      /* ! Review Suggested: End   !*/
0000.20      C                     END
```

Interpreting the report and composite member

The following summarises the relationship between what appears in the report and what is included in the composite output:

- When a statement is deleted in the 1st version, and the same statement is moved to a different location in the 2nd version, the deleted statement will be flagged with action DEL_MOV and with action DEL_TO in the report (the two locations representing the original and target location of the move effected in the 2nd version), and will not be included in the composite member.

- When a statement is moved in the 1st version to another location, and the same statement is deleted in the 2nd version, the statement in the original location will be flagged with action MOV_DEL and will not be included in the composite member; the statement in the target location will be flagged with action TO_DEL, and will be identified as a potential conflict, and will appear between **Review Suggested** comment lines in both the report and composite member.

- By default the program assumes that the 1st version changes supersede or override changes made by the 2nd version. The composite member reflects that assumption. Since you nominate which member is known to the program as the 1st version and which is known as the 2nd version, you can reverse the preference for the 1st version by exchanging the nominated members for the 1st and 2nd versions, and re-running the merge operation.

Example of complex activities type 2

- **A statement was moved to another location in both versions, but to a different target location in each version**

```
Actions
V1  V2
```

`MOV MOV` This statement was moved to another location in both the 1st and 2nd versions. In both versions the statement was moved in the same relative direction, i.e; either downwards or upwards in the source member.

`MOV*MOV` This statement was moved to another location in both the 1st and 2nd versions. The statement was moved downwards in one version and upwards in the other version.

`TO` The statement was moved to this location in the 1st version.

`MOV<TO` This statement was moved to this location in the 2nd version. The same statement was moved in the 1st version to a location further upwards in the source member.

`    TO` The statement was moved to this location in the 2nd version.

`TO <MOV` This statement was moved to this location in the 1st version. The same statement was moved in the 2nd version to a location further upwards in the source member.

1st version - complex activities type 2

```
C                     MOVE @SELTY    LPMV08
C                     MOVE @SELAT    LPMV09
C           @PRMC     IFEQ '$DA1'
C                     MOVE @SSRCF    LPMV03
C                     MOVE @SOMSN    LPMV07
C                     MOVE @SXXX1    LPMV04
C                     MOVE @SXXX2    LPMV05
C                     MOVE @SELNM    LPMV01
C                     MOVE *BLANKS   LPMV16
C                     MOVEL@SSRCT    LPMV16
C                     MOVE *BLANKS   LPMV17
C                     MOVEL@SSRCQ    LPMV17
C                     END
 **
C                     MOVE IFSBMT    LPSBMT
C                     MOVE @SELNM    LPSBMN
C           @ITMFL    IFEQ 'N'
C                     MOVE IFSBMN    LPSBMN
C                     END
C                     MOVE IFMSGE    LPMSGE
```

2nd version - complex activities type 2

```
C           @PRMC     IFEQ '$DA1'
C                     MOVE @SSRCF    LPMV03
C                     MOVE @SOMSN    LPMV07
C                     MOVE @SXXX1    LPMV04
C                     MOVE @SXXX2    LPMV05
C                     MOVE @SELNM    LPMV01
C                     MOVE *BLANKS   LPMV16
C                     MOVEL@SSRCT    LPMV16
C                     MOVE *BLANKS   LPMV17
C                     MOVEL@SSRCQ    LPMV17
C                     MOVE @SELTY    LPMV08
C                     MOVE @SELAT    LPMV09
C                     END
 **
C                     MOVE IFSBMT    LPSBMT
C                     MOVE @SELNM    LPSBMN
C                     MOVE IFMSGE    LPMSGE
C           @ITMFL    IFEQ 'N'
C                     MOVE IFSBMN    LPSBMN
C                     END
```

Merge report - complex activities type 2

```
Base   Ver-1  Ver-2  Actions ................Source statement.................
000600 000100 001100 TO             C                     MOVE @SELTY    LPMV08
000700 000200 001200 TO             C                     MOVE @SELAT    LPMV09
000100 000300 000100 === ===        C           @PRMC     IFEQ '$DA1'
000200 000400 000200 === ===        C                     MOVE @SSRCF    LPMV03
000300 000500 000300 === ===        C                     MOVE @SOMSN    LPMV07
000400 000600 000400 === ===        C                     MOVE @SXXX1    LPMV04
000500 000700 000500 === ===        C                     MOVE @SXXX2    LPMV05
000600 000100 001100 MOV*MOV        C                     MOVE @SELTY    LPMV08
000700 000200 001200 MOV*MOV        C                     MOVE @SELAT    LPMV09
000800 000800 000600 === ===        C                     MOVE @SELNM    LPMV01
000900 000900 000700 === ===        C                     MOVE *BLANKS   LPMV16
001000 001000 000800 === ===        C                     MOVEL@SSRCT    LPMV16
001100 001100 000900 === ===        C                     MOVE *BLANKS   LPMV17
001200 001200 001000 === ===        C                     MOVEL@SSRCQ    LPMV17
                                    /* ! Review Suggested: Begin !*/
000600 000100 001100 MOV<TO  *      C                     MOVE @SELTY    LPMV08
000700 000200 001200 MOV<TO  *      C                     MOVE @SELAT    LPMV09
                                    /* ! Review Suggested: End   !*/
001300 001300 001300 === ===        C                     END
001400 001400 001400 === ===         **
001500 002000 001700 MOV MOV        C                     MOVE IFMSGE    LPMSGE
001600 001500 001500 === ===        C                     MOVE IFSBMT    LPSBMT
001700 001600 001600 === ===        C                     MOVE @SELNM    LPSBMN
001500 002000 001700     TO         C                     MOVE IFMSGE    LPMSGE
001800 001700 001800 === ===        C           @ITMFL    IFEQ 'N'
001900 001800 001900 === ===        C                     MOVE IFSBMN    LPSBMN
002000 001900 002000 === ===        C                     END
```

Composite output - complex activities type 2

```
0000.01      C                     MOVE @SELTY    LPMV08
0000.02      C                     MOVE @SELAT    LPMV09
0000.03      C           @PRMC     IFEQ '$DA1'
0000.04      C                     MOVE @SSRCF    LPMV03
0000.05      C                     MOVE @SOMSN    LPMV07
0000.06      C                     MOVE @SXXX1    LPMV04
0000.07      C                     MOVE @SXXX2    LPMV05
0000.08      C                     MOVE @SELNM    LPMV01
0000.09      C                     MOVE *BLANKS   LPMV16
0000.10      C                     MOVEL@SSRCT    LPMV16
0000.11      C                     MOVE *BLANKS   LPMV17
0000.12      C                     MOVEL@SSRCQ    LPMV17
0000.13      /* ! Review Suggested: Begin !*/
0000.14      C                     MOVE @SELTY    LPMV08
0000.15      C                     MOVE @SELAT    LPMV09
0000.16      /* ! Review Suggested: End   !*/
0000.17      C                     END
0000.18       **
0000.19      C                     MOVE IFSBMT    LPSBMT
0000.20      C                     MOVE @SELNM    LPSBMN
0000.21      C                     MOVE IFMSGE    LPMSGE
0000.22      C           @ITMFL    IFEQ 'N'
0000.23      C                     MOVE IFSBMN    LPSBMN
0000.24      C                     END
0000.25      /* ! Review Suggested: Begin !*/
0000.26      C                     MOVE IFMSGE    LPMSGE
0000.27      /* ! Review Suggested: End   !*/
```

Interpreting the report and composite member

The following summarises the relationship between what appears in the report and what is included in the composite output:

- When a statement is moved to a different target location in both versions, the activity is treated as a potential conflict. The statement will appear in the composite member in both locations where it has been moved TO. The statement moved to the location farthest downwards will appear between **Review Suggested** comment lines in both the report and composite member.
- Note that a MOV MOV and MOV*MOV are always matched with a TO, and either a MOV<TO or TO<MOV. You can associate these action codes by matching the statement line number in the appropriate column in area 1.

Example of complex activities type 3

- **A statement is inserted in both versions, and also moved by one version**

Actions
V1 V2

INS>MOV This statement was inserted at this location in the 1st version. The same statement was inserted and moved in the 2nd version to a location further downwards in the source member.

MOV<INS This statement was inserted at this location in the 2nd version. The same statement was inserted and moved in the 1st version to a location further upwards in the source member.

MOV>INS This statement was inserted at this location in the 2nd version. The same statement was inserted and moved in the 1st version to a location further downwards in the source member.

INS<MOV This statement was inserted at this location in the 1st version. The same statement was inserted and moved in the 2nd version to a location further upwards in the source member.

1st version - complex activities type 3

```
0001.00      C           @PRMC     IFEQ '$DA1'
0002.00      C                     MOVE @SSRCF    LPMV03
0003.00      C                     MOVE @SOMSN    LPMV07
0004.00      C                     MOVE @SXXX1    LPMV04
0005.00      C                     MOVE @SXXX2    LPMV05
0006.00      C                     MOVE @SELTY    LPMV08
0007.00      C                     MOVE @SELAT    LPMV09
0008.00      C                     MOVE @SXXX3    LPMV11
0009.00      C                     MOVE @SXXX4    LPMV12
0010.00      C                     MOVE @SELNM    LPMV01
0011.00      C                     MOVE *BLANKS   LPMV16
0012.00      C                     MOVEL@SSRCT    LPMV16
0013.00      C                     MOVE *BLANKS   LPMV17
0014.00      C                     MOVEL@SSRCQ    LPMV17
0015.00      C                     END
0016.00       **
0017.00      C                     MOVE IFMSGE    LPMSGE
0018.00      C                     MOVE IFSBMT    LPSBMT
0019.00      C                     MOVE @SELNM    LPSBMN
0020.00      C           @ITMFL    IFEQ 'N'
0021.00      C                     MOVE IFSBMN    LPSBMN
0022.00      C                     END
```

2nd version - complex activities type 3

```
0001.00      C           @PRMC     IFEQ '$DA1'
0002.00      C                     MOVE @SSRCF    LPMV03
0003.00      C                     MOVE @SOMSN    LPMV07
0004.00      C                     MOVE @SXXX1    LPMV04
0005.00      C                     MOVE @SELNM    LPMV01
0006.00      C                     MOVE *BLANKS   LPMV16
0007.00      C                     MOVEL@SSRCT    LPMV16
0008.00      C                     MOVE *BLANKS   LPMV17
0009.00      C                     MOVEL@SSRCQ    LPMV17
0010.00      C                     END
0011.00       **
0012.00      C                     MOVE @SXXX2    LPMV05
0013.00      C                     MOVE @SELTY    LPMV08
0014.00      C                     MOVE @SELAT    LPMV09
0015.00      C                     MOVE @SXXX3    LPMV11
0016.00      C                     MOVE @SXXX4    LPMV12
0017.00      C                     MOVE IFMSGE    LPMSGE
0018.00      C                     MOVE IFSBMT    LPSBMT
0019.00      C                     MOVE @SELNM    LPSBMN
0020.00      C           @ITMFL    IFEQ 'N'
0021.00      C                     MOVE IFSBMN    LPSBMN
0022.00      C                     END
```

Merge report - complex activities type 3

```
Base   Ver-1  Ver-2  Actions .................Source statement.................
000100 000100 000100 === ===        C           @PRMC     IFEQ '$DA1'
000200 000200 000200 === ===        C                     MOVE @SSRCF    LPMV03
000300 000300 000300 === ===        C                     MOVE @SOMSN    LPMV07
000400 000400 000400 === ===        C                     MOVE @SXXX1    LPMV04
000500 000500 001200     MOV        C                     MOVE @SXXX2    LPMV05
000600 000600 001300     MOV        C                     MOVE @SELTY    LPMV08
000700 000700 001400     MOV        C                     MOVE @SELAT    LPMV09
                                    /* ! Review Suggested: Begin !*/
       000800 001500 INS>MOV *      C                     MOVE @SXXX3    LPMV11
       000900 001600 INS>MOV *      C                     MOVE @SXXX4    LPMV12
                                    /* ! Review Suggested: End   !*/
000800 001000 000500 === ===        C                     MOVE @SELNM    LPMV01
000900 001100 000600 === ===        C                     MOVE *BLANKS   LPMV16
001000 001200 000700 === ===        C                     MOVEL@SSRCT    LPMV16
001100 001300 000800 === ===        C                     MOVE *BLANKS   LPMV17
001200 001400 000900 === ===        C                     MOVEL@SSRCQ    LPMV17
001300 001500 001000 === ===        C                     END
001400 001600 001100 === ===         **
000500 000500 001200     TO         C                     MOVE @SXXX2    LPMV05
000600 000600 001300     TO         C                     MOVE @SELTY    LPMV08
000700 000700 001400     TO         C                     MOVE @SELAT    LPMV09
       000800 001500 MOV<INS        C                     MOVE @SXXX3    LPMV11
       000900 001600 MOV<INS        C                     MOVE @SXXX4    LPMV12
001500 001700 001700 === ===        C                     MOVE IFMSGE    LPMSGE
001600 001800 001800 === ===        C                     MOVE IFSBMT    LPSBMT
001700 001900 001900 === ===        C                     MOVE @SELNM    LPSBMN
001800 002000 002000 === ===        C           @ITMFL    IFEQ 'N'
```

Composite output - complex activities type 3

```
0000.01      C           @PRMC     IFEQ '$DA1'
0000.02      C                     MOVE @SSRCF    LPMV03
0000.03      C                     MOVE @SOMSN    LPMV07
0000.04      C                     MOVE @SXXX1    LPMV04
0000.05      /* ! Review Suggested: Begin !*/
0000.06      C                     MOVE @SXXX3    LPMV11
0000.07      C                     MOVE @SXXX4    LPMV12
0000.08      /* ! Review Suggested: End   !*/
0000.09      C                     MOVE @SELNM    LPMV01
0000.10      C                     MOVE *BLANKS   LPMV16
0000.11      C                     MOVEL@SSRCT    LPMV16
0000.12      C                     MOVE *BLANKS   LPMV17
0000.13      C                     MOVEL@SSRCQ    LPMV17
0000.14      C                     END
0000.15       **
0000.16      C                     MOVE @SXXX2    LPMV05
0000.17      C                     MOVE @SELTY    LPMV08
0000.18      C                     MOVE @SELAT    LPMV09
0000.19      C                     MOVE IFMSGE    LPMSGE
0000.20      C                     MOVE IFSBMT    LPSBMT
0000.21      C                     MOVE @SELNM    LPSBMN
0000.22      C           @ITMFL    IFEQ 'N'
0000.23      C                     MOVE IFSBMN    LPSBMN
0000.24      C                     END
```

Interpreting the report and composite member

The following summarises the relationship between what appears in the report and what is included in the composite output:

- When a statement is initially inserted at the same location in both versions, and then in one version a block of statement lines, including the insertion, are moved to another location, the activity is treated as a potential conflict. The statement will appear in the composite member in the location where it has been inserted in the 1st version. It will appear between **Review Suggested** comment lines in both the report and composite member.
- Note that an INS>MOV is always matched with a MOV<INS, and that MOV>INS is always matched with an INS<MOV. You can associate these action codes by matching the statement line number in the appropriate column in area 1.
- By default the program assumes that the 1st version changes supersede or override changes made by the 2nd version. The composite member reflects that assumption. Since you nominate which member is known to the program as the 1st version and which is known as the 2nd version, you can reverse the preference for the 1st version by swapping the nominated members for the 1st and 2nd versions, and re-running the merge operation.

  The following pages show the report and composite members generated by the same merge operation but with swapped 1st and 2nd versions.

1st version - complex activities type 3 (swapped...)

```
0001.00      C           @PRMC     IFEQ '$DA1'
0002.00      C                     MOVE @SSRCF    LPMV03
0003.00      C                     MOVE @SOMSN    LPMV07
0004.00      C                     MOVE @SXXX1    LPMV04
0005.00      C                     MOVE @SELNM    LPMV01
0006.00      C                     MOVE *BLANKS   LPMV16
0007.00      C                     MOVEL@SSRCT    LPMV16
0008.00      C                     MOVE *BLANKS   LPMV17
0009.00      C                     MOVEL@SSRCQ    LPMV17
0010.00      C                     END
0011.00       **
0012.00      C                     MOVE @SXXX2    LPMV05
0013.00      C                     MOVE @SELTY    LPMV08
0014.00      C                     MOVE @SELAT    LPMV09
0015.00      C                     MOVE @SXXX3    LPMV11
0016.00      C                     MOVE @SXXX4    LPMV12
0017.00      C                     MOVE IFMSGE    LPMSGE
0018.00      C                     MOVE IFSBMT    LPSBMT
0019.00      C                     MOVE @SELNM    LPSBMN
0020.00      C           @ITMFL    IFEQ 'N'
0021.00      C                     MOVE IFSBMN    LPSBMN
0022.00      C                     END
```

2nd version - complex activities type 3 (swapped...)

```
0001.00      C           @PRMC     IFEQ '$DA1'
0002.00      C                     MOVE @SSRCF    LPMV03
0003.00      C                     MOVE @SOMSN    LPMV07
0004.00      C                     MOVE @SXXX1    LPMV04
0005.00      C                     MOVE @SXXX2    LPMV05
0006.00      C                     MOVE @SELTY    LPMV08
0007.00      C                     MOVE @SELAT    LPMV09
0008.00      C                     MOVE @SXXX3    LPMV11
0009.00      C                     MOVE @SXXX4    LPMV12
0010.00      C                     MOVE @SELNM    LPMV01
0011.00      C                     MOVE *BLANKS   LPMV16
0012.00      C                     MOVEL@SSRCT    LPMV16
0013.00      C                     MOVE *BLANKS   LPMV17
0014.00      C                     MOVEL@SSRCQ    LPMV17
0015.00      C                     END
0016.00       **
0017.00      C                     MOVE IFMSGE    LPMSGE
0018.00      C                     MOVE IFSBMT    LPSBMT
0019.00      C                     MOVE @SELNM    LPSBMN
0020.00      C           @ITMFL    IFEQ 'N'
0021.00      C                     MOVE IFSBMN    LPSBMN
0022.00      C                     END
```

Merge report - complex activities type 3 (swapped...)

```
Base   Ver-1  Ver-2  Actions .................Source statement..................
000100 000100 000100 === ===        C           @PRMC     IFEQ '$DA1'
000200 000200 000200 === ===        C                     MOVE @SSRCF    LPMV03
000300 000300 000300 === ===        C                     MOVE @SOMSN    LPMV07
000400 000400 000400 === ===        C                     MOVE @SXXX1    LPMV04
000500 001200 000500 MOV            C                     MOVE @SXXX2    LPMV05
000600 001300 000600 MOV            C                     MOVE @SELTY    LPMV08
000700 001400 000700 MOV            C                     MOVE @SELAT    LPMV09
       001500 000800 MOV>INS        C                     MOVE @SXXX3    LPMV11
       001600 000900 MOV>INS        C                     MOVE @SXXX4    LPMV12
000800 000500 001000 === ===        C                     MOVE @SELNM    LPMV01
000900 000600 001100 === ===        C                     MOVE *BLANKS   LPMV16
001000 000700 001200 === ===        C                     MOVEL@SSRCT    LPMV16
001100 000800 001300 === ===        C                     MOVE *BLANKS   LPMV17
001200 000900 001400 === ===        C                     MOVEL@SSRCQ    LPMV17
001300 001000 001500 === ===        C                     END
001400 001100 001600 === ===         **
000500 001200 000500 TO             C                     MOVE @SXXX2    LPMV05
000600 001300 000600 TO             C                     MOVE @SELTY    LPMV08
000700 001400 000700 TO             C                     MOVE @SELAT    LPMV09
                                    /* ! Review Suggested: Begin !*/
       001500 000800 INS<MOV *      C                     MOVE @SXXX3    LPMV11
       001600 000900 INS<MOV *      C                     MOVE @SXXX4    LPMV12
                                    /* ! Review Suggested: End   !*/
001500 001700 001700 === ===        C                     MOVE IFMSGE    LPMSGE
001600 001800 001800 === ===        C                     MOVE IFSBMT    LPSBMT
001700 001900 001900 === ===        C                     MOVE @SELNM    LPSBMN
001800 002000 002000 === ===        C           @ITMFL    IFEQ 'N'
```

Composite output - complex activities type 3 (swapped...)

```
0000.02      C                     MOVE @SSRCF    LPMV03
0000.03      C                     MOVE @SOMSN    LPMV07
0000.04      C                     MOVE @SXXX1    LPMV04
0000.05      C                     MOVE @SELNM    LPMV01
0000.06      C                     MOVE *BLANKS   LPMV16
0000.07      C                     MOVEL@SSRCT    LPMV16
0000.08      C                     MOVE *BLANKS   LPMV17
0000.09      C                     MOVEL@SSRCQ    LPMV17
0000.10      C                     END
0000.11       **
0000.12      C                     MOVE @SXXX2    LPMV05
0000.13      C                     MOVE @SELTY    LPMV08
0000.14      C                     MOVE @SELAT    LPMV09
0000.15      /* ! Review Suggested: Begin !*/
0000.16      C                     MOVE @SXXX3    LPMV11
0000.17      C                     MOVE @SXXX4    LPMV12
0000.18      /* ! Review Suggested: End   !*/
0000.19      C                     MOVE IFMSGE    LPMSGE
0000.20      C                     MOVE IFSBMT    LPSBMT
0000.21      C                     MOVE @SELNM    LPSBMN
0000.22      C           @ITMFL    IFEQ 'N'
0000.23      C                     MOVE IFSBMN    LPSBMN
0000.24      C                     END
```

Conflict comment lines

When changes are made by both the 1st and 2nd versions to the same set of statement lines of the base version, the program identifies this as either a conflict or a potential conflict.

Even as intelligent assumptions are made in the development of the composite member, the program identifies that conflicts or potential conflicts exist so that you can review the results of the default integration. In some cases, you may want to manually adjust the composite file to select an alternative to the base assumptions.

When the composite member is created and a conflict or a potential conflict is identified between changes made in the two versions, the program inserts a line before and after the conflicting lines, in both the composite and report members. These lines have the format:

```
PPPPPPP ! Review Required: Begin ! SSSSSSS
   (conflicting statement lines)
   (conflicting statement lines)
   (conflicting statement lines)
PPPPPPP ! Review Required: End ! SSSSSSS
```

or

```
PPPPPPP ! Review Suggested: Begin ! SSSSSSS
   (potentially conflicting statement lines)
   (potentially conflicting statement lines)
   (potentially conflicting statement lines)
PPPPPPP ! Review Suggested: End ! SSSSSSS
```

where **PPPPPPP** and **SSSSSSS** are the character strings which you specify for parameters PREFIX and SUFFIX when requesting the merge. The first prefix character is inserted in column position 1 of the composite and report members.

On the merge report, all conflicting or potentially conflicting statement lines appear with an ***** next to the action codes.

Re-executing compare and merge operations

You can re-execute one or all compare and merge operations which were previously executed and logged under the same execution request name.

Use **option 60** from menu ONE to get to menu ONE2; followed by **option 1**. Alternatively you can prompt command EXCMRGRQS.

You have to specify the execution request name to be used:

- The default is *USER.
- You can use F4 to show a list of all existing execution request names found in the system.

Once you have specified the execution request name, you have to select the required sequence number:

- The default is *ALL, i.e; all entries in the specified execution request name will be re-executed.
- You can specify *LAST, i.e; the last sequence in the specified execution request name will be re-executed.
- You can use F4 to show a list of all existing sequence numbers under the specified execution request name, and select the sequence number you want. The list shows against each sequence number the name of the Base Version member used. Since the same request can contain both compare and merge operation logs, the Base Version member name is prefixed with C: to designate the sequence as a compare log, or M: to designate the sequence as a merge log.
- You can also specify keyword *SEARCH to search for the last entry under the specified execution request name involving a specific member name. Then you have to specify the member name to search.

For a detailed description of all parameters, refer to *Command EXCMRGRQS* on page 77.

Removing compare and merge logs

You can remove one or all execution request logs for operations which were previously executed and logged under the same execution request name.

Removing execution request logs does not remove, or in any way change, the report and composite members which were generated by these operations.

Once you have removed the execution request logs, you cannot access the generated report and composite members via command STRMRGSEU, or re-execute these operations via command EXCMRGRQS. Use this house-keeping function to remove logs which are no longer needed.

Use **option 60** from menu ONE to get to menu ONE2; followed by **option 2**. Alternatively you can prompt command RMVMRGRQS.

You have to specify the execution request name to be used:

- The default is *USER.
- You can use F4 to show a list of all existing execution request names found in the system.

Once you have specified the execution request name, you have to select the required sequence number:

- The default is *ALL, i.e; all entries in the specified execution request name will be removed.
- You can use F4 to show a list of all existing sequence numbers under the specified execution request name, and select the sequence number you want. The list shows against each sequence number the name of the Base Version member used. Since the same request can contain both compare and merge operation logs, the Base Version member name is prefixed with C: to designate the sequence as a compare log, or M: to designate the sequence as a merge log.

For a detailed description of all parameters, refer to *Command RMVMRGRQS* on page 95.

Interface with Thenon/SEE

Purpose and benefits

Interface with Thenon/SEE allows extended usage and control of the Concurrent Development feature of Thenon/SEE:

- Typically for large programs, you can assign two programmers to work concurrently on the same program, and then later merge both changes into a single CR source member. This can provide significant productivity gains.
- In the case where an emergency fix is carried out while long term development of a new version is underway, you can now reduce (or in most cases, eliminate) programming time for coding the fix in the new version under development.
- You can create permanent links between source members in different CRs, or even outside the CR environment, including the compare and/or merge parameters, so that changes can continuously be made to the two versions being concurrently developed and the composite source member re-generated ad-hoc.
- From within function RTVCRSRC (Retrieve CR Source) you can:
    - Compare source member currently being developed with its original live/production version
    - Compare frozen source member with its original live/production version
    - Compare site/group specific source member with either other site/group specific, or with the base application source members
    - Compare any two source members (of the same function) which are being developed concurrently in two different CRs
    - Merge site/group specific source member with other site/group specific source
    - Merge any two source members (of the same function) which are being developed concurrently in two different CRs
- From within function WRKCROBJ (Work with CR Objects) you can:
    - re-generate and access the latest compare report and merge composite involving a member which is already registered under the CR, via a set of new action options
    - perform any ad-hoc compare and/or merge operation using any source member you have access to, via a set of new status options

How the interface works

The interface with Thenon/SEE is achieved by evoking contextually the various Thenon/ONE commands. Each compare/merge related option appearing in Thenon/SEE is associated with a Thenon/ONE command. Whenever a compare or merge operation is requested, the command parameters are set automatically, depending on your request, and are prompted so you can verify the values being used and alter them if necessary.

While using compare/merge options, the Execution Request Name (parameter EXCRQS) is automatically set to the name: CRxxxxxxxx, where xxxxxxxx is the IR/CR number of the CR from which the compare and merge operation is requested. The Execution Request Sequence (parameter EXCRQSSEQ) is automatically set to *NEXT. This allows all the operation details to be logged separately under each CR.

The default source file names for the generated report and composite members are unchanged, that is, they are CMPREPORT, MRGREPORT and MRGOUTPUT respectively, and the library is *CURLIB. This results with the generated members typically being stored in the CR library (which is used as the job current library when working with CR objects).

You can also request ad-hoc compare and merge functions, as if you were using Thenon/ONE menu options, through the Status window in function WRKCROBJ (Work with CR Objects).

Creating compare and merge links between CR source members

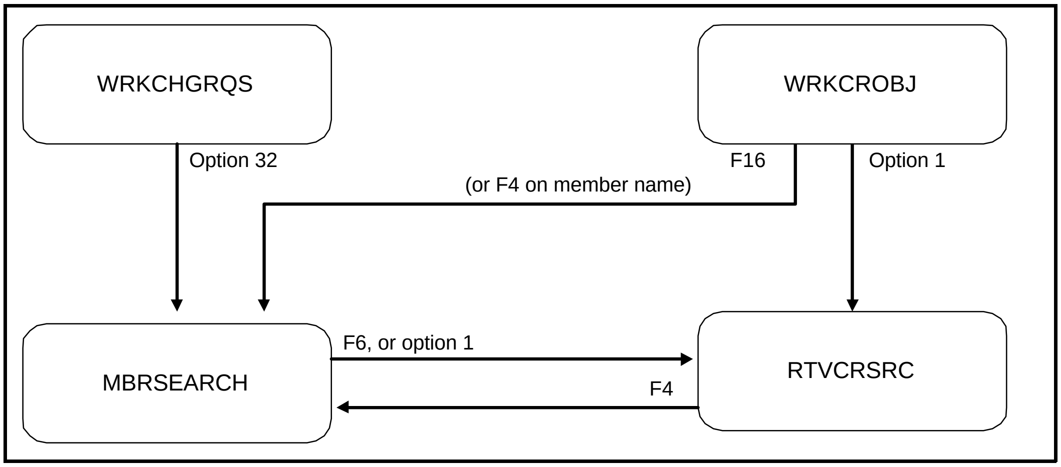
You create the compare and merge links by requesting a compare or merge operation from function RTVCRSRC (Retrieve CR Source).

In the normal course of development, the RTVCRSRC panel appears only when errors are detected when **option 1** is selected from function WRKCROBJ (Work with CR Objects), or when you attempt to initiate a concurrent development cycle for a source member which is already under development.

You can explicitly get to the RTVCRSRC panel by selecting **option 1** from the first line in WRKCROBJ panel without specifying object name (that is, object name is left with blanks).

Alternatively, you can use F6 from function MBRSEARCH (Source Member Search) panel, which is shown when you use F16 from WRKCROBJ panel, or when you select **option 32** from WRKCHGRQS (Work with Change Requests).

To create compare and merge links you must first get the RTVCRSRC panel. The following diagram illustrates the connectivity and the way to get to this panel as explained above:



The RTVCRSRC panel shows the overall member status throughout the change management cycle for the CR application. Once a member is selected, the top section of the list shows:

- all versions of the member which are currently being developed, the associated CR number, retrieval type and CR status code
- all versions of the member which have been 'frozen', the associated CR number, retrieval type and CR status code
- all versions of the member which are currently in the Live/Production environment, and their current version number

The bottom section of the list shows:

- all additional new versions of the member which can be initiated for the application base, site and group specific objects (if any)

The following options can be used to execute compare or merge operations and to establish a link between the source members which appear in the top section of the list:

Option 45=Compare

- You can compare any two members appearing in the top section of the list
- You can select a single member, if that member is a subsequent version of a live/production version, i.e; it is not a new member. A compare operation will be initiated using the selected member as the Changed Version, and the current live/production member as the Base Version.

For a further description, refer to *Specifying and executing a simple compare operation* on page 11, *Command CMPSRCMBR* on page 71, and *Example of a compare operation* on page 103.

If the compare operation completed normally, message id MRG0004 will indicate the name and location of the generated report source member.

Option 46=Merge

- You can merge any two members appearing in the top section of the list
- Two, and only two, members must be specified; one will be used as 1st Version and the other as 2nd Version. The Base Version will be determined automatically in the following order:
    - If one of the selected members is a current live/production member, this member will be used as both 1st and Base versions. If both members are current live/production members, the one for base application (*BAS) will be used as both 1st and Base versions. The other member will be used as 2nd Version.
    - If both selected members are currently under development, and both have originated from the same live/production member, then this common live/production member will be used as Base Version, the selected member with earliest retrieval date will be used as 1st Version, and the other member will be used as 2nd Version.
    - If both selected members are new members (i.e; they did not originate from any existing live/production member), the selected member with the earliest retrieval date will be used as both 1st and Base versions, and the other member will be used as 2nd Version.

For a further description, refer to *Specifying and executing a simple merge operation* on page 17, *Command MRGSRCMBR* on page 87, and *Example of a merge operation* on page 113.

If the merge operation completed normally, message id MRG0004 will indicate the name and location of the generated report source member, and message id MRG0005 will indicate the name and location of the generated composite source member.

Refreshing compare and merge links

All the compare and merge operations you execute from function RTVCRSRC (using default settings) are logged, regardless of the location of the input and generated members, under an execution request specific to the CR from which you initiate the operations.

You can re-execute all operations which are logged under the CR. This is useful if the input members for the compare or merge operation are under continuous development and you need to regenerate the report and composite members.

The following options from function WRKCROBJ can be used to re-execute previous operations:

Option 47=Exc Cmp/Mrg - from the Status window

- You can re-execute all operations previously logged under the CR
- You can use the *LAST keyword to re-execute the latest operation logged under the CR
- You can use the *SEARCH keyword to search and re-execute the latest operation involving a specific member name

Option 47=Exc Cmp/Mrg - against a member registered in the CR

- You can use this option directly from the WRKCROBJ main panel to search the CR execution request and re-execute the latest operation involving the selected member name. This is like using the Status window option with the *SEARCH keyword, but the command is not prompted.

For a further description, refer to *Re-executing compare and merge operations* on page 47, and *Command EXCMRGRQS* on page 77.

Accessing compare and merge report and composite members

All the compare and merge operations you execute from function RTVCRSRC (using default settings) are logged, regardless of the location of the input and generated members, under an execution request specific to the CR from which you initiate the operations.

The following options from function WRKCROBJ can be used to access the report and composite members:

Option 48=Cmp/Mrg Rept - from the Status window

- You can use the *LAST keyword to access the report generated by the latest operation logged under the CR
- You can use the *SEARCH keyword to search and access the report generated by the latest operation involving a specific member name

Option 48=Cmp/Mrg Rept - against a member registered in the CR

- You can use this option directly from the WRKCROBJ main panel to search the CR execution request, and access the report generated by the latest operation involving the selected member name. This is like using the Status window option with the *SEARCH keyword, but the command is not prompted.

For a further description, refer to *Accessing the generated compare report* on page 14, *Accessing the generated merge report and composite members* on page 20, and *Command STRMRGSEU* on page 97.

Option 49=Mrg Composite - from the Status window

- You can use the *LAST keyword to access the composite member generated by the latest operation logged under the CR.

- You can use the *SEARCH keyword to search and access the composite member generated by the latest operation involving a specific member name.

Option 49=Mrg Composite - against a member registered in the CR

- You can use this option directly from the WRKCROBJ main panel to search the CR execution request, and access the composite member generated by the latest operation involving the selected member name. This is like using the Status window option with the *SEARCH keyword, but the command is not prompted.

For a further description, refer to *Accessing the generated merge report and composite members* on page 20, and *Command STRMRGSEU* on page 97.

Summary of Thenon/ONE options in Thenon/SEE

Thenon/ONE Command	**Menu**	**Opt**	**Thenon/SEE Function**	**Opt**	**Usage**
CMPSRCMBR	ONE	01	RTVCRSRC WRKCROBJ	45 45 (Status)	Contextual compare - prompt Ad-hoc compare - prompt
MRGSRCMBR	ONE	02	RTVCRSRC WRKCROBJ	46 46 (Status)	Contextual merge - prompt Ad-hoc merge - prompt
STRMRGSEU	ONE	21	WRKCROBJ	48 (Status) 48	Access generated report - prompt Access generated report - no prompt
	ONE	22	WRKCROBJ	49 (Status) 49	Access generated composite - prompt Access generated composite - no prompt
EXCMRGRQS	ONE2	01	WRKCROBJ	47 (Status) 47	Re-execute / Refresh links - prompt Re-execute / Refresh links - no prompt
RMVMRGRQS	ONE2	02			Remove execution request logs
CMPSRCF	ONE	11			Bulk compare - *generic* members
MRGSRCF	ONE	12			Bulk merge - *generic* members
CHGOBJDFT	ONE2	15			Change object defaults for FRMCOL, TOCOL, PREFIX and SUFFIX

Appendix A: Command summary

Command CHGOBJDFT: Change Object Defaults

The Change Object Defaults (CHGOBJDFT) command allows you to change the default source file name and the way source is managed for each source based object type used under Thenon/SEE. It also allows you to change default column positions and comment prefix and suffix which are used by Thenon/ONE Compare and Merge Manager.

Minimum required Thenon/SEE authority: ***TECH / Configuration Manager.**

Object reference id (OBJREF)

Specifies the THENON object reference (id) for which the defaults are changed. Use F4 to prompt for a list of valid object ids. Once you specify the required object reference id, the current default values will be shown. You can then change them to different values.

This is a required parameter.

Default source file (DFTSRCF)

Specifies the default source file associated with a source based object reference id. It is applicable to Thenon/SEE.

If you prompt this command with a valid object reference id, the current default value is shown. You can then change it to a different default value. Possible values are:

source-file-name
Specify the default source file name associated with this object reference id.

*SAME
Retain the existing default value.

The default source file name is used when:

- creating the CR library. For each object id configured for the CR application, the default source file name is created within the CR library.
- interpretive (*INTERPRET), memo (*MEMO) or copy reference (*CPYREF) source members are promoted within the change management cycle, the member is copied into the default source file name in the designated target library (as configured for the application, or as overridden for the object).

The default source file is also used in the application configuration process when defining the application live source pools. However, once you have specified the pool source file name and library name, any later change to the default source file does not affect your pool configuration.

This facility is provided for use when THENON is installed initially, or when an object type is configured for use for the first time.

The following should be considered when changing the default source file name after one or more objects of the specified object id have been, or are, under Thenon Change Management:

You have to ensure that the default source file exists in each of the following libraries:

- All existing CR libraries containing one or more objects of the specified object id.
- All libraries configured to accept *INTERPRET, *MEMO or *CPYREF source members. You can do this by creating the source file, or renaming the previously used default source file name to the new name.

Source usage (SRCUSG)

Specifies the way source is managed in Thenon/SEE. Possible values are:

*COMPILE

- Source member is used to compile an object.
- Source is promoted at the development centre only to the live source pool.
- Source is loaded into the release packet if either Distribute source code or Re-compile objs at remote sites are specified in the application configuration as *YES, or if Load source to release packet is specified in object overrides as Y.
- Source is promoted at remote sites if either Distribute source code is specified in the application configuration as *YES, or if Unload source from release packet is specified in object overrides as Y; program type member is moved, database type member is copied.
- Target library and distribution can be overridden on object level.

*MEMO

- Source member is used as a memorandum, for example: program specifications, or internal documentation.
- Source is promoted at the development centre only to the live source pool.
- Source is loaded into the release packet if Distribute source code is specified in the application configuration as *YES.
- Source is promoted at remote sites if Distribute source code is specified in the application configuration as *YES; the member is moved.
- No object level overrides can be specified.

*INTERPRET

- Source member is used as run-time data (interpretive source).
- Source is promoted at the development centre to all target environments; the member is copied; the member is also promoted to the live source pool.
- Source is always loaded to the release packet and promoted at remote sites; the member is copied.
- Target library can be overridden on object level.

*CPYREF

- Source member is used as compile time copy reference, for example: member referenced in /COPY statement in RPG program.
- Source member is promoted at the development centre to all target environments; the member is moved; the member is also promoted to the live source pool.
- Source is loaded into the release packet if either Distribute source code or Re-compile objs at remote sites are specified in the application configuration as *YES, or if Load source to release packet is specified in object overrides as Y.
- Source is promoted at remote sites if either Distribute source code is specified in the application configuration as *YES, or if Unload source from release packet is specified in object overrides as Y; the member is moved.
- Target library and distribution can be overridden on object level.

*SAME
Retain the existing source usage value.

This facility is provided for use when Thenon is installed initially, or when an object type is configured for usage for the first time.

When changing the source usage after one or more objects, of the specified object id, have been, or are, under Thenon Change Management, you should review the usage of the default source file name in all target libraries.

Source record length (SRCRCDLEN)

Specifies the source file record length.

The record length you specify here is used by Thenon/SEE when creating source files in CR libraries and in temporary work areas.

If you prompt this command with a valid object reference id, the current value is shown. You can then change it to a different value. Possible values are:

record-length

Specify the record length. The value you specify must be equal to or greater than the value you have specified for parameter TOCOL.

*SAME

Retain the existing record length value.

This facility is provided for use when Thenon is installed initially, or when an object type is configured for use for the first time.

When changing the source usage after one or more objects of the specified object id have been, or are, under Thenon Change Management, you should review the usage of the default source file name in all target libraries.

Source statement from column (FRMCOL)

Specifies the starting position of source statements in source members.

The source statement starting position is used by Thenon/ONE Compare & Merge Manager to identify characters in the source file which are eligible for compare and/or merge operations. If you prompt this command with a valid object reference id, the current value is shown. You can then change it to a different value. Possible values are:

column-position

Specify a numeric value greater than or equal to 1 and less than the value specified for parameter TOCOL.

*SAME

Retain the existing from-column-position value.

Source statement to column (TOCOL)

Specifies the ending position of source statements in source members.

The source statement ending position is used by Thenon/ONE Compare & Merge Manager to identify characters in the source file which are eligible for compare and/or merge operations. If you prompt this command with a valid object reference id, the current value is shown. You can then change it to a different value. Possible values are:

column-position

Specify a numeric value greater than the value specified for parameter FRMCOL and less than or equal to the value specified for parameter SRCRCDLEN.

*SAME

Retain the existing to-column-position value.

Source comment prefix (PREFIX)

Specifies the character string which is used as the prefix for comment source statements.

The comment prefix string is used by Thenon/ONE Compare & Merge Manager when a comment is inserted in the report and composite source members which are generated by the compare or merge operations. If you prompt this command with a valid object reference id, the current value is shown. You can then change it to a different value. Possible values are:

character-string
Specify the comment prefix. To include leading blanks, enter a quoted character string.

*SAME
Retain the existing prefix value.

Source comment suffix (SUFFIX)

Specifies the character string which is used as the suffix for comment source statements.

The comment suffix string is used by Thenon/ONE Compare & Merge Manager when a comment is inserted in the report and composite source members which are generated by the compare or merge operations. If you prompt this command with a valid object reference id, the current value is shown. You can then change it to a different value. Possible values are:

character-string
Specify the comment suffix. To include leading blanks, enter a quoted character string.

*SAME
Retain the existing suffix value.

Command CMPSRCF: Compare Source File

The Compare Source File (CMPSRCF) command allows you to compare all members of any two source files. You can specify a source file which will contain the compare report members. The command will scan all members in the base and changed version source files, and will perform compare operations for every member name which is found in both these source files. Error messages are issued for any name which is found in one, but not in both source files. An information message is sent to the job log for every member being processed, so that you can determine which members are missing from which source file. Also, a completion message will indicate the number of members processed, the number of compare operations which have completed successfully, and the number of compare operations which terminated with errors. None of the input source files are changed or deleted by this command.

Minimum required Thenon/SEE authority: ***PGMR / Development Manager.**

Member name (MBR)

Specifies the name, or *generic* name of the members to be processed. Possible values are:

*ALL
All members are processed.

member-name
Specify the member name to be processed.

generic name
Type a partial member name followed by an asterisk (*) to process a list of members that meet the specific criteria. The generic name can be in one of the following formats:

ABC*
Processes a list of all member names that begin with the characters ABC. For example: ABC or ABCZZZ.

*ABC
Processes a list of all member names ending with the characters ABC. For example: ABC or ZZZABC.

ABC
Processes a list of all member names that include the characters ABC anywhere in the name. For example: ABC, ZZZABC, ABCZZZ or ZZZABCZZZ.

AB*C
Processes a list of all member names that begin with the characters AB and end with the character C. For example: ABC or ABZZZC.

This is a required parameter.

Base version source file (BASEFILE)

Specifies the name and library of the source file that contains the base version members. Possible library values are:

*LIBL
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located.

This is a required parameter.

Changed version source file (CHGFILE)

Specifies the name and library of the source file that contains the changed version members. Possible library values are:

*LIBL
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located.

This is a required parameter.

Execution request name (EXCRQS)

Specifies the execution request name under which this compare operation is logged.

Execution request name can be associated with each compare or merge operation. When you request a compare operation or a merge operation you can specify a request name and sequence. The values of all command parameters are then logged under the specified request name and sequence for the purpose of re-executing the same operation at a later time, or for accessing the associated output members which are generated by the compare or merge operations. For every member name processed by this command, a separate log entry is added to the execution request you specify here. Possible values are:

*USER
The execution request name is the same as the current job user profile name. Use this option to log all operations under your name. Since this is the default on all Thenon/ONE commands, it is the most convenient option to use.

request-name
Specify the execution request name under which this compare operation is logged.

Execution request sequence (EXCRQSSEQ)

Specifies the execution request sequence for the first member processed by this merge operation. Subsequent members will be logged sequentially under the specified execution request name.

Every log entry in the execution request name is assigned a sequence number in the range of 001 - 999. You can have a maximum of 999 operations logged under any given request name; a mixture of compare and merge operations can be logged under the same request name. If you log an operation under a request sequence which was previously used, the details of the previous operations are erased, and the details of the current operation are recorded against the specified request name and sequence. Possible values are:

sequence-number

Specify the actual sequence number to be used for the first member in the list. The default is 001. All existing log entries in the specified execution request are erased.

*NEXT

The sequence number to be used for the first member in the list is the last used sequence number, in the specified request name, plus 1. Use this option if you want to unconditionally retain all existing logs under the specified execution request name.

Execution request text (EXCRQSTXT)

Specifies the execution request text which will appear in all compare report headers generated by this operation. Possible values are:

*BASEMBRTXT

The base version member text is used as the execution request text.

text

Specify no more than 50 characters of text, enclosed in apostrophes.

Report source file (REPTFILE)

Specifies the name and library of the source file that contains the report members. For each member processed successfully by this operation, a report is generated in a member name the same as the processed members, in the source file you specify here. If the source file does not exist, it will be created by this command.

The default source file name is CMPREPORT.

Possible library values are:

*CURLIB

The current library for the job is used to locate the source file. If the source file is not found, it is created in the current library. If no library is specified as the current library for the job, library QGPL is used.

*LIBL
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located. If the source file is not found, it is created in the specified library.

Report format (RPTFMT)

Specifies the type of report which is generated for each member by this compare operation. Possible values are:

*FULL or 'F'
Full report is generated, showing all source statements from the base and changed source members, and the action codes associated with each statement.

*SUMM or 'S'
Summary report is generated, showing only statements which have changed in the changed version as compared to the base version.

Compress blanks ? (COMPRESS)

Specifies whether differences in the number of continuous blanks embedded in the source statement are identified by this merge operation as actual source changes. Possible values are:

*YES or '1'
Continuous blanks are compressed, and differences in the number of continuous blanks are ignored.

*NO or '0'
Blanks are not compressed, and differences in the number of continuous blanks are detected as source changes by this merge operation.

Analyse from column (FRMCOL)

Specifies the starting column position for source statement comparison.

This parameter is relevant to column oriented source syntax, where certain positions in the source statement are reserved for comments, like columns 1-5 in RPG syntax. In these circumstances you can restrict the comparison process to certain starting and ending positions. Possible values are:

*BASESRCTYP
The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

column-position
Specify the starting column position for this merge operation. It must be a number in the range of 001-999 and less than the value specified for parameter TOCOL.

Analyse to column (TOCOL)

Specifies the ending column position for source statement comparison.

This parameter is relevant to column oriented source syntax, where certain positions in the source statement are reserved for comments, like columns 1-5 in RPG syntax. In these circumstances you can restrict the comparison process to certain starting and ending positions. Possible values are:

*<u>BASESRCTYP</u>

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

column-position

Specify the ending column position for this merge operation. It must be a number in the range of 001-999 and greater than the value specified for parameter FRMCOL.

Prefix for comments (PREFIX)

Specifies the prefix used when comments are inserted in the report and composite members by this merge operation.

In certain circumstances, the program inserts a comment line in the member(s) which are generated, to warn you of unresolved conflicts. The comment is prefixed and suffixed by the character strings you specify in parameters PREFIX and SUFFIX. Possible values are:

*<u>BASESRCTYP</u>

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

character-value

Specify a string no longer than 7 characters, enclosed in apostrophes, which will be used as comment prefix.

Suffix for comments (SUFFIX)

Specifies the suffix used when comments are inserted in the report and composite members by this merge operation.

In certain circumstances, the program inserts a comment line in the member(s) which are generated, to warn you of unresolved conflicts. The comment is prefixed and suffixed by the character strings you specify in parameters PREFIX and SUFFIX. Possible values are:

*BASESRCTYP

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

character-value

Specify a string no longer than 7 characters, enclosed in apostrophes, which will be used as comment suffix.

Command CMPSRCMBR: Compare Source Member

The Compare Source Member (CMPSRCMBR) command allows you to compare any two source members. You can specify a source member which will contain the compare report. None of the input members are changed or deleted by this command.

Minimum required Thenon/SEE authority: ***PGMR / Development Manager.**

Base version member (BASEMBR)

Specifies the name of the source member which is used as the base for this compare operation. The changed version source member (specified in parameter CHGMBR) includes changes made to the source statements contained in the base version member specified here. This is a required parameter.

Base version source file (BASEFILE)

Specifies the name and library of the source file that contains the base version member (which was specified in parameter BASEMBR). Possible library values are:

*<u>LIBL</u>
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located.

This is a required parameter.

Changed version member (CHGMBR)

Specifies the name of the source member which is used as the changed version member for this compare operation. Possible values are:

*<u>BASEMBR</u>
The changed version member name is the same as the base member name (as specified in parameter BASEMBR).

member-name
Specify the changed version member name.

Changed version source file (CHGFILE)

Specifies the name and library of the source file that contains the changed version member (which was specified in parameter CHGMBR). Possible library values are:

*<u>LIBL</u>
The library list is used to locate the source file.

library-name

Specify the name of the library where the source file is located.

This is a required parameter.

Execution request name (EXCRQS)

Specifies the execution request name under which this compare operation is logged.

Execution request name can be associated with each compare or merge operation. When you request a compare operation or a merge operation you can specify a request name and sequence. The values of all command parameters are then logged under the specified request name and sequence for the purpose of re-executing the same operation at a later time, or for accessing the associated output members which are generated by the compare or merge operations. Possible values are:

*USER

The execution request name is the same as the current job user profile name. Use this option to log all operations under your name. Since this is the default on all Thenon/ONE commands, it is the most convenient option to use.

*BASEMBR

The execution request name is the same as the base version member name as specified in parameter BASEMBR.

*NONE

This compare operation is not logged; execution request name is not required.

request-name

Specify the execution request name under which this compare operation is logged.

Execution request sequence (EXCRQSSEQ)

Specifies the execution request sequence under which this compare operation is logged.

Every log entry in the execution request name is assigned a sequence number in the range of 001 - 999. You can have a maximum of 999 operations logged under any given request name; a mixture of compare and merge operations can be logged under the same request name. If you log an operation under a request sequence which was previously used, the details of the previous operations are erased, and the details of the current operation are recorded against the specified request name and sequence. Possible values are:

sequence-number

Specify the actual sequence number to be used. The default is 001.

*NEXT

The sequence number to be used is the last used sequence number in the specified request name, plus 1. Use this option if you want to unconditionally retain all existing logs under the specified execution request name.

Execution request text (EXCRQSTXT)

Specifies the execution request text which will appear in the compare report headers. Possible values are:

*BASEMBRTXT
: The base version member (as specified in parameter BASEMBR) text is used as the execution request text.

text
: Specify no more than 50 characters of text, enclosed in apostrophes.

Report member (REPTMBR)

Specifies the name of the source member which will hold the report which is generated by this compare operation. If the member does not exist, it will be created by this command. Possible values are:

*BASEMBR
: The report member name is the same as the base member name (as specified in parameter BASEMBR).

member-name
: Specify the report member name.

Report source file (REPTFILE)

Specifies the name and library of the source file that contains the report member (which was specified in parameter REPTMBR). If the source file does not exist, it is created by this command.

The default source file name is CMPREPORT.

Possible library values are:

*CURLIB
: The current library for the job is used to locate the source file. If the source file is not found, it is created in the current library. If no library is specified as the current library for the job, library QGPL is used.

*LIBL
: The library list is used to locate the source file.

library-name
: Specify the name of the library where the source file is located. If the source file is not found, it is created in the specified library.

Report format (RPTFMT)

Specifies the type of report which is generated by this compare operation. Possible values are:

*FULL or 'F'
Full report is generated, showing all source statements from the base and changed version members, and the action codes associated with each statement.

*SUMM or 'S'
Summary report is generated, showing only statements which have changed in the changed version as compared to the base version.

Compress blanks ? (COMPRESS)

Specifies whether differences in the number of continuous blanks embedded in the source statement are identified by this compare operation as actual source changes. Possible values are:

*YES or '1'
Continuous blanks are compressed, and differences in the number of continuous blanks are ignored.

*NO or '0'
Blanks are not compressed, and differences in the number of continuous blanks are detected as source changes by this compare operation.

Analyse from column (FRMCOL)

Specifies the starting column position for source statement comparison.

This parameter is relevant to column oriented source syntax, where certain positions in the source statement are reserved for comments, like columns 1-5 in RPG syntax. In these circumstances you can restrict the comparison process to certain starting and ending positions. Possible values are:

*BASESRCTYP
The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

column-position
Specify the starting column position for this compare operation. It must be a number in the range of 001-999 and less than the value specified for parameter TOCOL.

Analyse to column (TOCOL)

Specifies the ending column position for source statement comparison.

This parameter is relevant to column oriented source syntax, where certain positions in the source statement are reserved for comments, like columns 1-5 in RPG syntax. In these circumstances you can restrict the comparison process to certain starting and ending positions. Possible values are:

*<u>BASESRCTYP</u>

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

column-position

Specify the ending column position for this compare operation. It must be a number in the range of 001-999 and greater than the value specified for parameter FRMCOL.

Prefix for comments (PREFIX)

Specifies the prefix used when comments are inserted in the report member by this compare operation.

In certain circumstances, the program inserts a comment line in the member(s) which are generated to warn you of unresolved conflicts. The comment is prefixed and suffixed by the character strings you specify in parameters PREFIX and SUFFIX. Possible values are:

*<u>BASESRCTYP</u>

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

character-value

Specify a string no longer than 7 characters, enclosed in apostrophes, which will be used as the comment prefix.

Suffix for comments (SUFFIX)

Specifies the suffix used when comments are inserted in the report member by this compare operation.

In certain circumstances, the program inserts a comment line in the member(s) which are generated to warn you of unresolved conflicts. The comment is prefixed and suffixed by the character strings you specify in parameters PREFIX and SUFFIX. Possible values are:

*<u>BASESRCTYP</u>

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

character-value

Specify a string no longer than 7 characters, enclosed in apostrophes, which will be used as comment suffix.

Command EXCMRGRQS: Execute Compare/Merge Request

The Execute Compare/Merge Request (EXCMRGRQS) command allows you to execute one or more compare and/or merge operations which were previously executed and logged under an execution request.

Minimum required Thenon/SEE authority: ***PGMR / Development Manager.**

Execution request name (EXCRQS)

Specifies the name of the execution request which will be processed by this command.

Execution request name can be associated with each compare or merge operation. When you request a compare or merge operation you can specify a request name and sequence. The values of all command parameters are then logged under the specified request name and sequence for the purpose of re-executing the same operation at a later time, or for accessing the associated output members which are generated by the compare or merge operations. Possible values are:

*<u>USER</u>
The execution request name to be used is the same as the current job user profile name.

request-name
Specify the execution request name to be processed by this command.

You can use F4 to show a list of existing execution request names which can be selected. If you specify an execution request name which does not exist, the command will terminate with errors.

Execution request sequence (EXCRQSSEQ)

Specifies the execution request sequence (within the execution request name specified in parameter EXCRQS) which will be processed by this command.

Every log entry in the execution request name is assigned a sequence number in the range of 001 - 999. You can have a maximum of 999 operations logged under any given request name; a mixture of compare and merge operations can be logged under the same request name. If you log an operation under a request sequence which was previously used, the details of the previous operations are erased, and the details of the current operation are recorded against the specified request name and sequence. Possible values are:

*<u>LAST</u>
The last sequence number logged under the specified execution request name will be processed.

*ALL
All log entries under the specified execution request name will be processed sequentially.

*SEARCH
The specified execution request name will be searched for the latest log entry referring to the member name specified in parameter SCHMBR. The latest sequence referring to that name as either base member, or changed member in a compare operation, or base member, version-1 member or version-2 member in a merge operation, will be processed.

sequence-number

Specify the execution request sequence number which will be executed.

You can use F4 to show a list of existing sequence numbers for the execution request name you have selected in parameter EXCRQS. In the following example, 3 entries are shown for execution request name CR88889721:

```
Execution request sequence . . .  _______
 *ALL
 *LAST
 *SEARCH
or one of CR88889721 sequences:
  001  C:RPGPGMA
  002  M:DSPF001FM
  003  C:CLPGMC
```

The sequence number is highlighted. Against each sequence number the base member name used in the operation is shown, prefixed by C: for Compare request, or M: for Merge request.

Search member name (SCHMBR)

Specifies the name of the member which will be used to search the log entries under the specified execution request name. The search is in LIFO (last in first out) order; search will terminate with a match if either the base member or changed member name in a compare operation, or if either the base member, version-1 or version-2 member name in a merge operation match the member name you specify here. The matched sequence will be processed.

This parameter is prompted only if EXCRQSSEQ(*SEARCH) is specified.

This is a required parameter.

Command MRGSRCF: Merge Source File

The Merge Source File (MRGSRCF) command allows you to merge all members of any two source files which have originated from the same source code.

You can specify a source file which will contain the merge report members and a source file which will contain the merge composite source members. The command will scan all members in the base, 1st version and 2nd version source files, and will perform a merge operation for every member name which is found in all these source files. Error messages are issued for any name which is found in one or more source files but which is not found in all source files. An information message is sent to the job log for every member being processed, so that you can determine which members are missing from which source file. Also, a completion message will indicate the number of members processed, the number of merge operations which have completed successfully, and the number of merge operations which terminated with errors. None of the input source files are changed or deleted by this command.

Minimum required Thenon/SEE authority: ***PGMR / Development Manager.**

Member name (MBR)

Specifies the name, or *generic* name of the members to be processed. Possible values are:

*ALL
All members are processed.

member-name
Specify the member name to be processed.

generic name
Type a partial member name followed by an asterisk (*) to process a list of members that meet the specific criteria. The generic name can be in one of the following formats:

ABC*
Processes a list of all member names that begin with the characters ABC. For example: ABC or ABCZZZ.

*ABC
Processes a list of all member names ending with the characters ABC. For example: ABC or ZZZABC.

ABC
Processes a list of all member names that include the characters ABC anywhere in the name. For example: ABC, ZZZABC, ABCZZZ or ZZZABCZZZ.

AB*C
Processes a list of all member names that begin with the characters AB and end with the character C. For example: ABC or ABZZZC.

This is a required parameter.

Base version source file (BASEFILE)

Specifies the name and library of the source file that contains the base version members. Possible library values are:

*LIBL
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located.

This is a required parameter.

1st version source file (VER1FILE)

Specifies the name and library of the source file that contains the 1st version members. Possible library values are:

*LIBL
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located.

This is a required parameter.

2nd version source file (VER2FILE)

Specifies the name and library of the source file that contains the 2nd version members. Possible library values are:

*LIBL
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located.

This is a required parameter.

Execution request name (EXCRQS)

Specifies the execution request name under which this merge operation is logged.

Execution request name can be associated with each compare or merge operation. When you request a compare or merge operation you can specify a request name and sequence. The values of all command parameters are then logged under the specified request name and sequence for the purpose of re-executing the same operation at a later time, or for accessing the associated output members which are generated by the compare or merge operations. For every member name processed by this command, a separate log entry is added to the execution request you specify here. Possible values are:

*USER

The execution request name is the same as the current job user profile name. Use this option to log all operations under your name. Since this is the default on all Thenon/ONE commands, it is the most convenient option to use.

request-name

Specify the execution request name under which this merge operation is logged.

Execution request sequence (EXCRQSSEQ)

Specifies the execution request sequence for the first member processed by this merge operation. Subsequent members will be logged sequentially under the specified execution request name.

Every log entry in the execution request name is assigned a sequence number in the range of 001 - 999. You can have a maximum of 999 operations logged under any given request name; a mixture of compare and merge operations can be logged under the same request name. If you log an operation under a request sequence which was previously used, the details of the previous operations are erased, and the details of the current operation are recorded against the specified request name and sequence. Possible values are:

sequence-number

Specify the actual sequence number to be used for the first member in the list. The default is 001. All existing log entries in the specified execution request are erased.

*NEXT

The sequence number to be used for the first member in the list is the last used sequence number in the specified request name, plus 1. Use this option if you want to unconditionally retain all existing logs under the specified execution request name.

Execution request text (EXCRQSTXT)

Specifies the execution request text which will appear in all merge report headers generated by this operation. Possible values are:

*BASEMBRTXT

The base version member text is used as the execution request text.

text

Specify no more than 50 characters of text, enclosed in apostrophes.

Report source file (REPTFILE)

Specifies the name and library of the source file that contains the report members. For each member processed successfully by this operation, a report is generated in a member name the same as the processed members, in the source file you specify here. If the source file does not exist, it will be created by this command.

The default source file name is MRGREPORT.

Possible library values are:

*CURLIB
The current library for the job is used to locate the source file. If the source file is not found, it is created in the current library. If no library is specified as the current library for the job, library QGPL is used.

*LIBL
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located. If the source file is not found, it is created in the specified library.

Composite source file (COMPFILE)

Specifies the name and library of the source file that contains the composite members. For each member processed successfully by this operation, a composite source is generated in a member name same as the processed members, in the source file you specify here. Possible values are:

source-file-name
The default source file name is MRGOUTPUT; you can change it to any valid source file name.

*NONE
No composite members are generated by this merge operation; only report members are generated.

If a name other than *NONE is specified, and if the file does not exist, it will be created by this command.

Possible library values are:

*CURLIB
The current library for the job is used to locate the source file. If the source file is not found, it is created in the current library. If no library is specified as the current library for the job, library QGPL is used.

*LIBL
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located. If the source file is not found, it is created in the specified library.

Report format (RPTFMT)

Specifies the type of report which is generated for each member by this merge operation. Possible values are:

*FULL or 'F'

Full report is generated, showing all source statements from the base, 1st version and 2nd version source members; the action codes associated with each statement, and the resulting composite member statements.

*SUMM or 'S'

Summary report is generated, showing only statements which have changed in 1st and 2nd versions as compared to the base version.

Replace composite member ? (RPLCOMPMBR)

Specifies whether each generated composite member is allowed to replace any existing member. Possible values are:

*YES or '1'

If the composite member name already exists, it will be replaced by the composite source which is generated by this merge operation.

*NO or '0'

If the composite member name already exists, and it contains one or more records, it will not be replaced by this operation. The merge operation for that member will terminate with errors.

Compress blanks ? (COMPRESS)

Specifies whether differences in the number of continuous blanks embedded in the source statement are identified by this merge operation as actual source changes. Possible values are:

*YES or '1'

Continuous blanks are compressed, and differences in the number of continuous blanks are ignored.

*NO or '0'

Blanks are not compressed, and differences in the number of continuous blanks are detected as source changes by this merge operation.

Analyse from column (FRMCOL)

Specifies the starting column position for source statement comparison.

This parameter is relevant to column oriented source syntax, where certain positions in the source statement are reserved for comments, like columns 1-5 in RPG syntax. In these circumstances you can restrict the comparison process to certain starting and ending positions. Possible values are:

*BASESRCTYP

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

column-position

Specify the starting column position for this merge operation. It must be a number in the range of 001-999 and less than the value specified for parameter TOCOL.

Analyse to column (TOCOL)

Specifies the ending column position for source statement comparison.

This parameter is relevant to column oriented source syntax, where certain positions in the source statement are reserved for comments, like columns 1-5 in RPG syntax. In these circumstances you can restrict the comparison process to certain starting and ending positions. Possible values are:

*<u>BASESRCTYP</u>

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

column-position

Specify the ending column position for this merge operation. It must be a number in the range of 001-999 and greater than the value specified for parameter FRMCOL.

Prefix for comments (PREFIX)

Specifies the prefix used when comments are inserted in the report and composite members by this merge operation.

In certain circumstances, the program inserts a comment line in the member(s) which are generated to warn you of unresolved conflicts. The comment is prefixed and suffixed by the character strings you specify in parameters PREFIX and SUFFIX. Possible values are:

*<u>BASESRCTYP</u>

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

character-value

Specify a string no longer than 7 characters, enclosed in apostrophes, which will be used as comment prefix.

Suffix for comments (SUFFIX)

Specifies the suffix used when comments are inserted in the report and composite members by this merge operation.

In certain circumstances, the program inserts a comment line in the member(s) which are generated to warn you of unresolved conflicts. The comment is prefixed and suffixed by the character strings you specify in parameters PREFIX and SUFFIX. Possible values are:

*BASESRCTYP

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

character-value

Specify a string no longer than 7 characters, enclosed in apostrophes, which will be used as comment suffix.

Command MRGSRCMBR: Merge Source Member

The Merge Source Member (MRGSRCMBR) command allows you to merge any two source members which have originated from the same source code. You can specify a source member which will contain the merge report, and a member which will contain the merge composite source. None of the input members are changed or deleted by this command.

Minimum required Thenon/SEE authority: ***PGMR / Development Manager.**

Base version member (BASEMBR)

Specifies the name of the source member which is used as the base for comparison and merge.

The 1st version and 2nd version source members (specified in parameters VER1MBR and VER2MBR) include changes made to the source statements contained in the base version member specified here. This is a required parameter.

Base version source file (BASEFILE)

Specifies the name and library of the source file that contains the base version member (which was specified in parameter BASEMBR). Possible library values are:

*LIBL
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located.

This is a required parameter.

1st version member (VER1MBR)

Specifies the name of the source member which is used as the first version member for comparison and merge. Possible values are:

*BASEMBR
The 1st version member name is the same as the base member name (as specified in parameter BASEMBR).

member-name
Specify the 1st version member name.

1st version source file (VER1FILE)

Specifies the name and library of the source file that contains the 1st version member (which was specified in parameter VER1MBR). Possible library values are:

*LIBL
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located.

This is a required parameter.

2nd version member (VER2MBR)

Specifies the name of the source member which is used as the second version member for comparison and merge. Possible values are:

*BASEMBR
The 2nd version member name is the same as the base member name (as specified in parameter BASEMBR).

member-name
Specify the 2nd version member name.

2nd version source file (VER2FILE)

Specifies the name and library of the source file that contains the 2nd version member (which was specified in parameter VER2MBR). Possible library values are:

*LIBL
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located.

This is a required parameter.

Execution request name (EXCRQS)

Specifies the execution request name under which this merge operation is logged.

Execution request name can be associated with each compare or merge operation. When you request a compare or merge operation you can specify a request name and sequence. The values of all command parameters are then logged under the specified request name and sequence for the purpose of re-executing the same operation at a later time, or for accessing the associated output members which are generated by the compare or merge operations. Possible values are:

*USER
The execution request name is the same as the current job user profile name. Use this option to log all operations under your name. Since this is the default on all Thenon/ONE commands, it is the most convenient option to use.

*BASEMBR
The execution request name is the same as the base version member name as specified in parameter BASEMBR.

*NONE

This merge operation is not logged; execution request name is not required.

request-name

Specify the execution request name under which this merge operation is logged.

Execution request sequence (EXCRQSSEQ)

Specifies the execution request sequence under which this merge operation is logged.

Every log entry in the execution request name is assigned a sequence number in the range of 001 - 999. You can have a maximum of 999 operations logged under any given request name; a mixture of compare and merge operations can be logged under the same request name. If you log an operation under a request sequence which was previously used, the details of the previous operations are erased, and the details of the current operation are recorded against the specified request name and sequence. Possible values are:

sequence-number

Specify the actual sequence number to be used. The default is 001.

*NEXT

The sequence number to be used is the last used sequence number in the specified request name, plus 1. Use this option if you want to unconditionally retain all existing logs under the specified execution request name.

Execution request text (EXCRQSTXT)

Specifies the execution request text which will appear in this merge report headers. Possible values are:

*BASEMBRTXT

The base version member (as specified in parameter BASEMBR) text is used as the execution request text.

text

Specify no more than 50 characters of text, enclosed in apostrophes.

Report member (REPTMBR)

Specifies the name of the source member which will hold the report which is generated by this merge operation. If the member does not exist, it will be created by this command. Possible values are:

*BASEMBR

The report member name is the same as the base member name (as specified in parameter BASEMBR).

member-name

Specify the report member name.

Report source file (REPTFILE)

Specifies the name and library of the source file that contains the report member (which was specified in parameter REPTMBR). If the source file does not exist, it will be created by this command.

The default source file name is MRGREPORT.

Possible library values are:

*CURLIB
The current library for the job is used to locate the source file. If the source file is not found, it is created in the current library. If no library is specified as the current library for the job, library QGPL is used.

*LIBL
The library list is used to locate the source file.

library-name
Specify the name of the library where the source file is located. If the source file is not found, it is created in the specified library.

Composite member (COMPMBR)

Specifies the name of the source member which will hold the composite source which is generated by this merge operation. If the member does not exist, it will be created by this command. Possible values are:

*BASEMBR
The composite member name is the same as the base member name (as specified in parameter BASEMBR).

member-name
Specify the composite member name.

Composite source file (COMPFILE)

Specifies the name and library of the source file that contains the composite member (which was specified in parameter COMPMBR). Possible values are:

source-file-name
The default source file name is MRGOUTPUT; you can specify any other valid member name.

*NONE
No composite member is generated by this merge operation; only report member is generated.

If a name other than *NONE is specified, and if the file does not exist, it will be created by this command. Possible library values are:

*CURLIB

The current library for the job is used to locate the source file. If the source file is not found, it is created in the current library. If no library is specified as the current library for the job, library QGPL is used.

*LIBL

The library list is used to locate the source file.

library-name

Specify the name of the library where the source file is located. If the source file is not found, it is created in the specified library.

Report format (RPTFMT)

Specifies the type of report which is generated by this merge operation. Possible values are:

*FULL or 'F'

Full report is generated, showing all source statements from the base, 1st version and 2nd version source members, the action codes associated with each statement and the resulting composite member statements.

*SUMM or 'S'

Summary report is generated, showing only statements which have changed in 1st and 2nd versions as compared to the base version.

Replace composite member ? (RPLCOMPMBR)

Specifies whether the generated composite member is allowed to replace any existing member. Possible values are:

*YES or '1'

If the composite member name (as specified in parameters COMPMBR and COMPFILE) already exists, it will be replaced by the composite source which is generated by this merge operation.

*NO or '0'

If the composite member name (as specified in parameters COMPMBR and COMPFILE) already exists, and it contains one or more records, it will not be replaced by this operation. The requested merge operation will terminate with errors.

Compress blanks ? (COMPRESS)

Specifies whether differences in the number of continuous blanks embedded in the source statement are identified by this merge operation as actual source changes. Possible values are:

*YES or '1'

Continuous blanks are compressed, and differences in the number of continuous blanks are ignored.

*NO or '0'

Blanks are not compressed, and differences in the number of continuous blanks are detected as source changes by this merge operation.

Analyse from column (FRMCOL)

Specifies the starting column position for source statement comparison.

This parameter is relevant to column oriented source syntax, where certain positions in the source statement are reserved for comments, like columns 1-5 in RPG syntax. In these circumstances you can restrict the comparison process to certain starting and ending positions. Possible values are:

*BASESRCTYP

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

column-position

Specify the starting column position for this merge operation. It must be a number in the range of 001-999 and less than the value specified for parameter TOCOL.

Analyse to column (TOCOL)

Specifies the ending column position for source statement comparison.

This parameter is relevant to column oriented source syntax, where certain positions in the source statement are reserved for comments, like columns 1-5 in RPG syntax. In these circumstances you can restrict the comparison process to certain starting and ending positions. Possible values are:

*BASESRCTYP

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

column-position

Specify the ending column position for this merge operation. It must be a number in the range of 001-999 and greater than the value specified for parameter FRMCOL.

Prefix for comments (PREFIX)

Specifies the prefix used when comments are inserted in the report and composite members by this merge operation.

In certain circumstances, the program inserts a comment line in the member(s) which are generated to warn you of unresolved conflicts. The comment is prefixed and suffixed by the character strings you specify in parameters PREFIX and SUFFIX. Possible values are:

*BASESRCTYP

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

character-value

Specify a string no longer than 7 characters, enclosed in apostrophes, which will be used as comment prefix.

Suffix for comments (SUFFIX)

Specifies the suffix used when comments are inserted in the report and composite members by this merge operation.

In certain circumstances, the program inserts a comment line in the member(s) which are generated to warn you of unresolved conflicts. The comment is prefixed and suffixed by the character strings you specify in parameters PREFIX and SUFFIX. Possible values are:

*BASESRCTYP

The value associated with the base version member source type will be used. Thenon holds internal information about each source type, including the starting and ending column positions and the prefix and suffix for embedded comments. You can view and change these values by using Thenon command Change Object Defaults (CHGOBJDFT).

character-value

Specify a string no longer than 7 characters, enclosed in apostrophes, which will be used as comment suffix.

Command RMVMRGRQS: Remove Compare/Merge Request

The Remove Compare/Merge Request (RMVMRGRQS) command allows you to erase one or more existing execution request log entries.

Minimum required Thenon/SEE authority: ***PGMR / Development Manager.**

Execution request name (EXCRQS)

Specifies the name of the execution request which will be processed by this command.

Execution request name can be associated with each compare or merge operation. When you request a compare operation or a merge operation you can specify a request name and sequence. The values of all command parameters are then logged under the specified request name and sequence for the purpose of re-executing the same operation at a later time, or for accessing the associated output members which are generated by the compare or merge operations. Possible values are:

*<u>USER</u>
The execution request name to be used is the same as the current job user profile name.

request-name
Specify the execution request name to be processed by this command.

You can use F4 to show a list of existing execution request names which can be selected. If you specify an execution request name which does not exist, the command will terminate with errors.

Execution request sequence (EXCRQSSEQ)

Specifies the execution request sequence (within the execution request name specified in parameter EXCRQS) which will be processed by this command.

Every log entry in the execution request name is assigned a sequence number in the range of 001 - 999. You can have a maximum of 999 operations logged under any given request name; a mixture of compare and merge operations can be logged under the same request name. If you log an operation under a request sequence which was previously used, the details of the previous operations are erased, and the details of the current operation are recorded against the specified request name and sequence. Possible values are:

*<u>ALL</u>
All log entries under the specified execution request name will be removed.

sequence-number
Specify the execution request sequence number which will be removed.

You can use F4 to show a list of existing sequence numbers for the execution request name you have selected in parameter EXCRQS. In the following example, 3 entries are shown for execution request name CR88889721:

```
  Execution request sequence . . .  _______
         *ALL
        or one of CR88889721 sequences:
          001  C:RPGPGMA
          002  M:DSPF001FM
          003  C:CLPGMC
```

The sequence number is highlighted. Against each sequence number the base member name used in the operation is shown, prefixed by C: for Compare request, or M: for Merge request.

Command STRMRGSEU: Start SEU of Compare/Merge Members

The Start SEU of Compare/Merge Generated Members (STRMRGSEU) command allows you to access the source members which were generated by the compare or merge operations.

Minimum required Thenon/SEE authority: ***PGMR / Development Manager.**

Execution request name (EXCRQS)

Specifies the name of the execution request which will be processed by this command.

Execution request name can be associated with each compare or merge operation. When you request a compare or merge operation you can specify a request name and sequence. The values of all command parameters are then logged under the specified request name and sequence for the purpose of re-executing the same operation at a later time, or for accessing the associated output members which are generated by the compare or merge operations.

Possible values are:

*<u>USER</u>
The execution request name to be used is the same as the current job user profile name.

request-name
Specify the execution request name to be processed by this command.

You can use F4 to show a list of existing execution request names which can be selected. If you specify an execution request name which does not exist, the command will terminate with errors.

Execution request sequence (EXCRQSSEQ)

Specifies the execution request sequence (within the execution request name specified in parameter EXCRQS) which will be processed by this command.

Every log entry in the execution request name is assigned a sequence number in the range of 001 - 999. You can have a maximum of 999 operations logged under any given request name; a mixture of compare and merge operations can be logged under the same request name. If you log an operation under a request sequence which was previously used, the details of the previous operations are erased, and the details of the current operation are recorded against the specified request name and sequence.

Possible values are:

*<u>LAST</u>
The last sequence number logged under the specified execution request name will be processed.

*SEARCH

The specified execution request name will be searched for the latest log entry referring to the member name specified in parameter SCHMBR. The latest sequence referring to that name as either base member, or changed member in a compare operation, or base member, version-1 member or version-2 member in a merge operation, will be processed.

sequence-number

Specify the execution request sequence number which will be processed.

You can use F4 to show a list of existing sequence numbers for the execution request name you have selected in parameter EXCRQS. In the following example, 3 entries are shown for execution request name CR88889721:

```
 Execution request sequence . . .  _______
  *LAST
  *SEARCH
or one of CR88889721 sequences:
 001  C:RPGPGMA
 002  M:DSPF001FM
 003  C:CLPGMC
```

The sequence number is highlighted. Against each sequence number the base member name used in the operation is shown, prefixed by C: for Compare request, or M: for Merge request.

Search member name (SCHMBR)

Specifies the name of the member which will be used to search the log entries under the specified execution request name. The search is in LIFO (last in first out) order; search will terminate with a match if either the base member or changed member name in a compare operation, or if either the base member, version-1 or version-2 member name in a merge operation match the member name you specify here. The matched sequence will be processed.

This parameter is prompted only if EXCRQSSEQ(*SEARCH) is specified.

This is a required parameter.

Generated member type (MBRTYPE)

Specifies the type of generated member you want to access. Possible values are:

*REPORT or 'R'

The compare or merge report will be accessed.

*COMPOSITE or 'C'

The merge composite member will be accessed. If this option is specified for a log entry of a compare operation, the command will terminate with errors. If you use the EXCRQSSEQ(*SEARCH) option, the latest merge operation referring to the specified member is searched for, that is, any compare operation log entries will be ignored.

SEU option (OPTION)

Specifies the SEU function to perform on the selected member. Possible values are:

*BROWSE or '5'
Go to Browse display.

*EDIT or '2'
Go to Edit display.

*PRINT or '6'
Print the selected member.

Appendix B: Minimum required authority table

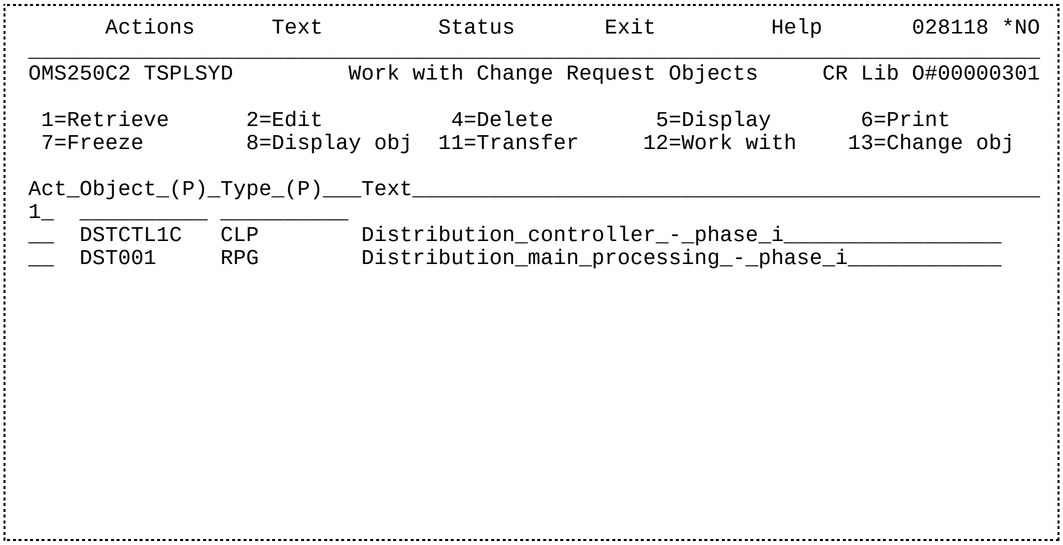
All Thenon/ONE commands are subject to Thenon/SEE user authorisation mechanism. The following table shows the minimum authorisation level the user must have to use each command.

User authorisation levels are assigned when users are enrolled in Thenon/SEE via function WRKUSRAUT (Work with User Authorisations).

FUNCTION	*VIEW	*PGMR	*APPL	*OPER	*TECH	*FULL
CHGOBJDFT: Change object defaults					X	X
CMPSRCF: Compare source file		X	X	X	X	X
CMPSRCMBR: Compare source member		X	X	X	X	X
EXCMRGRQS: Execute compare/merge request		X	X	X	X	X
MRGSRCF: Merge source file		X	X	X	X	X
MRGSRCMBR: Merge source member		X	X	X	X	X
RMVMRGRQS: Remove compare/merge request		X	X	X	X	X
STRMRGSEU: Start SEU of compare/merge mbrs		X	X	X	X	X

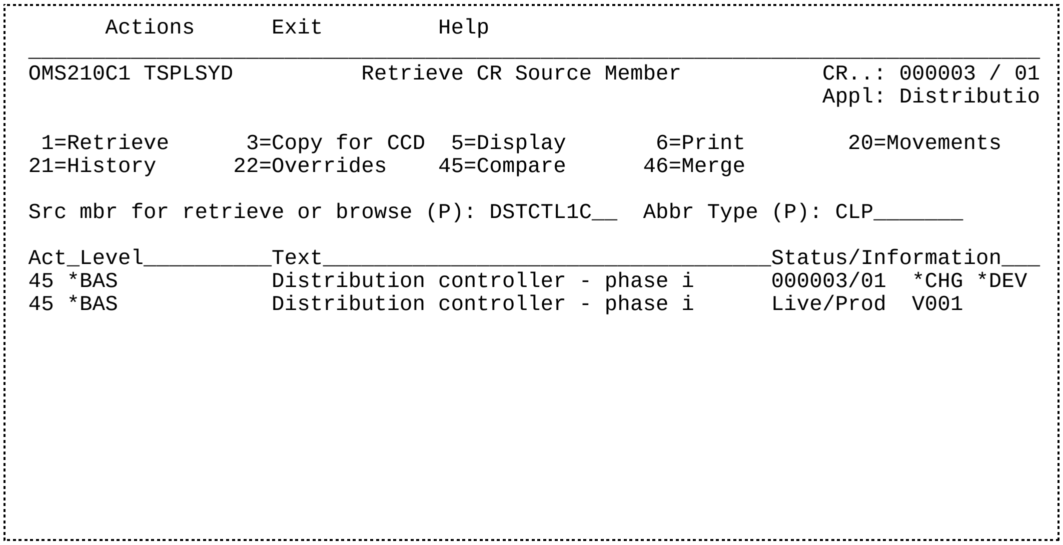
Appendix C: Example of a compare operation

In the following example of a compare operation, program DSTCTL1C was retrieved into CR 000003/01. You now want to determine the changes made to the CR source, by comparing the modified source with the current live version.



```
      Actions      Text         Status       Exit         Help       028118 *NO
_______________________________________________________________________________
OMS250C2 TSPLSYD         Work with Change Request Objects     CR Lib O#00000301

 1=Retrieve      2=Edit          4=Delete        5=Display       6=Print
 7=Freeze        8=Display obj  11=Transfer     12=Work with    13=Change obj

Act_Object_(P)_Type_(P)___Text_________________________________________________
1_  __________ __________
__  DSTCTL1C   CLP        Distribution_controller_-_phase_i_________________
__  DST001     RPG        Distribution_main_processing_-_phase_i____________
```

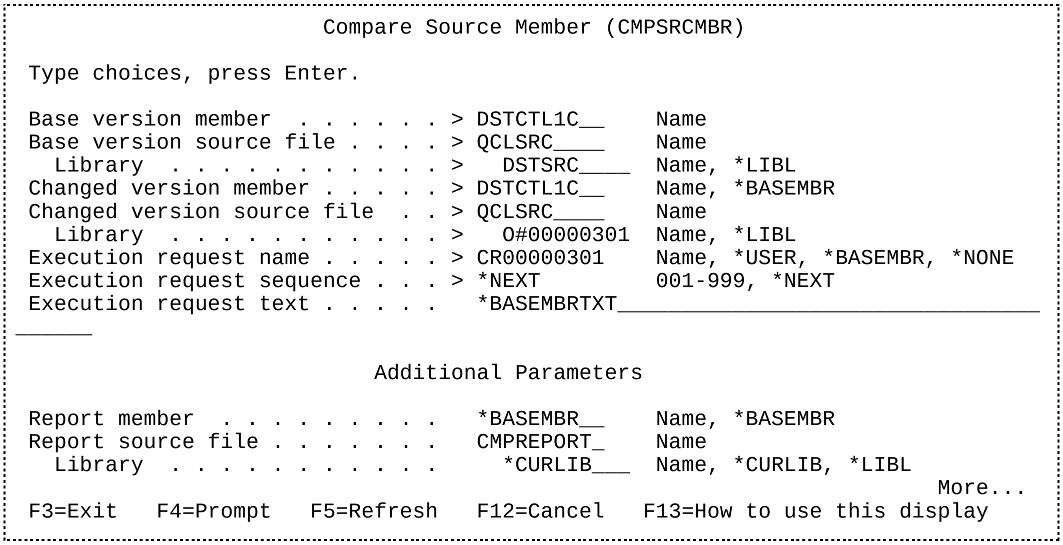
- Type 1 on the top line, and leave the name and type with blanks. The panel for function RTVCRSRC appears:



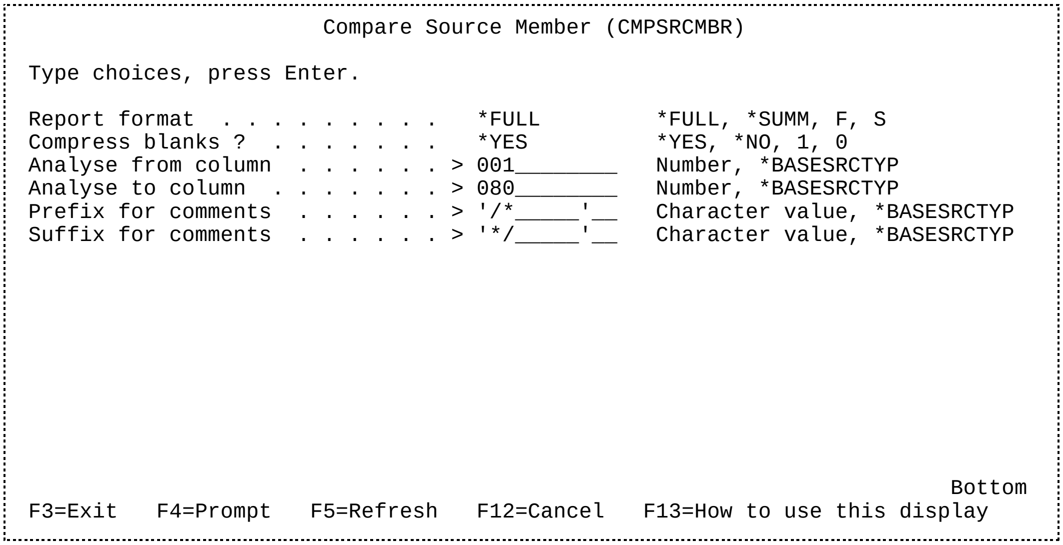
```
      Actions      Exit         Help
_______________________________________________________________________________
OMS210C1 TSPLSYD          Retrieve CR Source Member           CR..: 000003 / 01
                                                              Appl: Distributio

 1=Retrieve      3=Copy for CCD  5=Display       6=Print        20=Movements
21=History      22=Overrides    45=Compare      46=Merge

Src mbr for retrieve or browse (P): DSTCTL1C__  Abbr Type (P): CLP_______

Act_Level__________Text___________________________________Status/Information___
45 *BAS            Distribution controller - phase i      000003/01  *CHG *DEV
45 *BAS            Distribution controller - phase i      Live/Prod  V001
```

- Type the program name and type DSTCTL1C / CLP to show the development status for the program. Two entries are shown: one is for the modified current CR version, the other is for the current live/production version.

- Select both the current CR version and the current live/production, using the Compare **option 45**. Command CMPSRCMBR is prompted, with all parameters set to reflect your selection:



```
                       Compare Source Member (CMPSRCMBR)

 Type choices, press Enter.

 Base version member  . . . . . . > DSTCTL1C__    Name
 Base version source file . . . . > QCLSRC____    Name
   Library  . . . . . . . . . . . >   DSTSRC____  Name, *LIBL
 Changed version member . . . . . > DSTCTL1C__    Name, *BASEMBR
 Changed version source file  . . > QCLSRC____    Name
   Library  . . . . . . . . . . . >   O#00000301  Name, *LIBL
 Execution request name . . . . . > CR00000301    Name, *USER, *BASEMBR, *NONE
 Execution request sequence . . . > *NEXT         001-999, *NEXT
 Execution request text . . . . .   *BASEMBRTXT_________________________________
______

                           Additional Parameters

 Report member  . . . . . . . . .   *BASEMBR__    Name, *BASEMBR
 Report source file . . . . . . .   CMPREPORT_    Name
   Library  . . . . . . . . . . .     *CURLIB___  Name, *CURLIB, *LIBL
                                                                       More...
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
```

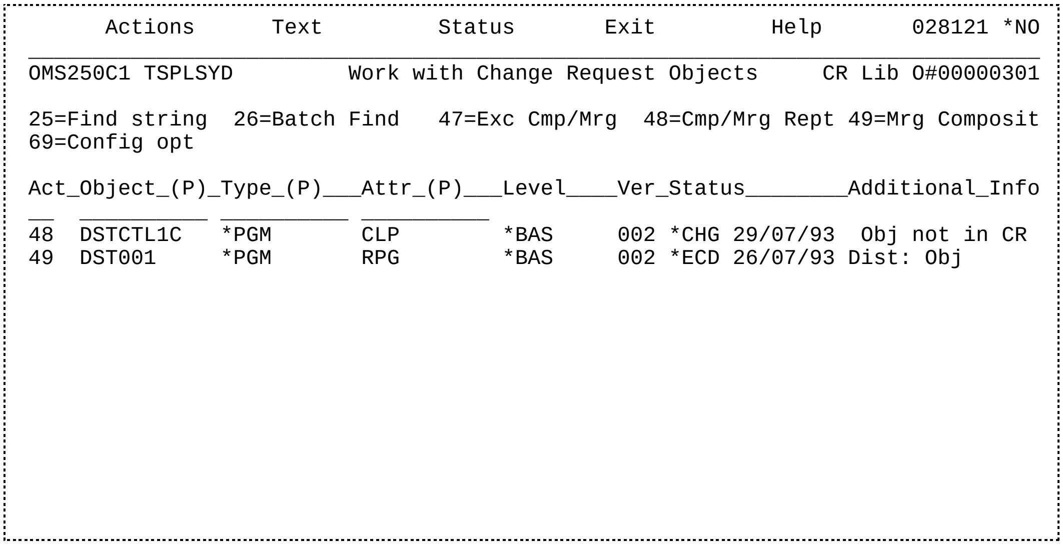


```
                       Compare Source Member (CMPSRCMBR)

 Type choices, press Enter.

 Report format  . . . . . . . . .   *FULL         *FULL, *SUMM, F, S
 Compress blanks ?  . . . . . . .   *YES          *YES, *NO, 1, 0
 Analyse from column  . . . . . . > 001________   Number, *BASESRCTYP
 Analyse to column  . . . . . . . > 080________   Number, *BASESRCTYP
 Prefix for comments  . . . . . . > '/*_____'__   Character value, *BASESRCTYP
 Suffix for comments  . . . . . . > '*/_____'__   Character value, *BASESRCTYP











                                                                        Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
```

- Note that the compare report default is member name DSTCTL1C in source file CMPREPORT in your CR library, and that the compare operation will be logged under execution request CR00000301, so after generation you can access the report via **option 48** in function WRKCROBJ.

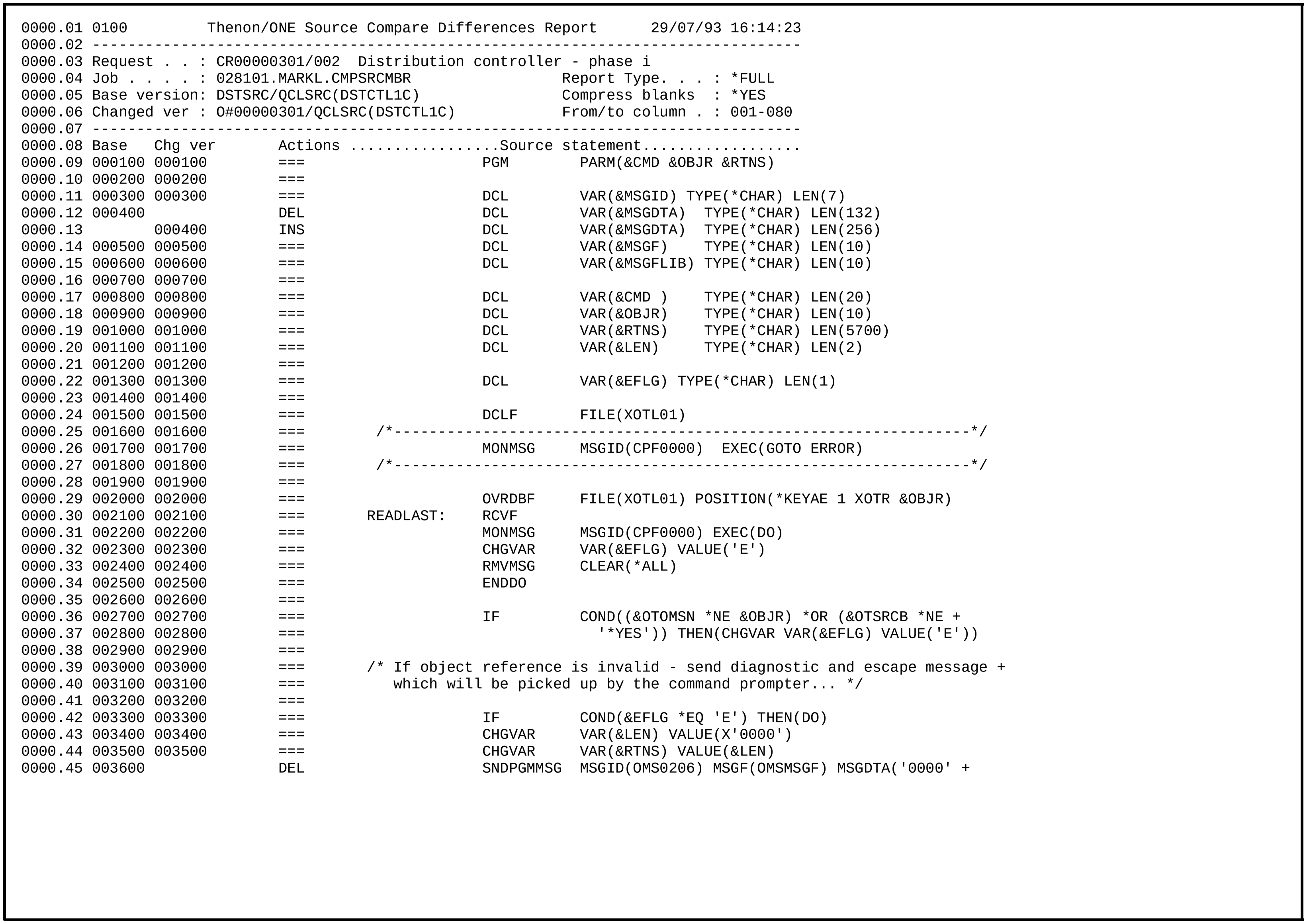
- After pressing enter, the submit window appears allowing you to execute the compare, either in batch (the default) or interactively.

- Once the compare job has completed, you can view the compare report by selecting **option 48** against DSTCTL1C in function WRKCROBJ:



```
      Actions      Text         Status       Exit         Help       028121 *NO
_______________________________________________________________________________
OMS250C1 TSPLSYD         Work with Change Request Objects     CR Lib O#00000301

25=Find string  26=Batch Find   47=Exc Cmp/Mrg  48=Cmp/Mrg Rept 49=Mrg Composit
69=Config opt

Act_Object_(P)_Type_(P)___Attr_(P)___Level____Ver_Status________Additional_Info
__  __________ __________ __________
48  DSTCTL1C   *PGM       CLP        *BAS     002 *CHG 29/07/93  Obj not in CR
49  DST001     *PGM       RPG        *BAS     002 *ECD 26/07/93 Dist: Obj
```

- The following pages show the generated compare report, and the associated Base and Changed source members:



```
0000.01 0100         Thenon/ONE Source Compare Differences Report      29/07/93 16:14:23
0000.02 --------------------------------------------------------------------------------
0000.03 Request . . : CR00000301/002  Distribution controller - phase i
0000.04 Job . . . . : 028101.MARKL.CMPSRCMBR                Report Type. . . : *FULL
0000.05 Base version: DSTSRC/QCLSRC(DSTCTL1C)               Compress blanks  : *YES
0000.06 Changed ver : O#00000301/QCLSRC(DSTCTL1C)           From/to column . : 001-080
0000.07 --------------------------------------------------------------------------------
0000.08 Base   Chg ver       Actions ................Source statement.................
0000.09 000100 000100        ===                    PGM        PARM(&CMD &OBJR &RTNS)
0000.10 000200 000200        ===
0000.11 000300 000300        ===                    DCL        VAR(&MSGID) TYPE(*CHAR) LEN(7)
0000.12 000400               DEL                    DCL        VAR(&MSGDTA)  TYPE(*CHAR) LEN(132)
0000.13        000400        INS                    DCL        VAR(&MSGDTA)  TYPE(*CHAR) LEN(256)
0000.14 000500 000500        ===                    DCL        VAR(&MSGF)    TYPE(*CHAR) LEN(10)
0000.15 000600 000600        ===                    DCL        VAR(&MSGFLIB) TYPE(*CHAR) LEN(10)
0000.16 000700 000700        ===
0000.17 000800 000800        ===                    DCL        VAR(&CMD )    TYPE(*CHAR) LEN(20)
0000.18 000900 000900        ===                    DCL        VAR(&OBJR)    TYPE(*CHAR) LEN(10)
0000.19 001000 001000        ===                    DCL        VAR(&RTNS)    TYPE(*CHAR) LEN(5700)
0000.20 001100 001100        ===                    DCL        VAR(&LEN)     TYPE(*CHAR) LEN(2)
0000.21 001200 001200        ===
0000.22 001300 001300        ===                    DCL        VAR(&EFLG) TYPE(*CHAR) LEN(1)
0000.23 001400 001400        ===
0000.24 001500 001500        ===                    DCLF       FILE(XOTL01)
0000.25 001600 001600        ===        /*-----------------------------------------------------------------*/
0000.26 001700 001700        ===                    MONMSG     MSGID(CPF0000)  EXEC(GOTO ERROR)
0000.27 001800 001800        ===        /*-----------------------------------------------------------------*/
0000.28 001900 001900        ===
0000.29 002000 002000        ===                    OVRDBF     FILE(XOTL01) POSITION(*KEYAE 1 XOTR &OBJR)
0000.30 002100 002100        ===       READLAST:    RCVF
0000.31 002200 002200        ===                    MONMSG     MSGID(CPF0000) EXEC(DO)
0000.32 002300 002300        ===                    CHGVAR     VAR(&EFLG) VALUE('E')
0000.33 002400 002400        ===                    RMVMSG     CLEAR(*ALL)
0000.34 002500 002500        ===                    ENDDO
0000.35 002600 002600        ===
0000.36 002700 002700        ===                    IF         COND((&OTOMSN *NE &OBJR) *OR (&OTSRCB *NE +
0000.37 002800 002800        ===                                 '*YES')) THEN(CHGVAR VAR(&EFLG) VALUE('E'))
0000.38 002900 002900        ===
0000.39 003000 003000        ===       /* If object reference is invalid - send diagnostic and escape message +
0000.40 003100 003100        ===          which will be picked up by the command prompter... */
0000.41 003200 003200        ===
0000.42 003300 003300        ===                    IF         COND(&EFLG *EQ 'E') THEN(DO)
0000.43 003400 003400        ===                    CHGVAR     VAR(&LEN) VALUE(X'0000')
0000.44 003500 003500        ===                    CHGVAR     VAR(&RTNS) VALUE(&LEN)
0000.45 003600               DEL                    SNDPGMMSG  MSGID(OMS0206) MSGF(OMSMSGF) MSGDTA('0000' +
```

Compare operation: Compare report (continued)

```
0000.46 003700               DEL                                  || &OBJR) MSGTYPE(*DIAG)
0000.47        003600        INS                    SNDPGMMSG  MSGID(OMS0206) MSGF(OMSMSGF) +
0000.48        003700        INS                                  MSGDTA('(''0000'' || &OBJR !! ''0000'')') +
0000.49        003800        INS                                  MSGTYPE(*DIAG)
0000.50 003800 003900        ===                    SNDPGMMSG  MSGID(CPF0011) MSGF(QCPFMSG) MSGTYPE(*ESCAPE)
0000.51 003900 004000        ===                    GOTO       CMDLBL(RETURN)
0000.52 004000 004100        ===                    ENDDO
0000.53 004100 004200        ===
0000.54 004200 004300        ===       /* Construct the returned string with the existing source file name */
0000.55 004300 004400        ===
0000.56 004400 004500        ===                    CHGVAR     VAR(&LEN) VALUE(X'01FF')
0000.57 004500 004600        ===                    CHGVAR     VAR(&RTNS) VALUE(&LEN || '??DFTSRCF(' || +
0000.58 004600 004700        ===                                  &OTSRCF || ') ??SRCUSG(' || &OTXXX5 || ') +
0000.59 004700 004800        ===                                  ??FRMCOL(' || %SST(&OTXXX7 1 3) || ') +
0000.60 004800 004900        ===                                  ??TOCOL(' || %SST(&OTXXX7 4 3) || ') +
0000.61 004900 005000        ===                                  ??PREFIX(''' || %SST(&OTXXX8 1 7) || ''') +
0000.62 005000 005100        ===                                  ??SUFFIX(''' || %SST(&OTXXX9 1 7) || ''')')
0000.63 005100 005200        ===
0000.64 006600 005300        TO                     DLTOVR     FILE(XOTL01)
0000.65 006700 005400        TO                     MONMSG     MSGID(CPF0000)
0000.66 006800 005500        TO
0000.67 005200 005600        ===                    GOTO       CMDLBL(RETURN)
0000.68 005300 005700        ===        /*-----------------------------------------------------------------*/
0000.69 005400 005800        ===        ERROR:
0000.70 005500 005900        ===        RCVMSG:     RCVMSG     MSGDTA(&MSGDTA) MSGID(&MSGID) MSGF(&MSGF) +
0000.71 005600 006000        ===                                  MSGFLIB(&MSGFLIB)
0000.72 005700 006100        ===
0000.73 005800 006200        ===                    IF         COND(&MSGID *EQ ' ') THEN(GOTO RETURN)
0000.74 005900 006300        ===
0000.75 006000 006400        ===                    SNDPGMMSG  MSGID(&MSGID) MSGF(&MSGFLIB/&MSGF) +
0000.76 006100 006500        ===                                  MSGDTA(&MSGDTA)
0000.77 006200 006600        ===                    GOTO       CMDLBL(RCVMSG)
0000.78 006300 006700        ===
0000.79 006400 006800        ===
0000.80 006500 006900        ===       RETURN:
0000.81 006600 005300        MOV                    DLTOVR     FILE(XOTL01)
0000.82 006700 005400        MOV                    MONMSG     MSGID(CPF0000)
0000.83 006800 005500        MOV
0000.84        007000        INS
0000.85 006900 007100        ===                    ENDPGM
```

Compare operations: Base version

```
             PGM        PARM(&CMD &OBJR &RTNS)                                       930310
                                                                                     930310
             DCL        VAR(&MSGID) TYPE(*CHAR) LEN(7)                               930310
             DCL        VAR(&MSGDTA)  TYPE(*CHAR) LEN(132)                           930310
             DCL        VAR(&MSGF)    TYPE(*CHAR) LEN(10)                            930310
             DCL        VAR(&MSGFLIB) TYPE(*CHAR) LEN(10)                            930310
                                                                                     930310
             DCL        VAR(&CMD )    TYPE(*CHAR) LEN(20)                            930310
             DCL        VAR(&OBJR)    TYPE(*CHAR) LEN(10)                            930310
             DCL        VAR(&RTNS)    TYPE(*CHAR) LEN(5700)                          930310
             DCL        VAR(&LEN)     TYPE(*CHAR) LEN(2)                             930310
                                                                                     930310
             DCL        VAR(&EFLG) TYPE(*CHAR) LEN(1)                                930709
                                                                                     930709
             DCLF       FILE(XOTL01)                                                 930709
 /*-------------------------------------------------------------------*/            930310
             MONMSG     MSGID(CPF0000)  EXEC(GOTO ERROR)                             930310
 /*-------------------------------------------------------------------*/            930310
                                                                                     930709
             OVRDBF     FILE(XOTL01) POSITION(*KEYAE 1 XOTR &OBJR)                   930709
READLAST:    RCVF                                                                    930709
             MONMSG     MSGID(CPF0000) EXEC(DO)                                      930709
             CHGVAR     VAR(&EFLG) VALUE('E')                                        930709
             RMVMSG     CLEAR(*ALL)                                                  930709
             ENDDO                                                                   930709
                                                                                     930709
             IF         COND((&OTOMSN *NE &OBJR) *OR (&OTSRCB *NE +                  930709
                          '*YES')) THEN(CHGVAR VAR(&EFLG) VALUE('E'))                930709
                                                                                     930310
/* If object reference is invalid - send diagnostic and escape message +             930310
   which will be picked up by the command prompter... */                             930310
                                                                                     930310
             IF         COND(&EFLG *EQ 'E') THEN(DO)                                 930709
             CHGVAR     VAR(&LEN) VALUE(X'0000')                                     930310
             CHGVAR     VAR(&RTNS) VALUE(&LEN)                                       930310
             SNDPGMMSG  MSGID(OMS0206) MSGF(OMSMSGF) MSGDTA('0000' +                 930310
                          || &OBJR) MSGTYPE(*DIAG)                                   930310
             SNDPGMMSG  MSGID(CPF0011) MSGF(QCPFMSG) MSGTYPE(*ESCAPE)                930310
             GOTO       CMDLBL(RETURN)                                               930310
             ENDDO                                                                   930310
                                                                                     930310
/* Construct the returned string with the existing source file name */               930310
                                                                                     930310
             CHGVAR     VAR(&LEN) VALUE(X'01FF')                                     930709
             CHGVAR     VAR(&RTNS) VALUE(&LEN || '??DFTSRCF(' || +                   930709
```

Compare operation: Base version (continued)

```
0046.00                           &OTSRCF || ') ??SRCUSG(' || &OTXXX5 || ') +                930709
0047.00                           ??FRMCOL(' || %SST(&OTXXX7 1 3) || ') +                    930709
0048.00                           ??TOCOL(' || %SST(&OTXXX7 4 3) || ') +                     930709
0049.00                           ??PREFIX(''' || %SST(&OTXXX8 1 7) || ''') +                930709
0050.00                           ??SUFFIX(''' || %SST(&OTXXX9 1 7) || ''')')                930709
0051.00                                                                                      930310
0052.00              GOTO       CMDLBL(RETURN)                                               930310
0053.00  /*-----------------------------------------------------------------*/               930310
0054.00  ERROR:                                                                              930310
0055.00  RCVMSG:     RCVMSG     MSGDTA(&MSGDTA) MSGID(&MSGID) MSGF(&MSGF) +                  930310
0056.00                           MSGFLIB(&MSGFLIB)                                          930310
0057.00                                                                                      930310
0058.00              IF         COND(&MSGID *EQ ' ') THEN(GOTO RETURN)                       930310
0059.00                                                                                      930310
0060.00              SNDPGMMSG  MSGID(&MSGID) MSGF(&MSGFLIB/&MSGF) +                         930310
0061.00                           MSGDTA(&MSGDTA)                                            930310
0062.00              GOTO       CMDLBL(RCVMSG)                                               930310
0063.00                                                                                      930310
0064.00                                                                                      930310
0065.00 RETURN:                                                                              930310
0066.00              DLTOVR     FILE(XOTL01)                                                 930709
0067.00              MONMSG     MSGID(CPF0000)                                               930709
0068.00                                                                                      930709
0069.00              ENDPGM                                                                  930310
```

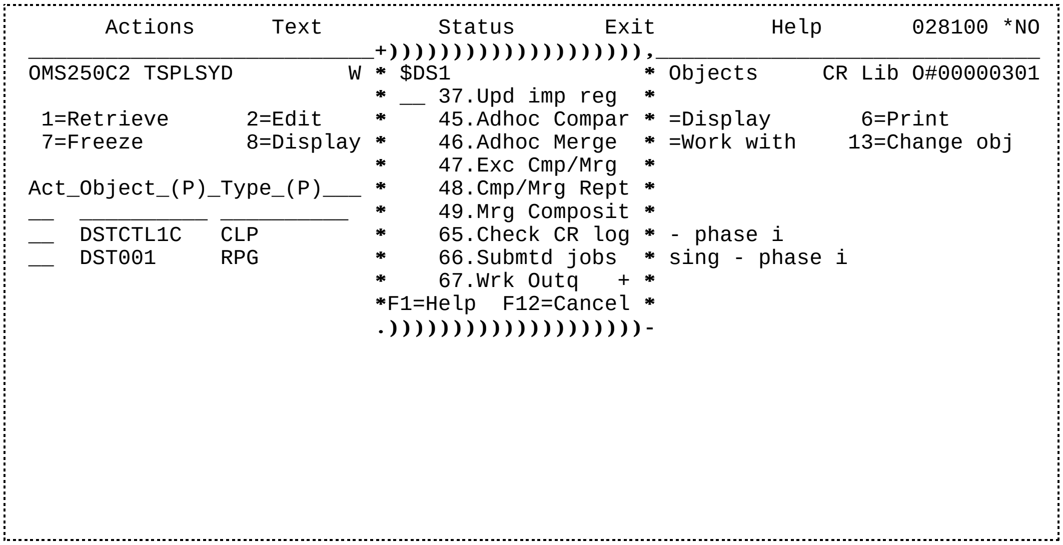
Compare operation: Changed version

```
0001.00             PGM        PARM(&CMD &OBJR &RTNS)                                      930310
0002.00                                                                                    930310
0003.00             DCL        VAR(&MSGID) TYPE(*CHAR) LEN(7)                              930310
0004.00             DCL        VAR(&MSGDTA)  TYPE(*CHAR) LEN(256)                          930729
0005.00             DCL        VAR(&MSGF)    TYPE(*CHAR) LEN(10)                           930310
0006.00             DCL        VAR(&MSGFLIB) TYPE(*CHAR) LEN(10)                           930310
0007.00                                                                                    930310
0008.00             DCL        VAR(&CMD )    TYPE(*CHAR) LEN(20)                           930310
0009.00             DCL        VAR(&OBJR)    TYPE(*CHAR) LEN(10)                           930310
0010.00             DCL        VAR(&RTNS)    TYPE(*CHAR) LEN(5700)                         930310
0011.00             DCL        VAR(&LEN)     TYPE(*CHAR) LEN(2)                            930310
0012.00                                                                                    930310
0013.00             DCL        VAR(&EFLG) TYPE(*CHAR) LEN(1)                               930709
0014.00                                                                                    930709
0015.00             DCLF       FILE(XOTL01)                                                930709
0016.00  /*-----------------------------------------------------------------*/             930310
0017.00             MONMSG     MSGID(CPF0000)  EXEC(GOTO ERROR)                            930310
0018.00  /*-----------------------------------------------------------------*/             930310
0019.00                                                                                    930709
0020.00             OVRDBF     FILE(XOTL01) POSITION(*KEYAE 1 XOTR &OBJR)                  930709
0021.00 READLAST:   RCVF                                                                   930709
0022.00             MONMSG     MSGID(CPF0000) EXEC(DO)                                     930709
0023.00             CHGVAR     VAR(&EFLG) VALUE('E')                                       930709
0024.00             RMVMSG     CLEAR(*ALL)                                                 930709
0025.00             ENDDO                                                                  930709
0026.00                                                                                    930709
0027.00             IF         COND((&OTOMSN *NE &OBJR) *OR (&OTSRCB *NE +                 930709
0028.00                          '*YES')) THEN(CHGVAR VAR(&EFLG) VALUE('E'))               930709
0029.00                                                                                    930310
0030.00 /* If object reference is invalid - send diagnostic and escape message +           930310
0031.00    which will be picked up by the command prompter... */                           930310
0032.00                                                                                    930310
0033.00             IF         COND(&EFLG *EQ 'E') THEN(DO)                                930709
0034.00             CHGVAR     VAR(&LEN) VALUE(X'0000')                                    930310
0035.00             CHGVAR     VAR(&RTNS) VALUE(&LEN)                                      930310
0036.00             SNDPGMMSG  MSGID(OMS0206) MSGF(OMSMSGF) +                              930729
0037.00                          MSGDTA('(''0000'' || &OBJR !! ''0000'')') +               930729
0038.00                          MSGTYPE(*DIAG)                                            930729
0039.00             SNDPGMMSG  MSGID(CPF0011) MSGF(QCPFMSG) MSGTYPE(*ESCAPE)               930310
0040.00             GOTO       CMDLBL(RETURN)                                              930310
0041.00             ENDDO                                                                  930310
0042.00                                                                                    930310
0043.00 /* Construct the returned string with the existing source file name */             930310
0044.00                                                                                    930310
0045.00             CHGVAR     VAR(&LEN) VALUE(X'01FF')                                    930709
```

Compare operation: Changed version (continued)

```
0046.00              CHGVAR     VAR(&RTNS) VALUE(&LEN || '??DFTSRCF(' || +                     930709
0047.00                           &OTSRCF || ') ??SRCUSG(' || &OTXXX5 || ') +                  930709
0048.00                           ??FRMCOL(' || %SST(&OTXXX7 1 3) || ') +                      930709
0049.00                           ??TOCOL(' || %SST(&OTXXX7 4 3) || ') +                       930709
0050.00                           ??PREFIX(''' || %SST(&OTXXX8 1 7) || ''') +                  930709
0051.00                           ??SUFFIX(''' || %SST(&OTXXX9 1 7) || ''')')                  930709
0052.00                                                                                        930310
0053.00              DLTOVR     FILE(XOTL01)                                                   930729
0054.00              MONMSG     MSGID(CPF0000)                                                 930729
0055.00                                                                                        930729
0056.00              GOTO       CMDLBL(RETURN)                                                 930310
0057.00  /*-----------------------------------------------------------------*/                930310
0058.00  ERROR:                                                                                930310
0059.00  RCVMSG:     RCVMSG     MSGDTA(&MSGDTA) MSGID(&MSGID) MSGF(&MSGF) +                    930310
0060.00                           MSGFLIB(&MSGFLIB)                                            930310
0061.00                                                                                        930310
0062.00              IF         COND(&MSGID *EQ ' ') THEN(GOTO RETURN)                         930310
0063.00                                                                                        930310
0064.00              SNDPGMMSG  MSGID(&MSGID) MSGF(&MSGFLIB/&MSGF) +                           930310
0065.00                           MSGDTA(&MSGDTA)                                              930310
0066.00              GOTO       CMDLBL(RCVMSG)                                                 930310
0067.00                                                                                        930310
0068.00                                                                                        930310
0069.00 RETURN:                                                                                930310
0070.00                                                                                        930709
0071.00              ENDPGM                                                                    930310
```

- If you have executed compare operations for a number of source members in the current CR, you can re-execute all of them in one job, by selecting **option 47** from the status window in function WRKCROBJ:

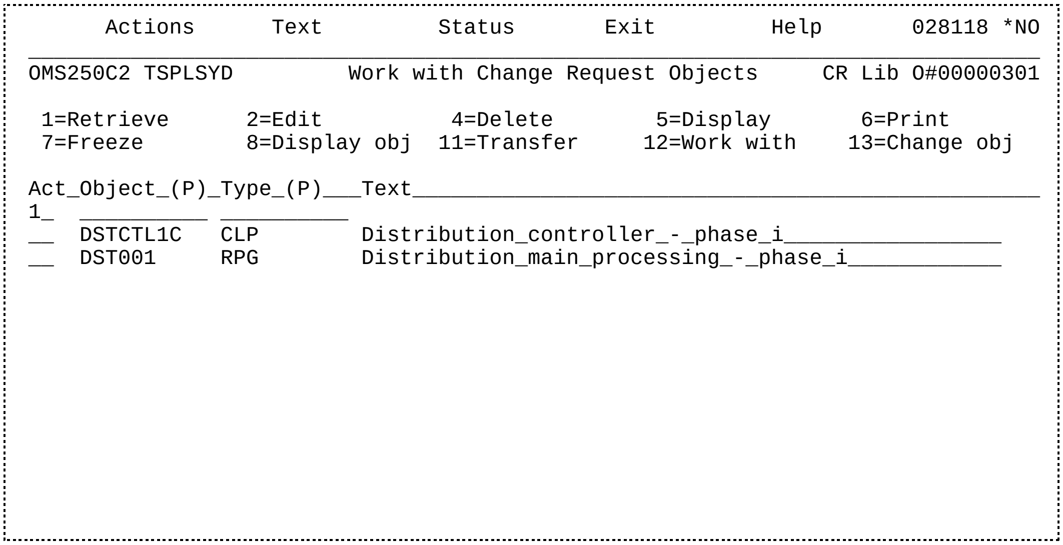


```
      Actions      Text         Status       Exit         Help       028100 *NO
___________________________+)))))))))))))))))))),_______________________________
OMS250C2 TSPLSYD         W * $DS1               * Objects     CR Lib O#00000301
                           * __ 37.Upd imp reg  *
 1=Retrieve      2=Edit    *    45.Adhoc Compar * =Display       6=Print
 7=Freeze        8=Display *    46.Adhoc Merge  * =Work with    13=Change obj
                           *    47.Exc Cmp/Mrg  *
Act_Object_(P)_Type_(P)___ *    48.Cmp/Mrg Rept *
__  __________ __________  *    49.Mrg Composit *
__  DSTCTL1C   CLP         *    65.Check CR log * - phase i
__  DST001     RPG         *    66.Submtd jobs  * sing - phase i
                           *    67.Wrk Outq   + *
                           *F1=Help  F12=Cancel *
                           .))))))))))))))))))))-
```

- Command EXCMRGRQS will be prompted, and you can select to re-execute a specific compare or merge operation, to re-execute the last operation, or to re-execute all compare and merge operations which were logged under the current CR execution request.

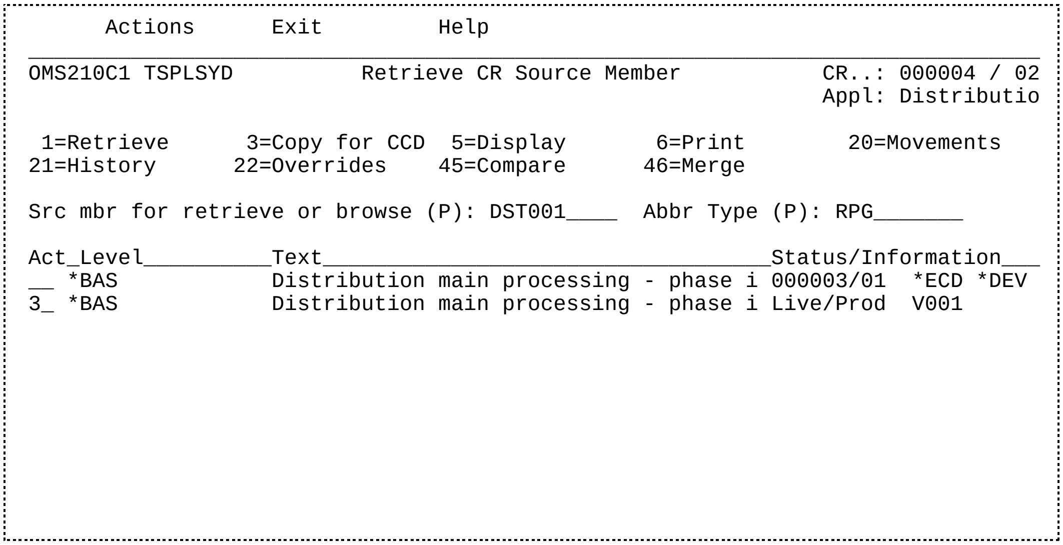
Appendix D: Example of a merge operation

In the following example of a merge operation, program DST001 was retrieved into CR 000003/01 for the purpose of long term development:



```
      Actions      Text         Status       Exit         Help       028118 *NO
_______________________________________________________________________________
OMS250C2 TSPLSYD         Work with Change Request Objects     CR Lib O#00000301

 1=Retrieve      2=Edit         4=Delete        5=Display       6=Print
 7=Freeze        8=Display obj  11=Transfer     12=Work with    13=Change obj

Act_Object_(P)_Type_(P)___Text_________________________________________________
1_  __________ __________
__  DSTCTL1C   CLP        Distribution_controller_-_phase_i_________________
__  DST001     RPG        Distribution_main_processing_-_phase_i____________
```

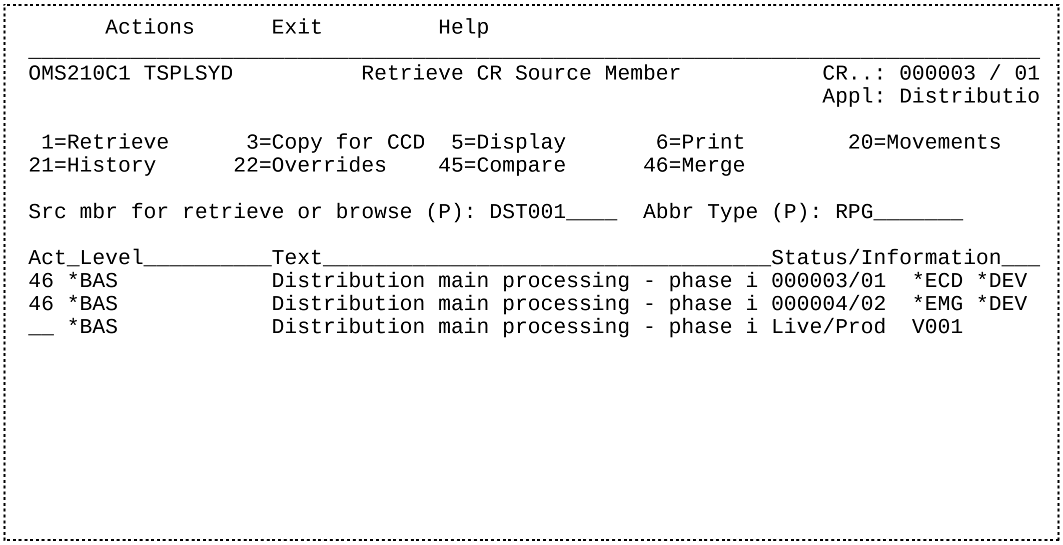
Subsequently, a bug was found in the current live/production version of DST001. A new Emergency Fix CR is created, CR 000004/02, and DST001 live/production version is retrieved for concurrent development:



```
      Actions      Exit         Help
_______________________________________________________________________________
OMS210C1 TSPLSYD          Retrieve CR Source Member           CR..: 000004 / 02
                                                              Appl: Distributio

 1=Retrieve      3=Copy for CCD  5=Display       6=Print        20=Movements
21=History      22=Overrides    45=Compare      46=Merge

Src mbr for retrieve or browse (P): DST001____  Abbr Type (P): RPG_______

Act_Level_________Text___________________________________Status/Information___
__ *BAS           Distribution main processing - phase i 000003/01  *ECD *DEV
3_ *BAS           Distribution main processing - phase i Live/Prod  V001
```

The bug has been fixed in CR 000004/02, and now you want to merge the emergency fix into the long term code changes in CR 000003/01.

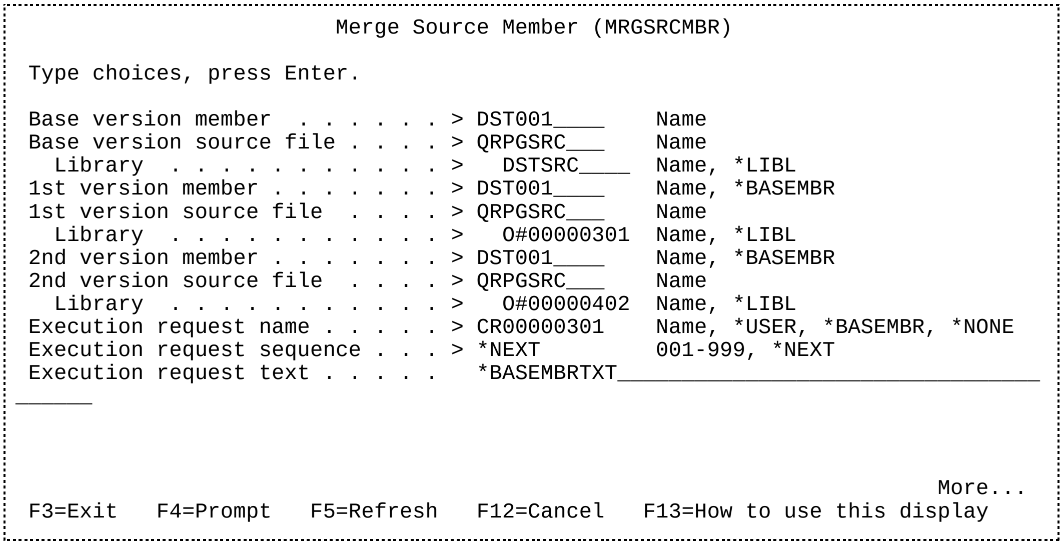
- In function WRKCROBJ type 1 on the top line, leaving the name and type with blanks. The panel for function RTVCRSRC appears:



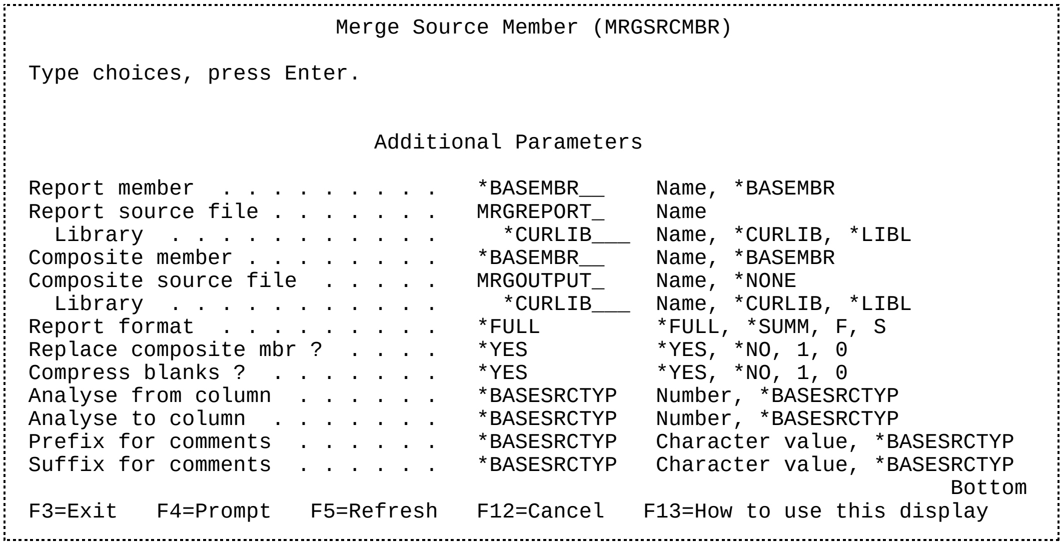
```
      Actions      Exit         Help
_______________________________________________________________________________
OMS210C1 TSPLSYD          Retrieve CR Source Member           CR..: 000003 / 01
                                                              Appl: Distributio

 1=Retrieve      3=Copy for CCD  5=Display       6=Print        20=Movements
21=History      22=Overrides    45=Compare      46=Merge

Src mbr for retrieve or browse (P): DST001____  Abbr Type (P): RPG_______

Act_Level__________Text___________________________________Status/Information___
46 *BAS            Distribution main processing - phase i 000003/01  *ECD *DEV
46 *BAS            Distribution main processing - phase i 000004/02  *EMG *DEV
__ *BAS            Distribution main processing - phase i Live/Prod  V001
```

- Type the program name and type DST001 / RPG to show the development status for the program. Three entries are shown: the first is for the long term development version in CR 000003/01; the second is the emergency fix development in CR 000004/02; the third is for current live/production version.

- Select both the long term development version and the emergency fix version, using the Merge **option 46**. Command MRGSRCMBR is prompted, with all parameters set to reflect your selection:



```
                        Merge Source Member (MRGSRCMBR)

 Type choices, press Enter.

 Base version member  . . . . . . > DST001____    Name
 Base version source file . . . . > QRPGSRC___    Name
   Library  . . . . . . . . . . . >   DSTSRC____  Name, *LIBL
 1st version member . . . . . . . > DST001____    Name, *BASEMBR
 1st version source file  . . . . > QRPGSRC___    Name
   Library  . . . . . . . . . . . >   O#00000301  Name, *LIBL
 2nd version member . . . . . . . > DST001____    Name, *BASEMBR
 2nd version source file  . . . . > QRPGSRC___    Name
   Library  . . . . . . . . . . . >   O#00000402  Name, *LIBL
 Execution request name . . . . . > CR00000301    Name, *USER, *BASEMBR, *NONE
 Execution request sequence . . . > *NEXT         001-999, *NEXT
 Execution request text . . . . .   *BASEMBRTXT_________________________________
______




                                                                       More...
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
```

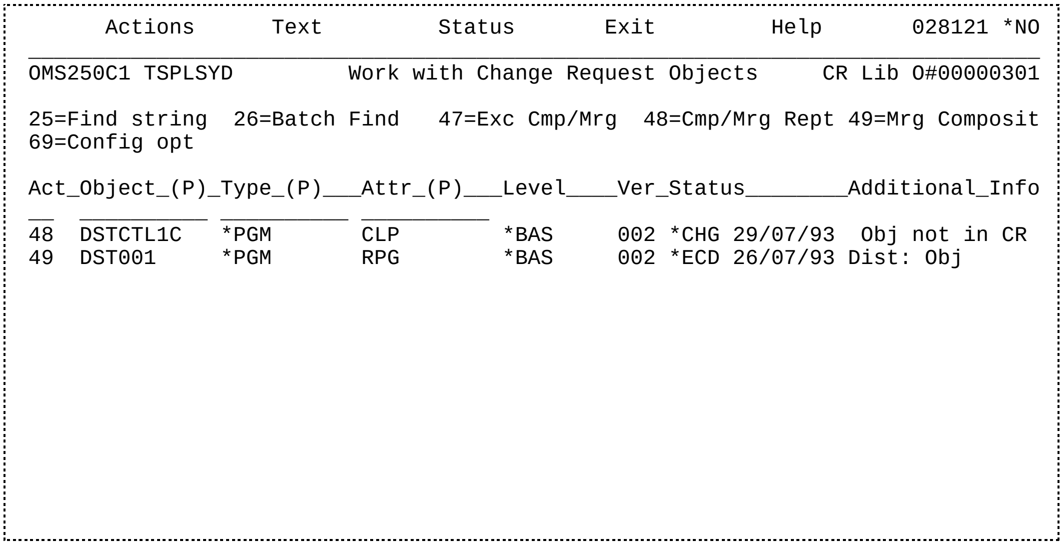


```
                        Merge Source Member (MRGSRCMBR)

 Type choices, press Enter.


                               Additional Parameters

 Report member  . . . . . . . . .   *BASEMBR__    Name, *BASEMBR
 Report source file . . . . . . .   MRGREPORT_    Name
   Library  . . . . . . . . . . .     *CURLIB___  Name, *CURLIB, *LIBL
 Composite member . . . . . . . .   *BASEMBR__    Name, *BASEMBR
 Composite source file  . . . . .   MRGOUTPUT_    Name, *NONE
   Library  . . . . . . . . . . .     *CURLIB___  Name, *CURLIB, *LIBL
 Report format  . . . . . . . . .   *FULL         *FULL, *SUMM, F, S
 Replace composite mbr ?  . . . .   *YES          *YES, *NO, 1, 0
 Compress blanks ?  . . . . . . .   *YES          *YES, *NO, 1, 0
 Analyse from column  . . . . . .   *BASESRCTYP   Number, *BASESRCTYP
 Analyse to column  . . . . . . .   *BASESRCTYP   Number, *BASESRCTYP
 Prefix for comments  . . . . . .   *BASESRCTYP   Character value, *BASESRCTYP
 Suffix for comments  . . . . . .   *BASESRCTYP   Character value, *BASESRCTYP
                                                                        Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
```

- Note that the merge report default is member name DST0001 in source file MRGREPORT, and that the merge composite default is member name DST001 is source file MRGOUTPUT **-** both in your CR library. Also note that the compare operation will be logged under execution request CR 00000301, so after generation you can access the report via **option 48**, and the composite output via **option 49** in function WRKCROBJ.

- After pressing enter, the submit window appears allowing you to execute the merge, either in batch (the default) or interactively.

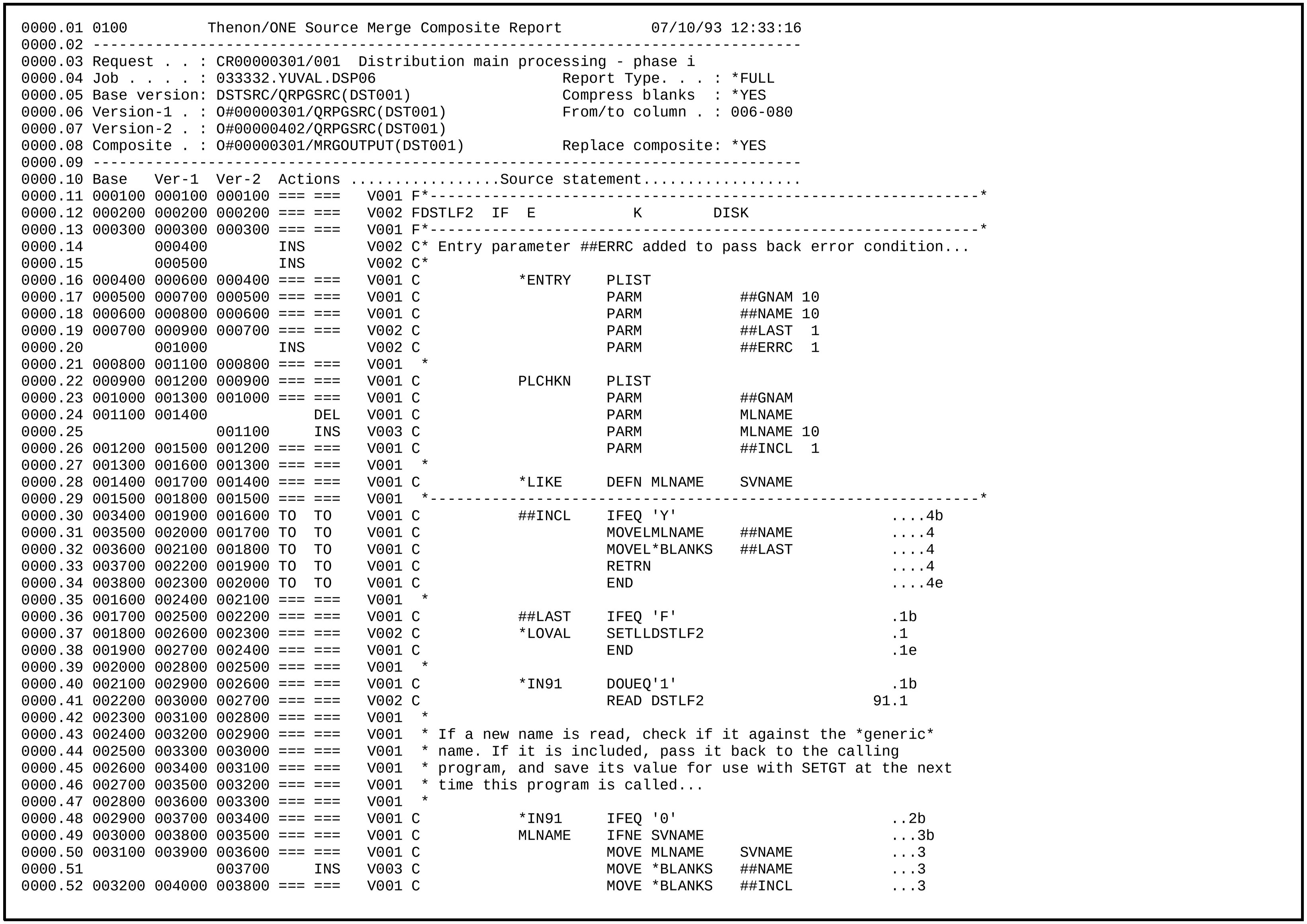
- Once the merge job has completed, you can view the merge report by selecting **option 48**, or the merge composite output by selecting **option 49** against DST001 in function WRKCROBJ:



```
      Actions      Text         Status       Exit         Help       028121 *NO
________________________________________________________________________________
OMS250C1 TSPLSYD         Work with Change Request Objects     CR Lib O#00000301

25=Find string  26=Batch Find   47=Exc Cmp/Mrg  48=Cmp/Mrg Rept 49=Mrg Composit
69=Config opt

Act_Object_(P)_Type_(P)___Attr_(P)___Level____Ver_Status________Additional_Info
___  __________ __________ __________
48  DSTCTL1C   *PGM       CLP        *BAS      002 *CHG 29/07/93  Obj not in CR
49  DST001     *PGM       RPG        *BAS      002 *ECD 26/07/93 Dist: Obj
```

- The following pages show the generated merge report and composite output, and the associated Base, 1st Version and 2nd Version source members:

Merge operation: Merge report



```
0000.01 0100         Thenon/ONE Source Merge Composite Report          07/10/93 12:33:16
0000.02 --------------------------------------------------------------------------------
0000.03 Request . . : CR00000301/001  Distribution main processing - phase i
0000.04 Job . . . . : 033332.YUVAL.DSP06                     Report Type. . . : *FULL
0000.05 Base version: DSTSRC/QRPGSRC(DST001)                 Compress blanks  : *YES
0000.06 Version-1 . : O#00000301/QRPGSRC(DST001)             From/to column . : 006-080
0000.07 Version-2 . : O#00000402/QRPGSRC(DST001)
0000.08 Composite . : O#00000301/MRGOUTPUT(DST001)           Replace composite: *YES
0000.09 --------------------------------------------------------------------------------
0000.10 Base   Ver-1  Ver-2  Actions .................Source statement..................
0000.11 000100 000100 000100 === ===   V001 F*--------------------------------------------------------------*
0000.12 000200 000200 000200 === ===   V002 FDSTLF2  IF  E           K        DISK
0000.13 000300 000300 000300 === ===   V001 F*--------------------------------------------------------------*
0000.14        000400        INS       V002 C* Entry parameter ##ERRC added to pass back error condition...
0000.15        000500        INS       V002 C*
0000.16 000400 000600 000400 === ===   V001 C           *ENTRY    PLIST
0000.17 000500 000700 000500 === ===   V001 C                     PARM           ##GNAM 10
0000.18 000600 000800 000600 === ===   V001 C                     PARM           ##NAME 10
0000.19 000700 000900 000700 === ===   V002 C                     PARM           ##LAST  1
0000.20        001000        INS       V002 C                     PARM           ##ERRC  1
0000.21 000800 001100 000800 === ===   V001  *
0000.22 000900 001200 000900 === ===   V001 C           PLCHKN    PLIST
0000.23 001000 001300 001000 === ===   V001 C                     PARM           ##GNAM
0000.24 001100 001400            DEL   V001 C                     PARM           MLNAME
0000.25               001100     INS   V003 C                     PARM           MLNAME 10
0000.26 001200 001500 001200 === ===   V001 C                     PARM           ##INCL  1
0000.27 001300 001600 001300 === ===   V001  *
0000.28 001400 001700 001400 === ===   V001 C           *LIKE     DEFN MLNAME    SVNAME
0000.29 001500 001800 001500 === ===   V001  *--------------------------------------------------------------*
0000.30 003400 001900 001600 TO  TO    V001 C           ##INCL    IFEQ 'Y'                         ....4b
0000.31 003500 002000 001700 TO  TO    V001 C                     MOVELMLNAME    ##NAME            ....4
0000.32 003600 002100 001800 TO  TO    V001 C                     MOVEL*BLANKS   ##LAST            ....4
0000.33 003700 002200 001900 TO  TO    V001 C                     RETRN                            ....4
0000.34 003800 002300 002000 TO  TO    V001 C                     END                              ....4e
0000.35 001600 002400 002100 === ===   V001  *
0000.36 001700 002500 002200 === ===   V001 C           ##LAST    IFEQ 'F'                         .1b
0000.37 001800 002600 002300 === ===   V002 C           *LOVAL    SETLLDSTLF2                      .1
0000.38 001900 002700 002400 === ===   V001 C                     END                              .1e
0000.39 002000 002800 002500 === ===   V001  *
0000.40 002100 002900 002600 === ===   V001 C           *IN91     DOUEQ'1'                         .1b
0000.41 002200 003000 002700 === ===   V002 C                     READ DSTLF2                    91.1
0000.42 002300 003100 002800 === ===   V001  *
0000.43 002400 003200 002900 === ===   V001  * If a new name is read, check if it against the *generic*
0000.44 002500 003300 003000 === ===   V001  * name. If it is included, pass it back to the calling
0000.45 002600 003400 003100 === ===   V001  * program, and save its value for use with SETGT at the next
0000.46 002700 003500 003200 === ===   V001  * time this program is called...
0000.47 002800 003600 003300 === ===   V001  *
0000.48 002900 003700 003400 === ===   V001 C           *IN91     IFEQ '0'                         ..2b
0000.49 003000 003800 003500 === ===   V001 C           MLNAME    IFNE SVNAME                      ...3b
0000.50 003100 003900 003600 === ===   V001 C                     MOVE MLNAME    SVNAME            ...3
0000.51               003700     INS   V003 C                     MOVE *BLANKS   ##NAME            ...3
0000.52 003200 004000 003800 === ===   V001 C                     MOVE *BLANKS   ##INCL            ...3
```

Merge operation: Merge report (continued)

```
0000.56 003600 002100 001800 MOV MOV   V001 C                     MOVEL*BLANKS   ##LAST           ....4
0000.57 003700 002200 001900 MOV MOV   V001 C                     RETRN                           ....4
0000.58 003800 002300 002000 MOV MOV   V001 C                     END                             ....4e
0000.59 003900 004200 004000 === ===   V001 C                     END                             ...3e
0000.60 004000 004300 004100 === ===   V001  *
0000.61 004100 004400 004200 === ===   V001  * If end of file reached, free program CHKNAM, and return
0000.62 004200 004500 004300 === ===   V001  * 'L' in ##LAST to indicate to the calling programs that end
0000.63 004300 004600 004400 === ===   V001  * of file has been reached.
0000.64        004700        INS       V002  * Also, initialise error code output parameter.
0000.65 004400 004800 004500 === ===   V001  *
0000.66 004500 004900 004600 === ===   V001 C                     ELSE                            ..2x
0000.67 004600 005000 004700 === ===   V002 C                     MOVEL'E'       ##INCL           ..2
0000.68        005100        INS       V002 C                     MOVEL'E'       ##ERRC           ..2
0000.69 004700 005200 004800 === ===   V001 C                     CALL 'CHKNAM'  PLCHKN           ..2
0000.70 004800 005300 004900 === ===   V001 C                     MOVEL*BLANKS   ##NAME           ..2
0000.71 004900 005400 005000 === ===   V001 C                     MOVEL'L'       ##LAST           ..2
0000.72                                     /* ! Review Suggested: Begin !*/
0000.73        005500 005200 INS>MOV *       *
0000.74        005600 005300 INS>MOV *       * Free lower invocations....
0000.75        005700 005400 INS>MOV *       *
0000.76        005800 005500 INS>MOV *      C                     FREE PGMA
0000.77        005900 005600 INS>MOV *      C                     FREE PGMB
0000.78                                     /* ! Review Suggested: End   !*/
0000.79        006000        INS            C                     FREE PGMX
0000.80                                     /* ! Review Suggested: Begin !*/
0000.81        006100 005700 INS>MOV *      C                     FREE PGMC
0000.82        006200 005800 INS>MOV *      C                     FREE PGMD
0000.83                                     /* ! Review Suggested: End   !*/
0000.84 005000 006300            DEL   V001 C                     MOVE '1'       *INLR            ..2
0000.85 005100 006400 005100 === ===   V001 C                     END                             ..2e
0000.86        005500 005200 MOV<INS         *
0000.87        005600 005300 MOV<INS         * Free lower invocations....
0000.88        005700 005400 MOV<INS         *
0000.89        005800 005500 MOV<INS        C                     FREE PGMA
0000.90        005900 005600 MOV<INS        C                     FREE PGMB
0000.91        006100 005700 MOV<INS        C                     FREE PGMC
0000.92        006200 005800 MOV<INS        C                     FREE PGMD
0000.93               005900     INS   V003 C                     MOVE '1'       *INLR            .1
0000.94 005200 006500 006000 === ===   V001 C                     END                             .1e
```

Merge operation: Composite version

```
0000.01 V001 F*--------------------------------------------------------------*               000000
0000.02 V002 FDSTLF2  IF  E           K        DISK                                         000000
0000.03 V001 F*--------------------------------------------------------------*               000000
0000.04 V002 C* Entry parameter ##ERRC added to pass back error condition...                000000
0000.05 V002 C*                                                                             000000
0000.06 V001 C           *ENTRY    PLIST                                                    000000
0000.07 V001 C                     PARM           ##GNAM 10                                 000000
0000.08 V001 C                     PARM           ##NAME 10                                 000000
0000.09 V002 C                     PARM           ##LAST  1                                 000000
0000.10 V002 C                     PARM           ##ERRC  1                                 000000
0000.11 V001  *                                                                             000000
0000.12 V001 C           PLCHKN    PLIST                                                    000000
0000.13 V001 C                     PARM           ##GNAM                                    000000
0000.14 V003 C                     PARM           MLNAME 10                                 000000
0000.15 V001 C                     PARM           ##INCL  1                                 000000
0000.16 V001  *                                                                             000000
0000.17 V001 C           *LIKE     DEFN MLNAME    SVNAME                                    000000
0000.18 V001  *--------------------------------------------------------------*               000000
0000.19 V001 C           ##INCL    IFEQ 'Y'                          ....4b                 000000
0000.20 V001 C                     MOVELMLNAME    ##NAME             ....4                  000000
0000.21 V001 C                     MOVEL*BLANKS   ##LAST             ....4                  000000
0000.22 V001 C                     RETRN                             ....4                  000000
0000.23 V001 C                     END                               ....4e                 000000
0000.24 V001  *                                                                             000000
0000.25 V001 C           ##LAST    IFEQ 'F'                          .1b                    000000
0000.26 V002 C           *LOVAL    SETLLDSTLF2                       .1                     000000
0000.27 V001 C                     END                               .1e                    000000
0000.28 V001  *                                                                             000000
0000.29 V001 C           *IN91     DOUEQ'1'                          .1b                    000000
0000.30 V002 C                     READ DSTLF2                     91.1                     000000
0000.31 V001  *                                                                             000000
0000.32 V001  * If a new name is read, check if it against the *generic*                    000000
0000.33 V001  * name. If it is included, pass it back to the calling                        000000
0000.34 V001  * program, and save its value for use with SETGT at the next                  000000
0000.35 V001  * time this program is called...                                              000000
0000.36 V001  *                                                                             000000
0000.37 V001 C           *IN91     IFEQ '0'                          ..2b                   000000
0000.38 V001 C           MLNAME    IFNE SVNAME                       ...3b                  000000
0000.39 V001 C                     MOVE MLNAME    SVNAME             ...3                   000000
0000.40 V003 C                     MOVE *BLANKS   ##NAME             ...3                   000000
0000.41 V001 C                     MOVE *BLANKS   ##INCL             ...3                   000000
0000.42 V001 C                     CALL 'CHKNAM'  PLCHKN             ...3                   000000
0000.43 V001 C                     END                               ...3e                  000000
0000.44 V001  *                                                                             000000
0000.45 V001  * If end of file reached, free program CHKNAM, and return                     000000
```

Merge operation: Composite version (continued)

```
0000.46 V001  * 'L' in ##LAST to indicate to the calling programs that end                  000000
0000.47 V001  * of file has been reached.                                                   000000
0000.48 V002  * Also, initialise error code output parameter.                               000000
0000.49 V001  *                                                                             000000
0000.50 V001 C                     ELSE                            ..2x                     000000
0000.51 V002 C                     MOVEL'E'       ##INCL           ..2                      000000
0000.52 V002 C                     MOVEL'E'       ##ERRC           ..2                      000000
0000.53 V001 C                     CALL 'CHKNAM'  PLCHKN           ..2                      000000
0000.54 V001 C                     MOVEL*BLANKS   ##NAME           ..2                      000000
0000.55 V001 C                     MOVEL'L'       ##LAST           ..2                      000000
0000.56      /* ! Review Suggested: Begin !*/                                               000000
0000.57       *                                                                             000000
0000.58       * Free lower invocations....                                                  000000
0000.59       *                                                                             000000
0000.60      C                     FREE PGMA                                                000000
0000.61      C                     FREE PGMB                                                000000
0000.62      /* ! Review Suggested: End   !*/                                               000000
0000.63      C                     FREE PGMX                                                000000
0000.64      /* ! Review Suggested: Begin !*/                                               000000
0000.65      C                     FREE PGMC                                                000000
0000.66      C                     FREE PGMD                                                000000
0000.67      /* ! Review Suggested: End   !*/                                               000000
0000.68 V001 C                     END                             ..2e                     000000
0000.69 V003 C                     MOVE '1'       *INLR            .1                       000000
0000.70 V001 C                     END                             .1e                      000000
```

Merge operation: Base version

```
0001.00 V001 F*--------------------------------------------------------------*               930630
0002.00 V001 FDSTLF2  IF  E           K        DISK                                          930726
0003.00 V001 F*--------------------------------------------------------------*               930630
0004.00 V001 C           *ENTRY    PLIST                                                     930630
0005.00 V001 C                     PARM           ##GNAM 10                                  930630
0006.00 V001 C                     PARM           ##NAME 10                                  930630
0007.00 V001 C                     PARM           ##LAST  1                                  930630
0008.00 V001  *                                                                              930630
0009.00 V001 C           PLCHKN    PLIST                                                     930630
0010.00 V001 C                     PARM           ##GNAM                                     930630
0011.00 V001 C                     PARM           MLNAME                                     930630
0012.00 V001 C                     PARM           ##INCL  1                                  930630
0013.00 V001  *                                                                              930630
0014.00 V001 C           *LIKE     DEFN MLNAME    SVNAME                                     930630
0015.00 V001  *--------------------------------------------------------------*               930630
0016.00 V001  *                                                                              930630
0017.00 V001 C           ##LAST    IFEQ 'F'                        .1b                       930630
0018.00 V001 C           *LOVAL    SETLLDSTLF2                     .1                        930726
0019.00 V001 C                     END                             .1e                       930630
0020.00 V001  *                                                                              930630
0021.00 V001 C           *IN91     DOUEQ'1'                        .1b                       930630
0022.00 V001 C                     READ DSTLF2                   91.1                        930726
0023.00 V001  *                                                                              930630
0024.00 V001  * If a new name is read, check if it against the *generic*                     930630
0025.00 V001  * name. If it is included, pass it back to the calling                         930630
0026.00 V001  * program, and save its value for use with SETGT at the next                   930630
0027.00 V001  * time this program is called...                                               930630
0028.00 V001  *                                                                              930630
0029.00 V001 C           *IN91     IFEQ '0'                        ..2b                      930630
0030.00 V001 C           MLNAME    IFNE SVNAME                     ...3b                     930630
0031.00 V001 C                     MOVE MLNAME    SVNAME           ...3                      930630
0032.00 V001 C                     MOVE *BLANKS   ##INCL           ...3                      930630
0033.00 V001 C                     CALL 'CHKNAM'  PLCHKN           ...3                      930630
0034.00 V001 C           ##INCL    IFEQ 'Y'                        ....4b                    930630
0035.00 V001 C                     MOVELMLNAME    ##NAME           ....4                     930630
0036.00 V001 C                     MOVEL*BLANKS   ##LAST           ....4                     930630
0037.00 V001 C                     RETRN                           ....4                     930630
0038.00 V001 C                     END                             ....4e                    930630
0039.00 V001 C                     END                             ...3e                     930630
0040.00 V001  *                                                                              930630
0041.00 V001  * If end of file reached, free program CHKNAM, and return                      930630
0042.00 V001  * 'L' in ##LAST to indicate to the calling programs that end                   930630
0043.00 V001  * of file has been reached.                                                    930630
0044.00 V001  *                                                                              930630
0045.00 V001 C                     ELSE                            ..2x                      930630
```

Merge operation: Base version (continued)

```
0046.00 V001 C                     MOVEL'E'        ##INCL           ..2                        930630
0047.00 V001 C                     CALL 'CHKNAM'   PLCHKN           ..2                        930630
0048.00 V001 C                     MOVEL*BLANKS    ##NAME           ..2                        930630
0049.00 V001 C                     MOVEL'L'        ##LAST           ..2                        930630
0050.00 V001 C                     MOVE '1'        *INLR            ..2                        930630
0051.00 V001 C                     END                              ..2e                       930630
0052.00 V001 C                     END                              .1e                        930630
```

Merge operation: 1st version

```
0001.00 V001 F*--------------------------------------------------------------*               930630
0002.00 V002 FDSTLF2  IF  E           K        DISK                                          930726
0003.00 V001 F*--------------------------------------------------------------*               930630
0004.00 V002 C* Entry parameter ##ERRC added to pass back error condition...                 930728
0005.00 V002 C*                                                                              930728
0006.00 V001 C           *ENTRY    PLIST                                                     930630
0007.00 V001 C                     PARM           ##GNAM 10                                  930630
0008.00 V001 C                     PARM           ##NAME 10                                  930630
0009.00 V002 C                     PARM           ##LAST  1                                  930728
0010.00 V002 C                     PARM           ##ERRC  1                                  930728
0011.00 V001  *                                                                              930630
0012.00 V001 C           PLCHKN    PLIST                                                     930630
0013.00 V001 C                     PARM           ##GNAM                                     930630
0014.00 V001 C                     PARM           MLNAME                                     930630
0015.00 V001 C                     PARM           ##INCL  1                                  930630
0016.00 V001  *                                                                              930630
0017.00 V001 C           *LIKE     DEFN MLNAME    SVNAME                                     930630
0018.00 V001  *--------------------------------------------------------------*               930630
0019.00 V001 C           ##INCL    IFEQ 'Y'                        ....4b                    931007
0020.00 V001 C                     MOVELMLNAME    ##NAME           ....4                     931007
0021.00 V001 C                     MOVEL*BLANKS   ##LAST           ....4                     931007
0022.00 V001 C                     RETRN                           ....4                     931007
0023.00 V001 C                     END                             ....4e                    931007
0024.00 V001  *                                                                              930630
0025.00 V001 C           ##LAST    IFEQ 'F'                        .1b                       930630
0026.00 V002 C           *LOVAL    SETLLDSTLF2                     .1                        930726
0027.00 V001 C                     END                             .1e                       930630
0028.00 V001  *                                                                              930630
0029.00 V001 C           *IN91     DOUEQ'1'                        .1b                       930630
0030.00 V002 C                     READ DSTLF2                   91.1                        930726
0031.00 V001  *                                                                              930630
0032.00 V001  * If a new name is read, check if it against the *generic*                     930630
0033.00 V001  * name. If it is included, pass it back to the calling                         930630
0034.00 V001  * program, and save its value for use with SETGT at the next                   930630
0035.00 V001  * time this program is called...                                               930630
0036.00 V001  *                                                                              930630
0037.00 V001 C           *IN91     IFEQ '0'                        ..2b                      930630
0038.00 V001 C           MLNAME    IFNE SVNAME                     ...3b                     930630
0039.00 V001 C                     MOVE MLNAME    SVNAME           ...3                      930630
0040.00 V001 C                     MOVE *BLANKS   ##INCL           ...3                      930630
0041.00 V001 C                     CALL 'CHKNAM'  PLCHKN           ...3                      930630
0042.00 V001 C                     END                             ...3e                     930630
0043.00 V001  *                                                                              930630
0044.00 V001  * If end of file reached, free program CHKNAM, and return                      930630
0045.00 V001  * 'L' in ##LAST to indicate to the calling programs that end                   930630
```

Merge operation: 1st version (continued)

```
0046.00 V001  * of file has been reached.                                                    930630
0047.00 V002  * Also, initialise error code output parameter.                                930728
0048.00 V001  *                                                                              930630
0049.00 V001 C                     ELSE                            ..2x                      930630
0050.00 V002 C                     MOVEL'E'       ##INCL           ..2                       930728
0051.00 V002 C                     MOVEL'E'       ##ERRC           ..2                       930728
0052.00 V001 C                     CALL 'CHKNAM'  PLCHKN           ..2                       930630
0053.00 V001 C                     MOVEL*BLANKS   ##NAME           ..2                       930630
0054.00 V001 C                     MOVEL'L'       ##LAST           ..2                       930630
0055.00       *                                                                              931007
0056.00       * Free lower invocations....                                                   931007
0057.00       *                                                                              931007
0058.00      C                     FREE PGMA                                                 931007
0059.00      C                     FREE PGMB                                                 931007
0060.00      C                     FREE PGMX                                                 931007
0061.00      C                     FREE PGMC                                                 931007
0062.00      C                     FREE PGMD                                                 931007
0063.00 V001 C                     MOVE '1'       *INLR            ..2                       930630
0064.00 V001 C                     END                             ..2e                      930630
0065.00 V001 C                     END                             .1e                       930630
```

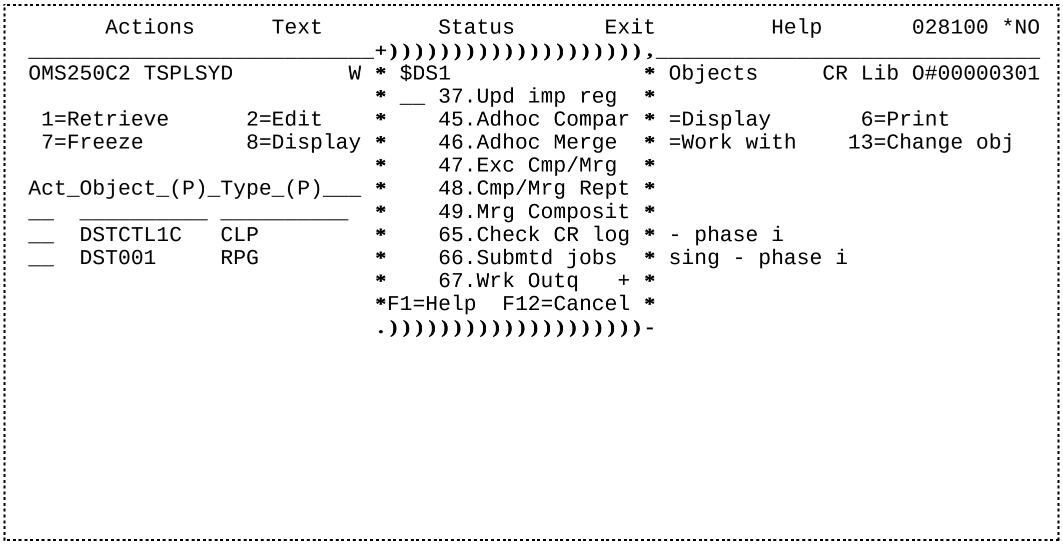
Merge operation: 2nd version

```
0001.00 V001 F*--------------------------------------------------------------*               930630
0002.00 V001 FDSTLF2  IF  E           K        DISK                                          930726
0003.00 V001 F*--------------------------------------------------------------*               930630
0004.00 V001 C           *ENTRY    PLIST                                                     930630
0005.00 V001 C                     PARM           ##GNAM 10                                  930630
0006.00 V001 C                     PARM           ##NAME 10                                  930630
0007.00 V001 C                     PARM           ##LAST  1                                  930630
0008.00 V001  *                                                                              930630
0009.00 V001 C           PLCHKN    PLIST                                                     930630
0010.00 V001 C                     PARM           ##GNAM                                     930630
0011.00 V003 C                     PARM           MLNAME 10                                  930903
0012.00 V001 C                     PARM           ##INCL  1                                  930630
0013.00 V001  *                                                                              930630
0014.00 V001 C           *LIKE     DEFN MLNAME    SVNAME                                     930630
0015.00 V001  *--------------------------------------------------------------*               930630
0016.00 V001 C           ##INCL    IFEQ 'Y'                        ....4b                    931007
0017.00 V001 C                     MOVELMLNAME    ##NAME           ....4                     931007
0018.00 V001 C                     MOVEL*BLANKS   ##LAST           ....4                     931007
0019.00 V001 C                     RETRN                           ....4                     931007
0020.00 V001 C                     END                             ....4e                    931007
0021.00 V001  *                                                                              930630
0022.00 V001 C           ##LAST    IFEQ 'F'                        .1b                       930630
0023.00 V001 C           *LOVAL    SETLLDSTLF2                     .1                        930726
0024.00 V001 C                     END                             .1e                       930630
0025.00 V001  *                                                                              930630
0026.00 V001 C           *IN91     DOUEQ'1'                        .1b                       930630
0027.00 V001 C                     READ DSTLF2                   91.1                        930726
0028.00 V001  *                                                                              930630
0029.00 V001  * If a new name is read, check if it against the *generic*                     930630
0030.00 V001  * name. If it is included, pass it back to the calling                         930630
0031.00 V001  * program, and save its value for use with SETGT at the next                   930630
0032.00 V001  * time this program is called...                                               930630
0033.00 V001  *                                                                              930630
0034.00 V001 C           *IN91     IFEQ '0'                        ..2b                      930630
0035.00 V001 C           MLNAME    IFNE SVNAME                     ...3b                     930630
0036.00 V001 C                     MOVE MLNAME    SVNAME           ...3                      930630
0037.00 V003 C                     MOVE *BLANKS   ##NAME           ...3                      930728
0038.00 V001 C                     MOVE *BLANKS   ##INCL           ...3                      930630
0039.00 V001 C                     CALL 'CHKNAM'  PLCHKN           ...3                      930630
0040.00 V001 C                     END                             ...3e                     930630
0041.00 V001  *                                                                              930630
0042.00 V001  * If end of file reached, free program CHKNAM, and return                      930630
0043.00 V001  * 'L' in ##LAST to indicate to the calling programs that end                   930630
0044.00 V001  * of file has been reached.                                                    930630
0045.00 V001  *                                                                              930630
```

Merge operation: 2nd version (continued)

```
0046.00 V001 C                     ELSE                            ..2x                      930630
0047.00 V001 C                     MOVEL'E'       ##INCL           ..2                       930630
0048.00 V001 C                     CALL 'CHKNAM'  PLCHKN           ..2                       930630
0049.00 V001 C                     MOVEL*BLANKS   ##NAME           ..2                       930630
0050.00 V001 C                     MOVEL'L'       ##LAST           ..2                       930630
0051.00 V001 C                     END                             ..2e                      930630
0052.00       *                                                                              931007
0053.00       * Free lower invocations....                                                   931007
0054.00       *                                                                              931007
0055.00      C                     FREE PGMA                                                 931007
0056.00      C                     FREE PGMB                                                 931007
0057.00      C                     FREE PGMC                                                 931007
0058.00      C                     FREE PGMD                                                 931007
0059.00 V003 C                     MOVE '1'       *INLR            .1                        930728
0060.00 V001 C                     END                             .1e                       930630
```

- If you have executed merge operations for a number of source members in the current CR, you can re-execute all of them in one job, by selecting **option 47** from the status window in function WRKCROBJ:



```
      Actions      Text         Status       Exit         Help       028100 *NO
___________________________+)))))))))))))))))))),______________________________
OMS250C2 TSPLSYD         W * $DS1                * Objects     CR Lib O#00000301
                           * __ 37.Upd imp reg  *
 1=Retrieve      2=Edit    *    45.Adhoc Compar * =Display       6=Print
 7=Freeze        8=Display *    46.Adhoc Merge  * =Work with    13=Change obj
                           *    47.Exc Cmp/Mrg  *
Act_Object_(P)_Type_(P)___ *    48.Cmp/Mrg Rept *
__  __________ __________  *    49.Mrg Composit *
__  DSTCTL1C   CLP         *    65.Check CR log * - phase i
__  DST001     RPG         *    66.Submtd jobs  * sing - phase i
                           *    67.Wrk Outq   + *
                           *F1=Help  F12=Cancel *
                           .))))))))))))))))))))-
```

- Command EXCMRGRQS will be prompted, and you can select to re-execute a specific compare or merge operation, to re-execute the last operation, or to re-execute all compare and merge operations which were logged under the current CR execution request.

Glossary

1st version A source member used in a merge operation which contains the first set of changes made to the *base version*. Changes contained in this member are merged with changes made in the *2nd version* to produce a *composite* member. By default the merge program assumes that the 1st version changes supersede or override changes made by the 2nd version.

2nd version A source member used in a merge operation which contains the second set of changes made to the *base version*. Changes contained in this member are merged with changes made in the *1st version* to produce a *composite* member.

Application base A term used in *Thenon/SEE* to identify an object and it's associated source member as being part of the collection of objects and source members which form the common part of an application. Contrast *with site/group specific.*

Base version A source member used in a compare or merge operation which contains the originating source statements against which *changed version*, *1st version* and *2nd version* members are analysed.

Change Request An entity used in *Thenon/SEE* which forms the basic unit of changes which can be promoted and implemented through the change management cycle. Change requests (or CRs) contain changed and new objects of a specific application, which are implemented as a single unit.

Changed version A source member used in a compare operation which contains the one or more changes made to the *base version.*

Column delimiters The beginning and ending column positions of a source statement portion which is analysed for the purposes of compare or merge operations.

Compare operation The process of comparing two similar source members, and identifying the changes included in one member (the *changed version*) in relation to the other (the *base version*). The result of a compare is a report showing which statements are identical and which are different. Statements may differ by being deleted, newly inserted or moved from one place to another in the *changed version*.

Composite A source member which is generated by a *merge* operation, containing source statements which reflect the integration and consolidation of two separate sets of changes (*1st version* and *2nd version*) which were effected on a common *base version*.

Concurrent development The instance where two different sets of changes are effected on the same *base version* at the same time, typically by two different persons in two different work areas. Certain features within *Thenon/SEE* allow the management of such activity in the application development cycle.

CR See *Change Request.*

Execution request A named entity used in Thenon/ONE to log the details of one or more *compare* and *merge* operations. Each execution request can contain one or more *execution request sequences*.

Execution request sequence A number entity used in Thenon/ONE to identify a specific log record for a single compare or merge operation.

Merge operation The process of examining two source members containing two separate sets of changes which were effected on a common *base version*, and the base version itself, and generating a *composite* member reflecting the consolidated changes to the *base version*.

Object defaults A set of values automatically supplied or assumed by *Thenon/SEE* and Thenon/ONE in relation to a variety of operations performed for a certain object type or source type.

Thenon/SEE A suite of integrated software products collectively named Software Environment Engineering (SEE), which enable change management that automates, controls, tracks and audits all activities within the development and maintenance life cycle..

Site/group specific A term used in *Thenon/SEE* to identify an object and its associated source member as being part of the collection of objects and source members which form a variation from the common part of an application, geared for use at a specific site or site group. Contrast with *application base*.

Index