



SDLMERGE

Reference



Consolidation of Log and Trace Datasets

1. Introduction.....	2
2. Parameter F00 – Input dataset template	3
3. Parameter F01 – Target dataset name template	4
4. Parameter F02 – Allocation size for output datasets	5
5. Parameter F03 – Setup for permanent operation	6
6. Parameter F04 – Definition of run times for permanent operation	7
7. Parameter F05 – Definition of Load Library	8
8. Parameter F06 – Definition of FTP job	9
9. Parameter F07 – Merging of trace files	10
10. Parameter F08 – Keep input files	11
11. Parameter F09 – No EOF records.....	12

SDLMERGE

Reference

1. Introduction

SDLMERGE is the general consolidation tool to manage the log (and trace) datasets generated by SDLTRACE. Depending on the number of application systems using the logging facility (and the total size of the installation) there may be thousands of datasets being generated each day. It is therefore vital to have a versatile tool that is simple to use and at the same time efficient in handling all the tasks that are needed. With SSDLMERGE the user can set up a system which runs automatically on a regular basis, completely unattended for weeks or months without any operator intervention. It will combine like datasets and send the results to a central system where they may be combined again and stored in a common repository.

The operation of SSDLMERGE is controlled by definitions specified in member SSDLMPARM (or any other name specified in the EXEC statement that calls SSDLMERGE). The functions to be defined are labeled with identifiers named F00, F01, F02, etc., located in position 1 through 3 in the input stream, followed by additional information starting in position 5. An asterisk "*", a slash "/" or a blank in position 1 indicates a comment, so that parameters which are temporarily not needed may easily be "commented out" by replacing F01 with 01 for example. A sample member may look like this:

```
/*-----*/
/* Parameter file for SSDLMERGE - updated for rel. 1.9 29.02.2012 */
F00 ABC.LOG.*. /* Input DSN Hilevel qualifier */
F01 ABC.LOG.SDDLALL.* /* Target DSN template */
 01 ABC.LOG.SDDLALL.&SYS.* /* Target DSN template */
F02 0100,0100 /* Pri.CYL, Sec.CYL allocation */
F03 ABC.LOG.SDDLTRACE.DEMO.CNTL(SDDLMPGRUN) /* DSN for next Job */
F04 0060,0005 /* Run interval, delay, minutes*/
 04 0000,0000 /* Run interval, delay, minutes*/
F05 ABC.LOG.SDDLTRACE.DEMO.LOAD /* LoadLibraryDSN */
F06 ABC.LOG.SDDLTRACE.DEMO.CNTL(SDDLMPGFTP) /* DSN for FTP Job */
F07 Merge trace files too /* Trace data to be merged too */
F08 Keep input datasets /* Do not delete merged files */
F09 No EOF records /* No summary dataset info */
/*-----*/
```

The parameters are processed according to their occurrence in the input stream, and they need not be sorted in ascending order. If a certain parameter is entered more than once then the last valid entry will be taken.

The only parameters required for a simple merge operation are F00, F01 and F05. All others either have defined defaults or are special options needed only for additional functionality.

In the following chapters the parameters F00 to F09 will be explained in detail.

SDLMERGE

Reference

2. Parameter F00 – Input dataset template

With parameter F00 the user defines the high level and the application-id of the log datasets that are to be combined. A placeholder (“**”) may be used rather than a specific application name which will cause the name found at that level to be used also as level in the receiving dataset defined in parameter F01.

Examples for F00 are:

```
F00 user-id.APPL1  
F00 ABC.LOG.*  
F00 user-id.TRACE.*.XYZ
```

The asterisk “*” in ABC.LOG.* for example specifies that the name found at that particular level in the input DSN is to be carried to the level in the target DSN defined in F01 where an “**” is found.

Only log or trace datasets are being considered for merging. They are recognized by the fact that trace or log dataset names have as their last two levels the date and the time in the format “Dyymmda” and “ThhmmL”, where in case of trace datasets the last letter is either “A”, “B” or “C” instead of “L”. All other datasets with the same high level as that defined in the F00 parameter will be ignored.

Please note that by default only log datasets are processed. If trace datasets are to be included in the merge operation then parameter F07 must be specified too.

SDLMERGE

Reference

3. Parameter F01 – Target dataset name template

Parameter F01 defines the name of the dataset which is to receive the log or trace data records of the applications that generated them. The name consists of fixed and variable parts. The fixed parts are the levels defined literally as desired, the variable parts are the application name and the date of generation.

The application name is defined as placeholder (“*”) at exactly that level which is to receive the name from the relevant input datasets, defined in parameter F00. The date is taken from the log/trace datasets and will be placed at the very end of the generated dataset name.

The general rule is that all records from a particular day are written to a dataset which has as last DSN level exactly that day defined as “*Dyymmd*”. Thus for each application there will be a daily combined dataset containing all records that were generated during that day.

A special case is the level “&SYS”. When that is specified in the template it will be replaced at execution time by the name of the System (Image name) on which SDLMERGE is running.

Examples are:

```
F01 ABC.LOG.SDLALL.*           /* Target DSN template      */
F01 ABC.LOG.SDLALL.&SYS.*.XY   /* Target DSN template      */
```

All target datasets will be allocated as needed. If a dataset already exists from a previous run then the current data will be added to the end of the existing records. Should a dataset be in use by another program or not be available because the user is currently looking at its contents then the current trace or log dataset will be skipped and the next file will be processed. The input dataset that should have been processed will be kept for a later run.

If an output dataset is new it will be allocated using the parameters specified in the following parameter F02.

SDLMERGE

Reference

4. Parameter F02 – Allocation size for output datasets

For a new output dataset the parameter F02 defines the sizes of the primary and the secondary allocation in cylinders as follows:

```
F02 0100,0100
```

The first value defines the primary, the second value the secondary amount of cylinders to be allocated. All other allocation parameters are automatically supplied by the program. In addition the multivolume feature of the operating system is used with MAXVOL(15). The record size is set to 133 and the block size to zero, causing the system to take the maximum available block size possible for the device.

If Parameter F02 is not specified then the system will assume a medium size default and set F02 to:

```
F02 0030,0030
```

SDLMERGE

Reference

5. Parameter F03 – Setup for permanent operation

SDLMERGE may be set up to run permanently on a regular basis. This is analogous to a started task that never terminates except that for SDLMERGE there is no system setup necessary and the operation may be started and stopped at any time by the user. We used the fact that the operating system does not start a job out of the input queue if a job with the same name is already running.

In parameter F03 the name of the job and the dataset which contains this job are specified, for example as follows:

```
F03 user-id.SDLTRACE.DEMO.CNTL(SDLMGRUN)
```

This is the job that will be submitted at the very start of SDLMERGE if parameter F04 (described further down) is not zero. If zero is specified in F04 then the definition in parameter F03 has no effect since the program is executed just once and no follow-up job is scheduled.

The name of the job should be the same one as the one just submitted. The reason is that as soon as the current one (with the same job name) terminates, the second copy starts, again as first step submitting its successor, and so on indefinitely, or until terminated by the operator. The default is that SDLMERGE completes its current operation at midnight, so that one job is executed per day.

Please note that in order to cancel a job running SDLMERGE on a continuous basis it is necessary to cancel (or purge) the second copy first (the one still in the job input queue). Cancelling the active job first will simply produce another active one, and it will not be possible to ever stop SDLMERGE. If the job queue is held, of course, because the system is being shut down, then the active job may be cancelled and the next copy will not start to run until after system restart and release of the queues. Thus SDLMERGE will automatically start to run again even after an IPL (Initial Program Load) of the system.

SDLMERGE

Reference

6. Parameter F04 – Definition of run times for permanent operation

With parameter F04 the user specifies how often per day SDLMERGE is to run and at what times the individual runs are to take place. Parameter F04 is specified as follows:

```
F04 0060,0005
```

The first value defines the interval in minutes between individual runs of SDLMERGE. A time between 1 and 1440 minutes can be specified with the restriction that the value must be divisible into 1440 with a remainder of zero, so that there is a whole number of periods during one day (1440 minutes). A value of 1 means that the program should run every minute, a value of 1440 means that the program should run once per day and a value of 60 means every hour.

The second value specifies the delay in minutes that the program should wait before it actually starts with the current run after the interval specified in the first value has expired. Therefore the second value must necessarily be less than the first one. The values F04 0060,0005 mean that the program should run 5 minutes past the hour, F04 0060,0055 mean 5 minutes before the hour, F04 0720,0360 mean at 6am and at 6pm, F04 1440,0000 mean once at midnight, F04 1440,1439 mean one minute before midnight and 1440,420 mean each morning at 7am.

The specification F04 0000,0000 is a special case and means that the program is to run once right away. This is identical to not specifying parameter F04 at all. There is no repeat operation in this case and the parameter F03 has no effect.

SDLMERGE Reference

7. Parameter F05 – Definition of Load Library

SDLMERGE is a REXX program that calls the Assembler routine SDLMERGA to actually perform the merging operation. The reason for this is the increase in performance which the Assembler provides; REXX alone is simply too slow if thousands of datasets are to be merged.

Parameter F05 is used to supply the name of the load library where SDLMERGA is located:

```
F05 user-id.SDLTRACE.DEMO.LOAD
```

In addition to SDLMERGA the library also contains the module SDLWAIT which is needed when the feature “Permanent operation” is specified. SDLWAIT is also used by certain sample log routines to simulate delays in application programs.

SDLMERGE Reference

8. Parameter F06 – Definition of FTP job

The merged datasets may be sent automatically to another system for further consolidation. For this purpose there is an FTP job in the distribution library which is the sample to be used for this transfer.

The SDLMERGE program changes this sample to insert the necessary dataset names and then submit the job for execution. Thus the user should first adapt the sample to his or her requirements, for example to set the name of the receiving system and possibly to adapt other parameters needed by FTP in the users application environment.

The FTP job to be used by SDLMERGE is defined as follows:

```
F06 user-id.SDLTRACE.DEMO.CNTL(SDLMGFTP)          /* DSN for FTP Job      */
```

In this case the member name of the job is SDLMGFTP. It is stored together with all the other programs in the distribution library. The job cannot be submitted as is because the name of the dataset to send and the corresponding receiving dataset is inserted dynamically by SDLMERGE before the job is submitted.

The name of the receiving dataset is always identical to the name of the merged dataset except that it will be extended with an additional level denoting the current time in the format "ThhmmL". This will allow another merging operation on the target system to combine all data received from remote systems. (It is assumed that the received datasets are again combined with datasets from the same or other systems in another run of SDLMERGE on the receiveing system).

After a successful transfer the datasets sent will be deleted (specified in the FTP parameters).

If parameter F06 is not defined then no file transfer jobs will be generated.

9. Parameter F07 – Merging of trace files

Usually only log files are consolidated into combined files. With parameter F07 it is possible to include trace files also in a merging operation. The specification is as follows:

F07 Merge trace files too

Trace files are identified by having the letters “A”, “B”, or “C” as last characters of the last DSN level that denotes the time of creation in the form “*ThhmmA*” (or “*ThhmmB*” or “*ThhmmC*”). Specification of F07 will select all trace datasets for a merge operation if they match the selection criteria defined in parameter F00.

Since the receiving target datasets do not contain a time level as last qualifier it is not possible to easily tell (from looking at the dataset name) if the records contained in the dataset are of type trace or log. That is why each record contains the identifier “L” or “T” in position 106 of the log/trace record.

If parameter F07 is not specified then all trace files are skipped.

SDLMERGE Reference

10. Parameter F08 – Keep input files

Once an input dataset has been successfully processed and its contents written to the target file the input dataset is automatically deleted. This is the default operation. For testing purposes there is an additional parameter to specify that the input datasets are to be kept after processing.

The definition is as follows:

```
F08 Keep input datasets
```

SDLMERGE can thus be executed several times with different parameters using identical input. This is useful when testing several options to compare results.

SDLMERGE

Reference

11. Parameter F09 – No EOF records

At the end of each input dataset a line with summary information will be added to the output dataset. This separate record, which is not an actual log record, may be suppressed by specifying parameter F09.

The definition is as follows:

F09 No EOF records

When this parameter is specified SDLMERGE will not write summary information to the output file. Only log (or trace) records will be contained in the output dataset.

The EOF record is a line stating the number of records in the input dataset, its name and the disposition (deleted or kept).

The exact format of the EOF record is as follows:

99999 x nnnnnnnnn -----dataset-name----- disposition

where:

99999 denotes the EOF record.

x may be any of the letters A, B, C or D. Their meaning is:

A the input dataset is kept; the output dataset is to be sent to another system. (see Parameter F06).

B the input dataset is deleted; the output dataset is to be sent to another system.

C the input dataset is kept; the output dataset is not to be sent to another system.

D the input dataset is deleted; the output dataset is not to be sent to another system.

nnnnnnnnn is the number of records in the input dataset just processed.

-----dataset-name----- is the name of the current input dataset.

disposition is either -kept- or -deleted-.