

SMFUTIL V8

Extended Support for LOGR SMF Data Structures

This file contains instructions for using the new DDALOGR and READLOGR keywords (implemented in SMFUTIL V8, ZL0090 and modified and enhanced at SU80126) to copy data from SMF LOGR LOGSTREAM STRUCTURES.

ORIGINAL SUPPORT ADDED BY SU80090:

DDALOGR is a DDA block keyword that defines the associated DDA block as an input dataset that is an SMF LOGR LOGSTREAM structure. The associated DDASTART block DSN keyword must define the name of the LOGSTREAM structure to be used as input.

RESTRICTIONS:

1. DDALOGR should only be used for input DDA blocks. The DD name must begin with SYSUT1.
2. DDALOGR should be the last keyword in the DDA block, just prior to the DDAEND keyword that terminates the block.
3. Only 1 occurrence of the DDALOGR keyword is allowed thus only one SMF LOGR LOGSTREAM structure can be input in a single execution. This is an IBM restriction on the IFASEXIT.
4. If DDALOGR is specified on a system that does not support the LOGR SMF LOGSTREAM structures and enhanced IFASEXIT or if SMF is not recording to the specified LOGSTREAM structure the allocation will fail.

When DDALOGR is specified to read a LOGSTREAM structure and no DATE TIME range has been selected prior to the DDASTART (see enhancements described below), the entire structure will be input from OLDEST to YOUNGEST. This means some other method must be used to prevent duplicated data from being copied to the output dataset archives. SMFUTIL supports two principle methods to allow this.

1. Specify an SMFUTIL checkpoint dataset in the execution and use the SKIPOLD keyword in the execution to bypass data previously copied to the outputs. You must ensure that this "DUMP" job is run often enough to ensure that SMF data does not "age out" of the SMF LOGR LOGSTREAM structure to be read. Note this job can be run as often as required to keep the SMF archives as current as needed. See the sample job LSDUMP1 in the INSTLIB dataset.
2. Specify an SMFUTIL implicit date range selection keyword such as YESTERDAY, LASTWEEK or LASTMONTH then schedule the "DUMP" job to be run on the required day(s).

EXAMPLE 1: For YESTERDAY you would run the job each day to collect the previous day's data.

EXAMPLE 2: For LASTWEEK you would run the job each week on the day defined as your week start. For WKSTART(MON) run the job on Monday to dump the previous Monday thru Sunday. See the sample job LSDUMP2 in the INSTLIB dataset.

EXAMPLE 3: For LASTMONTH you would run the job each month on the first day of the month to collect the previous months data.

IMPORTANT NOTES:

It is critical that the SMF LOGR LOGSTREAM structure be sized appropriately to ensure that sufficient space is available to retain the required amount of SMF data for the chosen dumping frequency but not so large as to cause excess overhead during dumping resulting from reading a large amount of data to dump a small amount of data. For example, if you will be dumping daily (using YESTERDAY) you might

want to size the structure to retain three full days of data at a minimum. If you will be dumping weekly (using LASTWEEK) you might want to size the structure to retain ten full days of data at a minimum. If you will be dumping monthly (using LASTMONTH) you might want to size the structure to retain ten weeks of data at a minimum.

Appropriate precautions must be taken to ensure that an ill-behaved SMF writing application does not flood the output SMF LOGR LOGSTREAM structure causing data to be "aged" off prematurely before the "DUMP" process can collect it to the archives.

When no date time range is specified, the SMFUTIL CKPT checkpoint structure cannot normally be used to prevent running the same job more than once because the beginning input data presented by the LOGSTREAM structure is constantly changing as data is "aged" off the beginning (oldest) part of the structure's active data. As an alternative the SKIPOLD keyword may be specified to prevent data from being copied twice. SKIPOLD can be added to the methods utilizing YESTERDAY, LASTWEEK or LASTMONTH as a precaution in case the job gets run twice accidentally. This will result in a return code greater than zero as no data will be found to be copied but the outputs will not be contaminated with duplicated data.

NEW EXTENDED SUPORT ADDED VIA SU80124:

The information that follows supplements and extends the capabilities described above.

Fix level SU80090 introduced support for the processing of SMF LOGR datasets as input via the DDALOG keyword in the allocation DDA block.

The IBM supplied LOGR SMF exit IFASEXIT is used to facilitate this capability. It acts as an input exit between SMFUTIL and the SMF LOGR read process.

Fix level SU80124 significantly expands this capability by introducing support for certain key options, the IFASEXIT exit offers, to significantly reduce the amount of data processed.

These options fall into two categories, AUTOMATIC and REQUESTED.

AUTOMATIC:

AUTOMATIC consists of date/time requests and a single SYSID request.

If a specific starting and ending date time range is requested in the SYSIN input control stream **PRIOR** to the DDA block containing the DDALOG keyword, an IFASEXIT sub-option of "FROM=datetime,TO=datetime" will be generated. Any SMFUTIL keyword that causes a specific date time range to be requested will generate this sub-option. Examples are YESTERDAY, LASTWEEK, DTRANGE, DATE, GDATE and EDATE. To restate from above, these keywords must occur in the SYSIN stream **before** the DDA block specifying the DDALOG keyword.

If a single, unmasked, SYSID filter has been specified for the job, SMFUTIL will generate an IFASEXIT sub-option of "SID(ssss)" to return records for that system only. Unmasked means the request may not contain any SMFUTIL supported masking wildcard characters such as "*" or "-". It may be 1 to 4 characters that exactly matches an SMF system ID for data contained within the LOGR structure being read.

NOTE: SID only has meaning to IFASEXIT if the LOGR structure contains more than one systems SMF data so the SMFUTIL keyword SYSID should not be specified in this case. If multiple SYSID's have been requested, no SID option will be supplied to IFASEXIT because the IBM implementation does not support more than a single SID.

REQUESTED:

The DDALOG keyword now supports a parameter list field that syntactically defines three sub-parameters as follows:

DDALOG (SMEP (hhmm) , [G/L] , [N/S])

SMEP(hhmm)- Requests an SMEP (SmartEndPoint) value be passed to IFASEXIT. This is a SmartEndPoint, as defined by IBM, for the IFASEXIT and IFASMFDL utility. In brief, the SmartEndPoint is only useful when a specific date time range has been requested and the LOGR structure being accessed contains multiple systems. The SmartEndPoint may help find data for the requested SMF ID when another systems data has been encountered with a date time value above (newer) than the ending date time value requested. Valid values for “hhmm” are as follows:

0000 – 0059

0100 – 0159

0200

NOTE: The maximum value that may be specified is two hours which is an IBM limitation.

NOTE: While it may always be specified, SMEP only has meaning to IFASEXIT if an SYSID was requested **and** the LOGR structure contains more than one systems SMF data **and** a date time range has been requested. If an SYSID ID is not requested and a date time range not specified, the SMEP sub-option to IFASEXIT will be suppressed. Since SMFUTIL has no way of knowing if a particular SMF LOGR structure has more than one System ID present, the SID sub-option will always be passed if a valid SYSID has been specified.

G/L - A single character “G” or “L” indicates the time zone of the date times requested from the IFASEXIT. These are “G” for the GMT time zone or “L” for the LOCAL time zone. If not specified, the IBM default value for this is “GMT”. This parameter only has meaning when a specific date time range has been requested.

If the data returned does not appear to be all you requested, try switching from GMT to LOCAL or vice versa.

N/S - A single character “N” or “S” indicate how the IFASEXIT will handle compressed SMF LOGR data if the ZLIB hardware decompression function is not available to the execution (ZLIB support). “N” indicates “NSI” or “No Soft Inflate”. When this option is specified, if compressed SMF LOGR data is encountered the run will be terminated. “S” (the default) indicates that “Soft Inflate” is requested. This causes any compressed data to be decompressed utilizing the ZEDC software decompression functions and will result in significantly increased CPU usage. Note that “S” is equivalent to omitting the parameter but is included for completeness. Specifying “S” in the parameter list has essentially no effect other than to serve as documentation that Soft Inflate is enabled.

NOTE: Do not specify “N” unless your system has been updated to support it. IBM APAR OA49157 and its associated PTF’s, UZ80795 and US80796, implements support for ZLIB/ZEDC compressed data for IFASEXIT.

NOTE: You can easily suppress the SYSID and date time range requests passed to the IFASEXIT by moving the requesting SMFUTIL keyword(s) to after the DDA block containing the DDALOG keyword. Take note, however, if the date time range request is moved, any SMEP specification will essentially be ignored as ALL of the available data will be returned by IFASEXIT. If the SYSID request is moved, ALL systems found in the SMF LOGR structure will be returned.

Example 1:

```
YESTERDAY
SYSID (SYSA)
DDASTART DDNAME (SYSUT1A)
    DISP (SHR)
    DSN (IFASMF.SMFLOGR)
    DDALOGR (SMEP (0200) , L , N)
DDAEND
```

The IFASMF.SMFLOGR structure is read and only data for System ID SYSA for the date time range for yesterday is returned. The time zone used is the LOCAL time zone. If the data found is compressed and hardware decompression is not available, IFASEXIT will terminate the run.

Example 2:

```
DDASTART DDNAME (SYSUT1A)
    DISP (SHR)
    DSN (IFASMF.SMFLOGR)
    DDALOGR ( , , N)
DDAEND
YESTERDAY
SYSID (SYSA)
```

The IFASMF.SMFLOGR structure is read from oldest to youngest and all data for all systems is returned. SMFUTIL will filter the data, keeping only data from yesterday for System SYSA. Again, if the data found is compressed and hardware decompression is not available, IFASEXIT will terminate the run.

NOTES:

1. Reading the data for all SMF Id's contained within a single SMF Logstream may or may not result in synchronous (in date time order) data. SMF data is presented to the LOGR recording process in a non-synchronous fashion and is "hardened" to the structure as blocks are constructed. The reason for the SmartEndPoint function is to overcome this limitation when retrieving data for a single system. Because of this functional limitation, ASPG continues to strongly recommend that SMF data being permanently archived be selected via the SYSID keyword and written to unique SMF ID specific archive datasets.
2. If at all possible, it is further recommended that the LOGR structures be unique to each SMF ID and that a single LOGR structure NOT be used to receive multiple systems SMF data.
3. When a specific date time range is specified BEFORE the DDA block via the YESTERDAY, LASTWEEK or LASTMONTH key words, the Check Point dataset can be used to prevent duplicate runs as the same first record will be returned by the IFASEXIT for the entire day, week or month. Sample job LSDUMP3.TXT is a valid example of dumping data a day at a time.
4. **It is essential to understand that if:**
 - a. **Multiple systems are being recorded to a single SMF LOGR Structure.**
 - b. **And, a single or multiple systems are retrieved in a single run intended to archive data from a specific time period (YESTERDAY for example).**

The use of the SKIPOLD keyword may cause data to be missed because of the asynchronous nature of the multiple systems in the SMF LOGR Structure. The SMEP keyword can help

mitigate this but it may or may not eliminate it depending on the value chosen for SMEP and on the specific situation. The reason for this is that when specifying, for example, YESTERDAY, the execution will select only data it sees for yesterday's date. If any of yesterday's data exists in "future" blocks not processed because SMEP was not specified or not set large enough, when the same job is run on the next day, the "missed" data will be ignored as it is now older than the selected date (day before YESTERDAY).

Careful testing in your environment is essential to ensure all required data is found. One way of testing this is to make two runs, one with the date selection verb (i.e. YESTERDAY) before the DDA block (so that the "FROM=dt/TO=dt" specification is given to IFASEXIT) and one with the date selection verb after the DDA block (to cause all LOGR data to be read). The output to SYSUT2 (or other output dataset(s)) should be identical. If not, the SMEP value must be increased until all data is recovered.

It is possible to create many scenarios for dumping data from SMF LOGR Structures. The simplest might be a single job run once a day and specifying the SYSID, YESTERDAY and DDALOG(SMEP(0200)) keywords to dump a specific system's SMF data to a daily SMF Archive tape. A subsequent job could then be used to roll this dataset to a weekly and/or monthly dataset.

READLOGR – New block level LOGR processing Capability

SU80126 and up implements a new SMFUTIL keyword, READLOGR, that requests that SMFUTIL input an SMF LOGR Data Structure using native IBM LOGSTREAM processing macros.

The READLOGR keyword supports a parameter list field that syntactically defines two required sub-parameters as follows:

READLOGR(IFASMF.smflogr, [COPY/MOVE | DUMP])

IFASMF.smflogr – The name of the SMF LOGR Data Structure to be read.

COPY/MOVE | DUMP – Indicates whether the data is to be simply read (**COPY**) or read and removed from the LOGR Data Structure at successful end of run (**MOVE** or **DUMP**). Note that **MOVE** and **DUMP** are exactly the same function and perform exactly the same. They are both included for convenience and syntactical compatibility with previous implementations and documentation.

READLOGR PROCESSING NOTES:

1. Using the native LOGSTREAM I/O macros to access the log stream structure directly is not officially supported by IBM for SMF log streams even though the macros are an officially published interface for log stream structures in general. With that in mind, ASPG is committed to continuing to support this interface if at all possible.
2. The delete function of the IBM LOGR structure macros cannot delete all data within the structure in that the last block read cannot be deleted. This is a IBM limitation with the Log Stream processing itself. To conform to this IBM LOGR standard, the final block of data read from the structure will not be processed for input data but will remain in the LOGR structure to be read and processed as input data the next time the structure is processed by the job.
3. MOVE or DUMP should only be used to copy **ALL** data from the data structure (all systems it contains). If you try to limit a DUMP run to a specific system ID or a range of date/time specifications, you will end up **losing data** as blocks that contain non-selected data will be deleted. Alternatively, size the structure such that it contains a sufficient amount of data to satisfy the request you will be making and then use one or more jobs to select data for copying to the SMF Archive as required. The data is then allowed to age off the structure as new records are added to the top. This works much the same as the examples for the DDALOGR keyword above with one exception. Since no SmartEndPoint can be specified for READLOGR, you must specify the GOTOEOF keyword to make SMFUTIL read the LOGSTREAM structure to the end of file to ensure that all required data is processed.

COMPARISON of DDALOG and READLOG Processing:

In general, DDALOG is more useful for reading small amounts of data from very large SMF LOG Data Structures. The data can be selected via a date/time limiting keyword (i.e. DTRANGE) and specific SYSID.

READLOG is useful for reading all data from any size structure. The DUMP function should only be used if all data is to be returned and processed. Input data can be limited via date/time range specifications and other limiting keywords but the COPY function only should be used.