

Line Text Access Method (LTAM)

Introduction

The Line Text Access Method (LTAM) provides line-orientated access to Windows or Unix text file. When reading from a text-file LTAM returns the body of a text-line, excluding any CR or LF characters. When writing to a text file LTAM adds the appropriate <LF> or <CR><LF> End-Of-Line (EOL) characters to the end of the text-line. The actual EOL character(s) are normally dependent on the host operating-system but can be overridden when opening the Windows (or Unix) file.

The ORGANISATION Statement

If a program uses LTAM the statement:

```
ORGANISATION OR$83L TYPE 3 EXTENSION 320
```

must be coded in the data division before the first FD or data declaration.

File Processing Statements

The following procedure division statements are provided for LTAM:

OPEN NEW	which must be executed prior to any other statement affecting the file;
OPEN OLD	which must be executed prior to any other statement affecting the file;
READ	to read a record at random;
READ FIRST	to read the very first record in the file;
READ NEXT	to read the next record during sequential processing;
WRITE NEXT	to write a record during sequential processing;
CLOSE	to terminate processing of a file.
CLOSE TRUNCATE	to terminate processing of a file.
CLOSE DELETE	to terminate processing of a file.

The File Definition

The file definition for a basic direct file is coded in either working storage or the linkage section as follows:

```
FD filename ORGANISATION OR$83L
[ASSIGN TO UNIT "?" FILE "?"]
RECORD LENGTH IS length
[KEY IS keyname]
[SIZE IS size]
01 FILLER REDEFINES filename
02 FILLER PIC X(80)
02 LPTR PIC PTR * Access Method Pointer for OPEN operation
02 FILLER PIC X(24)
02 LTNAME PIC X(256) * File name with terminating #00
02 FILLER PIC X(9)
```

LTAM – Line Text Access Method

```
02  LTPERM      PIC 9(4) COMP      * Unix permissions
02  FILLER      PIC X(4)
02  LTLENG      PIC 9(4) COMP      * Record length
02  LTOVER      PIC 9 COMP          * Override flag
                                   * 0 No override
                                   * 1 terminate lines with <LF>
                                   * (Unix file convention)
                                   * 2 terminate lines with <CR><LF>
                                   * (Windows file convention)
*
*   For GSM SP-35, and later.
*
02  LTTERM      PIC 9 COMP          * 0 = Treat #0A as EOL character
                                   * 1 = Treat #0D as EOL character
*
*   For GSM SP-36, and later.
*
02  LTNOLT      PIC 9 COMP          * 0 = Add line terminator as per LTOVER
                                   * 1 = Suppress line terminator char(s)
```

OPEN Operations

The path and file name of the file to be opened together with any terminating #00 must be moved into LTNAME before the open operation.

The record length in the FD must be set up on opening the file. This record length indicates the maximum length of line that can be read into the calling program.

For OPEN OLD, the file-size is returned.

The open operations cannot be called directly. They must be coded in the following way:

```
CALL LTPTR USING fd opcode buff
```

where *opcode* is a PIC 9(4) COMP field which is set to:

```
1 for OPEN NEW;
2 for OPEN OLD
```

and *buff* is the read buffer area for use by the access method as follows:

```
01  BF
02  BFLENG      PIC 9(4) COMP      * Buffer length
02  BFAREA      PIC X(nnnn)       * Buffer area
```

The BFLENG field which indicates the size of BFAREA must be at least twice the maximum record length supplied in the record length field.

The buffer block will be used by the access method and must not be overwritten whilst the FD remains open. If more than a single FD is opened at any time then each FD will require its own buffer block.

WRITE Operations

When writing a line to the text file you must either set the length of the line to be written in LTLENG, or set LTLENG to -1 to indicate that the line is terminated by #00.

If LTOVER = 0 the access method will terminate the line when writing with a line terminator appropriate to the host operating system.

If LTOVER = 1 it will always terminate the line with #0A.

If LTOVER = 2 it will always terminate the line with #0D0A.

For GSM SP-36, and later, it is possible to suppress the automatic line terminator character(s) by setting LTNOLT to 1. Each WRITE operation will reset LTNOLT to 0 so that subsequent lines are written normally (i.e. with the appropriate line-terminator character(s)).

READ Operations

The line read will always be terminated with #00. Your text-line buffer should therefore be one byte longer to accommodate this. The length of line returned excluding the terminating #00 will be returned in LTLENG.

For GSM SP-35, and later, it is possible to treat #0D (Carriage Return) as the End-of-Line character, rather than the default #0A (Line Feed). This option must be used when importing CSV files that may contain a Line-Feed character (within double-quotes) in the body of a text line. Set LTERM to 1 indicates that #0D should be considered as the EOL character.

CLOSE Operations

The close operation will simply close the file. Note that the CLOSE and CLOSE TRUNCATE operations are effectively the same.