FUJITSU Software BS2000 LMS

Version V3.5B
June 2018

Readme file

1 General

*04 This readme file describes the changes and extensions of LMS V3.5B10 with respect to the manuals:

LMS V3.4B
SDF Format
Edition June 2013
Order number: U8326-J-Z125-6-76

LMS V3.4B
Subroutine Interface
Edition June 2013
Order number: U6167-J-Z125-6-76

*04 The release level is that of: June 2018.

*04 Changes to LMS V3.5B01 are marked with *04.
*03 Changes to LMS V3.5A20 are marked with *03.
*02 Changes to LMS V3.5A11 are marked with *02.
*01 Changes to LMS V3.5A02 are marked with *01.

This and other current Readme files are shipped on the Softbooks DVD and are available online at http://manuals.ts.fujitsu.com/.

2 Extensions in LMS V3.5B (Manual "LMS V3.4B SDF Format")

2.1 Adding members to a library (page 84)

Add the following paragraphs before Notes section:

The Last Byte Pointer (LBP) can be preserved for PAM files that are added to a library as X or as C members. PAM X members always preserve the LBP whereas C members preserve the LBP only if they were added to the library using SOURCE-ATTRIBUTES=*KEEP(KEEP-TYPES=*ALL) in the //ADD-ELEMENT statement. For more details, see the following table:

	Type X, access type PAM	**Type C**
SOURCE-ATTR = *STD	LBP is stored	LBP is not stored
SOURCE-ATTR = *IGNORE	LBP is stored	LBP is not stored
SOURCE-ATTR = *KEEP (KEEP-TYPES = *STD)	LBP is stored	LBP is not stored
SOURCE-ATTR = *KEEP (KEEP-TYPES = *ALL)	LBP is stored	LBP is stored

*03 The net coded character set name (NETCCSN) can be preserved for SAM node
*03 files if they are stored in the library with file attributes using the operand SOURCE-
*03 ATTRIBUTES=*KEEP.

2.2 Outputting members to a file (page 86)

Add the following paragraph after Figure 7:

*03 If any file attributes, including Last Byte Pointer (LBP) for PAM files and net coded
*03 character set name (NETCCSN) for SAM node files, were stored when adding a member, they will be restored when outputting the member to a file.
*01 Please note that if LBP is stored when adding a member, the member can't be
*01 extracted to a tape or private disk.
*01 Please note that older versions than LMS V3.4C are not able to extract C members with stored file attributes and may create an empty PAM file and/or output message LMS0053. More details can be seen in the next table:

	Type X	**Type C**
*01 LMS V3.4C and up *01	Restored with attributes and LBP	Restored with attributes and LBP
LMS V3.4B and V3.4B10	Restored with attributes without LBP	Restored with attributes without LBP
LMS V3.4A82 and earlier	Restored with attributes without LBP	LMS0053 or empty PAM file

2.3 Statement ADD-ELEMENT

2.3.1 Operand SOURCE-ATTRIBUTES

The value *KEEP of the operand SOURCE-ATTRIBUTES in the operand ELEMENT-ATTRIBUTES has been extended by the substructure KEEP-TYPES.

Format:

ADD-ELEMENT

…

ELEMENT**-ATTR**IBUTES = ***LMS-DEF**AULT / ***PAR**AMETERS(...)

***PAR**AMETERS(...)

…

,**SOUR**CE**-ATTR**IBUTES = ***LMS-DEF**AULT / ***STD** / ***IGNORE** / ***KEEP(…)**

***KEEP**(...)

KEEP-TYPES =***LMS-DEF**AULT/ ***STD**/ ***ALL**

Description:

ELEMENT-ATTRIBUTES = *LMS-DEFAULT / *PARAMETERS(...)

…

SOURCE-ATTRIBUTES = *KEEP(…)

…

KEEP-TYPES = ***LMS-DEF**AULT / ***STD** / ***ALL**
Specifies types of members for which file attributes are to be stored.

KEEP-TYPES = ***STD**
Attributes are stored for members of base types S,M,P,D,J,X.

KEEP-TYPES = ***ALL**
Attributes are stored for members of all base types that are permissible for ADD-ELEMENT.

LMS stores the Last Byte Pointer (LBP) in addition to the previously stored file attributes

- for PAM members of type X
- for members of type C if KEEP-TYPES = *ALL has been specified

Notes:

Type C is added to the list of permissible member types in FROM-FILE.

*03 2.3.2 Operand USER-DATE

*03 The operand USER-DATE in the operand TO-ELEMENT has a new value *BY-
*03 SOURCE.

```
*03      Format:
*03      ADD-ELEMENT
*03      …
*03      TO-ELEMENT = *LIBRARY-ELEMENT(...)
*03            *LIBRARY-ELEMENT(...)
*03            …
*03                 ,USER-DATE = *TODAY / *BY-SOURCE /
*03                              <date 8..10 with-compl>
```

*03 Description:
*03 **TO-ELEMENT = *LIBRARY-ELEMENT(...)**
*03 …
*03 **USER-DATE = *TODAY / *BY-SOURCE / <date 8..10 with-compl>**
*03 Date given by the user.
*03 …
*03 **USER-DATE = *BY-SOURCE**
*03 The CHANGE-DATE of a file is given.

2.4 Statement EDIT-ELEMENT

The statement has now two aliases: EDE and EDIT.

2.5 Statement EXTRACT-ELEMENT

*02 If a text member is extracted to an existing PAM file, the (possibly implicit) setting
*02 of the operand ACCESS-METHOD determines the new access method (SAM or
*02 ISAM) of the target file.

2.6 Statement FIND-ELEMENT

If a search string with an incorrect regular expression (e.g. containing multiple adjacent duplication symbols) is specified, message LMS0298 is issued.

2.7 Statement MODIFY-LMS-DEFAULTS

2.7.1 Operand SOURCE-ATTRIBUTES

The value *KEEP of the operand SOURCE-ATTRIBUTES in the operand ELEMENT-ATTRIBUTES has been extended by the substructure KEEP-TYPES.

Format:
MODIFY**-LMS-DEF**AULTS
…
ELEMENT**-ATTR**IBUTES = ***UNCHA**NGED / ***PAR**AMETERS(...)
***PAR**AMETERS(...)
…
,**SOUR**CE**-ATTR**IBUTES = ***UNCHA**NGED / ***STD** / ***IGNORE** / ***KEEP(…)**
***KEEP**(...)
KEEP-TYPES = ***UNCHA**NGED / ***STD** / ***ALL**

Description:
ELEMENT-ATTRIBUTES = *LMS-DEFAULT / *PARAMETERS(...)
…
SOURCE-ATTRIBUTES = *KEEP(…)
…
KEEP-TYPES = ***UNCHA**NGED / ***STD** / ***ALL**
Specifies types of members for which file attributes are to be stored.
KEEP-TYPES = ***STD**
Attributes are stored for members of base types S,M,P,D,J,X.
KEEP-TYPES = ***ALL**
Attributes are stored for members of all base types that are permissible for ADD-ELEMENT.

LMS stores the Last Byte Pointer (LBP) in addition to the previously stored file attributes

- for PAM members of type X
- for members of type C if KEEP-TYPES = *ALL has been specified

*03 2.7.2 Operand RUN-MODE

*03 A new LMS default RUN-MODE is added between EDT-MODE and NEXT-
*03 ATTEMPT.

*03 Format:
*03 **MOD**IFY**-LMS-DEF**AULTS
*03 …
*03 **RUN-MODE** = ***UNCH**ANGED / ***STD** / ***BATCH**
*03 Determines whether LMS is running in interactive or batch mode.

*03 Description:
*03 **RUN-MODE = *UNCHANGED / *STD / *BATCH**
*03 When LMS is running in interactive mode, determines whether LMS should run
*03 normally or behave as if it is running in batch mode. (This operand has no effect in
*03 procedures or in batch mode.)

*03 **RUN-MODE = *STD**
*03 LMS is running normally.

*03 **RUN-MODE = *BATCH**
*03 LMS is running in batch mode.

2.8 Statement SHOW-ELEMENT

*02 For the operand TEXT-INFORMATION with value *FILE-ATTRIBUTES, the output
*02 was extended by LBP information (page 398).

*02 Format:
*02 **SHOW-ELEM**ENT
*02 …
*02 **ELEM**ENT**=*LIB**RARY**-ELEM**ENT(…)
*02 **,TEXT-INF**ORMATION = ***LMS-DEF**AULT / ***ALL** / ***STATIS**TICS /
*02 ***F**ILE-**ATTRI**BUTES / ***PAR**AMETERS(...)

*02 Defines the information scope for all members except member types R, C and L.

*02 Description:
*02 **TEXT-INF**ORMATION = ***F**ILE-**ATTRI**BUTES

*02 Only the stored file attributes are output. For all PAM members except those of
*02 member type L, the output includes a Last Byte Pointer (LBP) if it has been stored
*03 as a file attribute. For SAM members, the output includes the net coded character
*03 set name (NETCCSN) if it has been stored as a file attribute.

*02 Add the following example (page 405):

*02 The stored file attributes for the PAM member NET.LBP.2DD with LBP are to be
*02 output.

```
*02   //show-element (element=net.lbp.2dd,type=x),
*02          text-information=file-attributes
*02   INPUT  LIBRARY= :1OSQ:$USER.LIB.SHOW
*02   INPUT  ELEMENT= (X)NET.LBP.2DD/@(0001)/2014-12-17
*02   ORIGINAL FILE ATTRIBUTES :
*02   FILENAME= :1OSQ:$USER.LBP1.100
*02   FILSIZE = 0000016  FCBTYPE = PAM    2ND ALLO = 00016   LBP = 00733
*02   SHARE = NO         ACCESS = WRITE
*02   BLKCTRL = NO       BLKSIZE = 002048 RECFORM = U        RECSIZE = 00000
*02   PERFORM = STANDARD   USAGE = NOT-SPEC
```

*04 Add before table (page 392):

*04 Now LMS is able to display elements with non-default CCSN and to display the text
*04 with special characters. For it the session parameters should be setting to
*04 UNICODE:
*04 On the “Configuration” menu, select “**Terminal Settings (DSS)...**”**;** in the terminal
*04 mode (“DSS Mode”) select the option “**Unicode**”; in the terminal type (“DSS Type”)
*04 select the entry “**DSS9763**” (see UNICODE in BS2000/OSD, Introduction for details).
*04 And with MODIFY-TERMINAL-OPTIONS command is set UNICODE parameter :
*04 **MODIFY-TERMINAL-OPTIONS CCS = UTFE.**

2.9 Statement SHOW-ELEMENT-ATTRIBUTES

*04 2.9.1 Parameter dependencies

*04 If a member has the attribute DESTROY = *YES, this attribute is shown in the
*04 SECURITY block.

*04 If INFORMATION = *DELTA-STRUCTURE, LMS now outputs information in the
*04 S variable like for INFORMATION = *MEDIUM (s. section 8.3 of the LMS SDF
*04 format manual). In addition, the internal delta number and the corresponding
*04 number of the base are provided in the variables varname#.DLT and
*04 varname#.BASE, respectively.

2.10 Statement SHOW-LMS-DEFAULTS

*03 2.10.1 Operand RUN-MODE

*03 A new LMS default RUN-MODE is added between EDT-MODE and NEXT-
*03 ATTEMPT.

*03 Format:
*03 **SHOW-LMS-DEF**AULTS
*03 …
*03 **DEF**AULTS = … / ***RUN-MODE** / …

*03 Description:
*03 **DEFAULTS = *RUN-MODE**
*03 Outputs whether, in interactive mode, LMS should run normally or behave as if it
*03 were running in batch mode.

2.10.2 Output format

*03 A new LMS default RUN-MODE is added between EDT-MODE and NEXT-
*03 ATTEMPT.

Values of 3rd and 4th nesting level have an indent of 10 and 15 blanks respectively. The equal sign in the output has been shifted from 24th to 29th position.

Example:

```
*03   //SHOW-LMS-DEFAULTS *ALL
*03   ELEMENT-ATTRIBUTES
*03        TYPE                   = *NONE
*03        ELEMENT-VERSION        = *ALL
*03        TO-ELEM-VERSION        = *BY-SOURCE
*03        STORAGE-FORM           = *STD
*03        SOURCE-ATTRIBUTES      = *KEEP
*03             TYPE              = *ALL
*03   FILE-ATTRIBUTES
*03        ACCESS-METHOD          = *ISAM
*03   DESTROY-DATA                = *NO
*03   WRITE-MODE                  = *CREATE
*03   DIALOG-CONTROL              = *NO
*03   INFORMATION                 = *MEDIUM
*03   LAYOUT                      = *VARIABLE
*03   SORT                        = *BY-NAME
*03   OUTPUT-FORM                 = *STD
*03   DELETE-SOURCE               = *NO
*03   PROTECTION                  = *STD
*03   MAX-ERROR-WEIGHT            = *SERIOUS
*03   EDT-MODE                    = *COMPATIBLE
*03   RUN-MODE                    = *STD
*03   NEXT-ATTEMPT                = *NO
*03   COMPARE-PARAMETERS
*03        RECORD-PART            = *ALL
*03        SPACES                 = *STD
*03        INFORMATION            = *MEDIUM
*03        LAYOUT                 = *COMPATIBLE
*03        JOIN-ELEMENT-SETS      = *NO
*03   TEXT-INFORMATION            = *ALL
*03   MODULE-INFORMATION
*03        INFORMATION            = *TXT
*03             CSECT-NAME        = *ALL
*03             ADDRESS           = X'00000000'
*03                  BASE-ADDRESS = X'00000001'
*03             LENGTH            = 2147483647
*03   PHASE-INFORMATION           = *ALL
      LLM-INFORMATION             = *ALL
```

*02 *02 *02

3 Changes to the Manual “LMS V3.4B SDF Format”

*02 *02 *02

3.1 Update on the Inquiry Mechanism in Interactive Mode (page 24)

*02 The inquiry mechanism is activated by means of the operand DIALOG-CONTROL
*02 = *YES. The inquiry mechanism is possible in the ADD-, COPY-, DELETE-, EDIT-,
*02 EXTRACT-, MODIFY-ELEMENT, MODIFY-ELEMENT-ATTRIBUTES and
*02 MODIFY-ELEMENT-PROTECTION statements. The default value can be
*02 modified by the MODIFY-LMS-DEFAULTS statement.

*02 The inquiry mechanism can be restricted to error cases by DIALOG-CONTROL =
*02 *ERROR. In the following non-recoverable error cases, the inquiry mechanism is
*02 activated in interactive mode even if DIALOG-CONTROL = *NO:
*02 – A member cannot be accessed, e.g. because it is locked by another user
*02 (temporary exclusive use of a member, e.g. due to modification).
*02 – A library cannot be accessed, e.g. because the current access rights do
*02 not permit access (temporary exclusive use of a library).

*03 In interactive mode, it is possible to make LMS behave as if it were running in
*03 batch mode by setting RUN-MODE to *BATCH in the MODIFY-LMS-DEFAULTS
*03 statement. LMS does not activate the inquiry mechanism then.

*02 *02

3.2 Update of the DIALOG-CONTROL Operand Description in Various Statements

*02 The update refers to the following statements:
*02 ADD-ELEMENT
*02 COPY-ELEMENT
*02 DELETE-ELEMENT
*02 EDIT-ELEMENT
*02 EXTRACT-ELEMENT
*02 MODIFY-ELEMENT
*02 MODIFY-ELEMENT-ATTRIBUTES
*02 MODIFY-ELEMENT-PROTECTION
*02 PROVIDE-ELEMENT
*02 RETURN-ELEMENT

*02 **DIALOG-CONTROL = *NO**
*02 All members are processed without dialog queries, i.e. without the user being able
*02 to take control.
*02 Exception: If a member or a library is locked, LMS inquires whether the attempt to
*02 access it shall be repeated.

*03
3.3 Update on Setting CCSNs explicitly (page 118)

*03 Note:

*03 For SAM node files, the net coded character set name (NETCCSN) may be stored
*03 as a file attribute with the element. It is possible to change CCSN using the
*03 statement MODIFY-ELEMENT-ATTRIBUTES, but it is not possible to modify
*03 NETCCSN in a member. That means that after changing CCSN and extracting a
*03 member the processing of the resulting SAM node file might be rejected if
*03 ONETSTOR is not able to perform the desired code conversion. In this case,
*03 suitable character sets should be set using the command MODIFY-FILE-
*03 ATTRIBUTES.

*02
*02
3.4 Example on the Effect of the DIALOG-CONTROL Operand (page 350)

*02 The member ELEM1 of the library LIB1 is to be edited using the name ELEM2.
*02 The member ELEM2 already exists. Since EDIT-ELEMENT has WRITE-
*02 MODE=*CREATE by default, LMS rejects editing the member. After entering
*02 MODIFY-LMS-DEFAULTS DIALOG-CONTROL=*YES, the inquiry mechanism is
*02 activated so the user can control further LMS processing by his answer to the
*02 LMS0404 message.

```
*02   /START-LMS
*02   //OPEN-LIBRARY LIB1,*UPDATE
*02   //EDIT-ELEM (,ELEM1,S),(ELEM=ELEM2,TYP=S)
*02   INPUT  LIBRARY= :1OSQ:$USER.LIB1
*02   OUTPUT LIBRARY= :1OSQ:$USER.LIB1
*02   INPUT  ELEMENT= (S)ELEM1/@(0001)/2015-02-06
*02   OUTPUT ELEMENT= (S)ELEM2/@/2015-02-06
*02         NO CORRECT (S)ELEM1/@(0001)/2015-02-06 AS
*02                                           (S)ELEM2/@/2015-02-06
*02                    , OUTPUT EXISTING
*02   //MOD-LMS-DEF DIALOG-CONTROL=*YES
*02   //EDIT-ELEM (,ELEM1,S),(ELEM=ELEM2,TYP=S)
*02   %  LMS0522 EDIT ELEMENT (S)ELEM1/@(0001)/2015-02-06 IN
*02   LIBRARY :1OSQ:$USER.LIB1 AS ELEMENT (S)ELEM2/@/2015-02-06 IN
*02   LIBRARY :1OSQ:$USER.LIB1
*02   %  LMS0404 ELEMENT ALREADY EXISTS. OVERWRITE ? REPLY (Y=YES;
*02   N=NO; A=NO AND SWITCH OFF DIALOG CONTROL; T=ABORT)?
```

4 Changes to the Manual “LMS V3.4B Subroutine Interface”

*04 4.1 Function and format of the parameter structures, EI (Element *04 Information) (page 28)

	ACCESS-TIME	Time of last access to member	8	8 x X'40'
		reserved	1	1 x X'40'
	ELEMENT-SIZE	Member size indicator	4	4 x X’00’
*04	DESTROY-DATA	Destroy indicator	1	1 x X’40’
*04		reserved	39	39 x X'40'

4.2 Format of the attribute record (record type 164) (page 142)

	PMFSIZ	FILE SIZE >= 32 GB PMSIZE must be X'FFFFFF'	4	4 x X'00'
	PMLBP	LAST BYTE POINTER	2	2 x X'00'
	PMLBPV	LAST BYTE POINTER VALID	1	1 x X’00’
*03	PMNCCS	NET-CODED-CHAR-SET	8	8 x X’40’
*03		reserved (must be 0)		75 x X'00'

Fields PMLBP with length of 2 bytes and PMLBPV with length of 1 byte are added
to the attribute record (record type 164) in the beginning of reserved field.
*03 Field PMNCCS with length of 8 bytes is added to the attribute record (record type
*03 164) after fields PMLBP and PMLBPV in the beginning of reserved field.
*03 The reserved field was previously 86 bytes long and is now only 75 bytes long.

4.3 COBOL interface

4.3.1 LMSCOBEI (page 148)

```
            02  ACCESS-TIME        PIC X(8)            VALUE SPACES.
            02  FILLER             PIC X(1)            VALUE SPACES.
            02  ELEMENT-SIZE       PIC  S9(9) COMP     VALUE 0.
*04         02  DESTROY-DATA       PIC X(1)            VALUE SPACES.
*04         02  FILLER             PIC X(39)           VALUE SPACES.
```

4.3.2 Format of a record of type 164 (page 165)

```
            02  FILESIZE-32        PIC X(4)        VALUE LOW-VALUE.
            *                                 >= 32GB, FILESIZE MUST BE X'FFFFFF'
            02  LBP                PIC X(2)        VALUE 0.
            02  LBPV               PIC X           VALUE 0.
            *
*03         02  NCCS               PIC X(8)        NET-CODED-CHAR-SET
*03         *
*03         02  FILLER             PIC X(75)       VALUE LOW-VALUE.
```

4.4 C interface

4.4.1 Include element LMS.H (page 180)

```
            unsigned int element_size;/*element size                          out */
*04         char  destroy_data;       /*   destroy data                       out */
*04         char  filler9[39];        /*   reserved                               */

            ⋮

            0,                                   /* element size                 *\
*04         ' ',                                 /* destroy data                 *\
*04         {BLANK18,BLANK18,' ','  ','  '}      /* reserved                     *\
```

4.4.2 Include element LMSREC.H (pages 195 and 196)

```
            char  filesize_32[4];    /*   file size >= 32GB, filesize must be X'FFFFFF'    */
            char  lbp[2];            /*   last byte pointer                                */
            char  lbpv;              /*   last byte pointer valid                          */
*03         char  nccs[8];           /*   net-coded-char-set                               */
*03         char  filler3[75];       /*   reserved, must be 0                              */

            ⋮

            {'\0','\0','\0','\0'},   /*  file size >= 32GB                                *\
            {'\0','\0','\0'},        /*   lbp = 0 and lbpv not set                         *\
*03         {BLANK8},                /*  netccsn                                          *\
*03         {X0_32,X0_32,X0_8,   /*   reserved, must be 0                                 *\
*03         '\0','\0','\0'},\
```

4.5 Assembler interface

4.5.1 LMSASSEI (page 215)

```
           &P.ESIZE     DS   F           ELEMENT-SIZE    OUT    FL4'0'
*04        &P.DESTR     DS   CL1         DESTROY-DATA    OUT    CL1' '
           *
*04        &P.RES7      DS   CL39        FREE                   CL39' '
           &P.PLNG      EQU  *-&P.ELEM   EI LENGTH
```

4.5.2 Format of a record of type 164 (page 239)

```
           &P.FSIZ      DS   XL4         FILE SIZE >= 32GB,
           *                                 &P.SIZE MUST BE X'FFFFFF'
           &P.LBP       DS   XL2         LAST BYTE POINTER
           &P.LBPV      DS   X           LAST BYTE POINTER VALID
           *
*03        &P.NCCS      DS   CL8         NET-CODED-CHAR-SET
*03        *
*03                     DS   CL75        RESERVED, MUST BE 0      XL75'00’
```

5 Messages

The new message LMS0298 is output by statement //FIND-ELEMENT if the specified regular expression is incorrect.

% LMS0298 INCORRECT REGULAR EXPRESSION

During start of LMS the new message LMS0299 can be output if the LMS subsystem is created and the version of subsystem mismatches the version of the LMS starter phase or SYSLNK library:

*03 % LMS0299 LMS VERSIONS (&00) AND (&01) MISMATCH

*03 Meaning:
*03 Possible reasons:
*03 - Subsystem and library versions mismatch.
*03 - Launcher phase and library versions mismatch.
*03 - Launcher phase and subsystem versions mismatch.