

FUJITSU Software BS2000 EDT

Version 17.0D
June 2017

Release Notice

All rights reserved, including intellectual property rights.
Technical data subject to modifications and delivery subject to availability. Any liability that the data and illustrations are complete, actual or correct is excluded. Designations may be trademarks and/or copyrights of the respective manufacturer, the use of which by third parties for their own purposes may infringe the rights of such owner.

© 2017 Fujitsu Technology Solutions GmbH

Fujitsu and the Fujitsu logo are trademarks or registered trademarks of Fujitsu Limited in Japan and other countries. BS2000 is a trademark of Fujitsu Technology Solutions GmbH in Germany and other countries.

1	General	3
1.1	Ordering	3
1.2	Delivery	4
1.3	Documentation	5
2	Software extensions	6
2.1	Extensions/improvements for EDT V17.0A	6
2.1.1	EDT operating modes	6
2.1.2	Long records	7
2.1.3	Local character records	7
2.1.4	Consistent management of blank lines	7
2.1.5	Availability of all working files in Unicode mode	7
2.1.6	Standardizing command interfaces	8
2.1.7	Subsystems	8
2.2	Extensions/improvements for EDT V17.0B	8
2.3	Extensions/improvements for EDT V17.0C	9
2.4	Extensions/improvements for EDT V17.0D	9
2.5	Extensions/improvements for EDT V17.0D20	9
3	Technical information	10
3.1	Resource requirements	10
3.2	Software configuration	11
3.3	Product installation	11
3.3.1	General	11
3.3.2	Installing a private version	11
3.4	Product use	12
3.5	Discontinued functions (and those to be discontinued)	13
3.6	Incompatibilities	13
3.7	Restrictions	13
3.8	Procedure in the event of errors	14
4	Hardware requirements	14

1 General

EDT (EDITOR) is a program for editing files.

It can be used for editing SAM, ISAM and POSIX files as well as library elements, job variables and S variables. In Unicode mode SAM, ISAM, POSIX files and library elements can also be processed in Unicode format.

- *6 This Release Notice is a summary of the major extensions, dependencies and operating information with respect to EDT V17.0D under the BS2000 operating system.
- *7 The release level is that of: June 2017.
- *7 Changes in correction EDT V17.0D20 compared to EDT V17.0D00 are marked with *7.
- *6 Changes in correction EDT V17.0D00 compared to EDT V17.0C00 are marked with *6.
- *6 Changes in correction EDT V17.0C00 compared to EDT V17.0B10 are marked with *5.
- *5 Changes in correction EDT V17.0C00 compared to EDT V17.0B10 are marked with *5.
- *4 Changes in correction EDT V17.0B10 compared to EDT V17.0B00 are marked with *4.
- *4 Changes in correction EDT V17.0B10 compared to EDT V17.0B00 are marked with *4.
- *3 Changes in correction EDT V17.0B00 compared to EDT V17.0A10 are marked with *3.
- *3 Changes in correction EDT V17.0B00 compared to EDT V17.0A10 are marked with *3.
- *2 Changes in correction EDT V17.0A10 compared to EDT V17.0A07 are marked with *2.
- *2 Changes in correction EDT V17.0A10 compared to EDT V17.0A07 are marked with *2.
- *1 Changes in correction EDT V17.0A07 compared to EDT V17.0A00 are marked with *1.
- *1 Changes in correction EDT V17.0A07 compared to EDT V17.0A00 are marked with *1.

This and other current Release Notices are shipped on the SoftBooks DVD and are available online at <http://manuals.ts.fujitsu.com/>.

If one or more previous versions are skipped when this product version is used, the information from the Release Notices (and README files) of the previous versions must also be noted.

1.1 Ordering

- *6 EDT V17.0D can be ordered from your local distributors.
- *6 EDT V17.0D is supplied subject to a single payment or payment by installments.
- *6 EDT V17.0D is supplied subject to a single payment or payment by installments.

1.2 Delivery

- *6 The EDT V17.0D files are supplied via SOLIS.
 - *6 The following delivery groups are part of the EDT V17.0D delivery scope:
 - EDT Start phase in compatibility mode
 - *4 EDTSTART Procedure executed when starting EDT (*)
 - SINPRC.EDT.170 Procedure for installing a private version
 - SYSFGM.EDT.170.D Release Notice (German)
 - SYSFGM.EDT.170.E Release Notice (English)
 - SYSLIB.EDT.170 User macro library
 - SYSLNK.EDT.170 EDT module library
 - SYSLNK.EDT.170.INIT EDT module library for initialization
 - SYSMES.EDT.170 System message file (MSGMAKER)
 - *1 SYSPRG.EDT.170 Start phase in compatibility mode
 - SYSPRG.EDT.170.EDTU Start phase in Unicode mode
 - SYSREP.EDT.170 REP file
 - *7 SYSRME.EDT.170.D Readme file (file with German text)
 - *7 SYSRME.EDT.170.E Readme file (file with English text)
 - SYSRMS.EDT.170 Correction deposit for RMS
 - SYSSDF.EDT.170 Syntax file for SDF (START-EDT / START-EDTU)
 - SYSSII.EDT.170 Structure and installation information
 - SYSSMB.EDT.170 Symbol file for dump analysis
 - SYSSSC.EDT.170 Subsystem declarations for SSCM
 - *4 * entry in IMON SCI only
- The current file and volume characteristics are listed in the SOLIS2 delivery cover letter.

1.3 Documentation

*6 The OSD/BC documentation is available in German and English on DVD with the
*6 title BS2000 SoftBooks.

*4 The following manuals describe EDT V17.0:

EDT (BS2000/OSD) Unicode Mode V17.0A Statements
(Order number German: U41709-J-Z125-1
(Order number English: U41709-J-Z125-1-76

EDT (BS2000/OSD) Unicode Mode V17.0A Subroutine Interfaces
(Order number German: U41710-J-Z125-1
(Order number English: U41710-J-Z125-1-76

Only the Unicode mode is described in these two manuals. For compatible mode,
see the EDT V16.6B Manuals.

*6 The documentation is also available on the internet at
*6 <http://manuals.ts.fujitsu.com>. Manuals which are displayed with an order number
*6 can also be ordered in printed form.

*5 The manuals may be supplemented with README files. These contain changes
*5 and extensions to the manual of the product concerned.
*5 The file names are made up as follows:

*5 SYSRME.EDT.170.E (file with English text)

*6 The README files are available on the SoftBooks-DVD or online under
*6 <http://manuals.ts.fujitsu.com>.

2 Software extensions

- *4 Section 2.1 covers the extension of EDT V17.0A (in comparison to EDT V16.6B).
- *4 The differences between EDT V17.0A and V17.0B follow in section 2.2.
- *5 The differences between EDT V17.0B and V17.0C follow in section 2.3.
- *6 The differences between EDT V17.0C and V17.0D follow in section 2.4.
- *7 The differences between EDT V17.0D and V17.0D20 follow in section 2.5.

2.1 Extensions/improvements for EDT V17.0A

EDT V17.0A has been extended with important functions. The version supports the processing of files encoded with Unicode and the restriction of the record length in Unicode to 256 characters mode has now been withdrawn.

2.1.1 EDT operating modes

The main improvement in EDT V17.0A compared to its predecessor is the support for processing Unicode encoded files. Both of the following objectives must be equally considered:

- The user who wishes to process files encoded in Unicode must be provided with a convenient method of support. This includes the possibility of simultaneously processing various encoded files in various work EDT files and the lifting of the record length restriction (which was previously 256 characters).
- The user, who wants to be able to process his old files encoded in 7-bit or 8-bit character sets as before, should be offered compatible functions and interfaces as in EDT V16.6B. This is especially valid for handling EDT procedures and sub-program interfaces.

The extension of the permitted record length and the internal use of a Unicode representation in the work files mean that not all the interfaces can remain compatible where the user is permitted direct access to the work file records. This is valid for the L-mode subroutine interface, for the (old) @RUN-interface and for the *Locate-Mode* of the IEDTGLE interface. These interfaces can no longer be used if you want to use the new functions.

Nevertheless, programs that merely use the EDT subroutine interfaces to enable their users to edit files without having to exit the program can also work with files programmed in Unicode without any changes.

This is because EDT V17.0A can be operated in two modes:

- In *Compatibility mode*, which offers the full functionality of the EDT V16.6B, but does not permit Unicode file processing nor supports the extended record length or the local set-up of character sets.
- In *Unicode mode*, which has been extended to process Unicode files but has several incompatibilities especially on the subroutine interface.

2.1.2 Long records

In the EDT V17.0A unicode mode, the restriction of the record length to 256 characters has been lifted. When reading from or writing to a DMS-file EDT can process records with a maximum length of 32768 bytes. This restriction is predefined by the DMS and refers to the *byte quantity*. The maximum line length of library elements is limited by LMS at 32763 bytes.

When Unicode character sets are used, a character is mostly encoded with several bytes, the number of *characters* permitted per record is possibly less. Internally the EDT functions with larger buffers so that this restriction only becomes effective when the records to be written are recoded in the character record of the output file. The EDT offers a command (@CHECK, Format 2) which can be used to determine whether records must be shortened when written.

Lifting the record length restriction affects data records in the work files, character string variables and the command lengths in equal measure and is reflected in the syntax of the commands (e.g. with column commands), in the layout of the status display, in the partitioning of the monitor with @EDIT LONG and in the subroutine interface.

2.1.3 Local character records

In the EDT V17.0A Unicode mode, a different character set can be set for each work file. The following is permitted: the already supported 7-bit and 8-bit character sets and the Unicode character sets UTF8, UTF16 and UTFE, the ISO character sets supported by XHCS, and the user-defined character sets defined in XHCS.

The character sets for the individual work files are either implicitly set when reading in an appropriately coded file or explicitly via the @CODENAME command.

2.1.4 Consistent management of blank lines

The files to be processed by EDT can contain records with the length 0. POSIX files or SAM files have records with the length 0. ISAM files with standard features have records with the length 8 or 16 (with UTF16 encoded files).

2.1.5 Availability of all working files in Unicode mode

In F-mode, a change can be made to one of the work files 10 - 22 by entering the appropriate number in the command line. This was previously only possible in L-mode using the @PROC command.

2.1.6 Standardizing command interfaces

In EDT V17.0A, access can be made to files of *all* supported file types with a uniform set of commands (@OPEN, @CLOSE, @COPY, @WRITE). The commands have been extended in the respective operands. Although old commands can still be used, it is recommended to use only the new commands.

For reasons of completeness and uniformity, new operands have been introduced for the other commands. Details are in the command descriptions.

2.1.7 Subsystems

EDT V17.0A now comprises 3 subsystems which should be started as required:

EDTCON / EDT / EDTU

At first the subsystem EDTCON can be preloaded with
/START-SUBSYSTEM SUBSYSTEM-NAME=EDTCON,SYNC=*YES

Subsequently the subsystems EDTU and/or EDT can be preloaded in any order with
/START-SUBSYSTEM SUBSYSTEM-NAME=EDTU
resp.
/START-SUBSYSTEM SUBSYSTEM-NAME=EDT

Before starting an IMON installation (new/update) of EDT 17.0 all EDT subsystems must be terminated by hand (/STOP-SUB), until further notice.

Prior to an unlikely rollback to EDT 16.6B all EDT subsystems must have been terminated as well. Additionally the subsystem EDTU must be removed from the subsystem catalogue by hand (see example).

```
/START-SSCM
//START-CAT-MOD subsystem-catalogue
//REM-CAT-ENT SU-NA=EDTU(VER=17.0)
//CHECK
//SAVE
//END
/
/REM-SUB EDTU,17.0
```

2.2 Extensions/improvements for EDT V17.0B

- *4 With this version EDT does not automatically connect to the POSIX subsystem.
- *4 The connection will only be established if POSIX files have to be edited.
- *4 If only BS2000 files have to be edited no connection to POSIX will be established.

2.3 Extensions/improvements for EDT V17.0C

- *7 This EDT version includes change requests for EDTU as follows.
- *5
 - The statement @SDFTEST can also be used to test command and statement lines that contain semicolons.
 - *5
 - If a library list is defined via an S variable, the name of which begins with SYSPLAMALT, the name of the current input library is then output via the EDT0274 message once the library element has been read.
 - *5

2.4 Extensions/improvements for EDT V17.0D

- *6 This version supports write stations for use with telnet also in Unicode-mode.

2.5 Extensions/improvements for EDT V17.0D20

- *7 This EDT version includes changes as follows:
- *7
 - Line number display
 - *7 With EDTU it is possible to display line numbers in F-Mode display with all 4 digits after the decimal point (@INDEX, @PAR).
 - *7
 - Error logging for syntax check
 - *7 During execution of the statement @SDFTEST in batch the syntax error messages will be logged to SYSLST.
 - *7
 - SAM node file processing
 - *7 With EDTU it is possible to process SAM node files.

3 Technical information

3.1 Resource requirements

*6 The resource requirements of EDT V17.0D and V17.0C compared to EDT V17.0B
 *5 and EDT V17.0A have hardly changed.
 *6 EDT V17.0D in Compatible Mode requires the following memory:

*6	Minimum requirement: Class 4 memory, if EDT V17.0D is loaded as a subsystem, otherwise class 6 memory.	360 KB
----	---	--------

*6	Additional class 6 memory requirement for work levels and other purposes:	1460 KB
----	--	---------

*6 EDT V17.0D in Unicode Mode requires the following memory:

*4	Minimum requirement: Class 4 memory, if EDT V17.0D is loaded as a subsystem, otherwise class 6 memory.	1670 KB
----	---	---------

*6	Additional class 6 memory requirement for work levels and other purposes:	2700 KB
----	--	---------

General note: The memory requirement of EDT V17.0A in "Unicode Mode" is almost double compared to the EDT V16.6B memory requirements (for the same actions).

*1 POSIX resources, in case of active POSIX:

*1 Each BS2000 task loading EDT V17.0A temporarily creates a POSIX process
 *1 while running EDT.
 *1 This also applies for any program using the EDT subroutine interface.

*1 This could cause the problem that the actual values for the
 *1 POSIX control parameter NPROC, MAXUP and NOSTTY are no longer
 *1 sufficient. In case of a shortage these values can be increased with the
 *1 POSIX command "usp" without restarting POSIX.
 *1 Further information can be obtained in the user manual
 *1 "POSIX-Basics", chapter "POSIX Information file".

*4 From EDT V17.0B onwards a connection to POSIX will only be established if
 *4 POSIX files have to be to edit.

3.2 Software configuration

- *6 EDT V17.0D is released for servers of the S-Series as of operating system version BS2000/OSD-BC V6.0B and for servers of the SX-Series as of OSD/XC V2.2 and
- *6 for servers of the SQ-Series as of OSD/XC V4.1 and for servers of the SE Series
- *6 as of OSD/BC V10.0.
EDT is not dependent on the address mode.

- *1 For EDT V17.0A07 the correction release 1/2008 is required.
- *2 For EDT V17.0A10 the correction release 2/2008 is required.
- *3 For EDT V17.0B00 the correction release 2/2009 is required.
- *4 For EDT V17.0B10 the correction release 2/2010 is required.
- *5 For EDT V17.0C01 the correction release 2/2011 is required.

- *5 The EDT version V17.0C01 can only run in BS2000/OSD-BC V6.0 if at least the
- *5 versions of the subsystems of CRTE-BASYS V1.6F and SDF V4.7C are also in-
- *5 stalled on the last deliverable correction release 1/2011 of this OSD-BC version.

- *6 For EDT V17.0D00 the correction release 1/2015 is required.
- *7 For EDT V17.0D20 the correction release 1/2016 is required.

3.3 Product installation

3.3.1 General

Before installing the product, observe the information in the Release Notice, the installation information in the delivery documentation and the information in the Product Manual.

The necessary inputs and the sequence of the installation are described in the IMON documentation.

The module library SYSLNK.EDT.170 must also be known to the system under the name \$EDTLIB in order to support existing programs that call EDT as a sub-program with L-mode interface.

- *4 If EDT V16.0B remains on the system and should be used simultaneously with EDT V17.0 the correction release 2/2007 for EDT V16.6B is required.

3.3.2 Installing a private version

- *4 SINPRC.EDT.170 contains a procedure with which a private version of EDT V17.0 can be installed under any desired user ID. A private version should only be used for evaluation purposes and not for the coexistence of two EDT versions.

The following files must be available on the installation ID of the private EDT before calling this procedure:

SYSPRG.EDT.170	(starter phase compatibility mode)
SYSPRG.EDT.170.EDTU	(starter phase unicode mode)
SYSLNK.EDT.170	(EDT module library (EDTLIB))
SYSLNK.EDT.170.INIT	(Initialization library)
SYSREP.EDT.170	(Rep file)
SYSPRC.EDT.170	(Procedure for installing)

For compatibility reasons a file named EDT is used as starter phase for the Compatibility Mode (like in any public installation), if the file SYSPRG.EDT.170 does not exist in the installation path. But it is recommended to use SYS-PRG.EDT.170, as this is the only possibility to support private installations of different EDT versions on the same installation path.

The procedure can be called by the installation ID or TSOS:

```
/CALL-PROC SINPRC.EDT.170
```

The procedure parameters can be specified using prompts:

```

USERID      : Installation ID
ORIG        : Defines whether the private EDT is to be
              installed in the original files or copies
              (files are modified). The names of the copies
              must be specified with the following
              parameters:
EDT          : Name of the private starter phase C mode
EDTU        : Name of the private starter phase U mode
EDTLIB      : Name of the private module library
INILIB      : Name of the private initialization library
REPFIL      : Name of the private Rep file

```

The private EDT version can be started with the command

```
/START-PROG FROM-FILE=$<userid>.<name-private-starterphase>
```

With this procedure all relations to logical names defined in IMON will be reset for any private versions. This applies especially for a predefined assignment to SYSDAT.EDTSTART. Thus a centrally defined EDT start procedure will not be executed by any private version.

Programs which call this private EDT as a subprogram must

- either link in the connection module IEDTGLE from the created module library
- or load it dynamically using the ENTRY name IEDTGLE. The parameter 'SHARE=NO' must thereby be specified in the 'BIND' macro.
- IF EDT, EDTU and EDTCON are not loaded as subsystems, programs which load EDT from \$EDTLIB using the 'BIND' (not 'LINK') macro call, can assign a private module library with the following command:

```
/SET-FILE-LINK LINK-NAME=BLSLIBnn, FILE-NAME=<modulbib>
```

where nn=00..99

3.4 Product use

A Unicode file can only be used comfortably in dialog operation if a data terminal or terminal emulation is used that can process Unicode characters, e.g. under Windows the MT9750 emulation as of V7.0.

3.5 Discontinued functions (and those to be discontinued)

- IEDTCALL
The discontinued IEDTCALL interface for the EDT V16.5A is no longer part of EDT V17.0 (Replacement: IEDTGLE).

*4

3.6 Incompatibilities

The *compatibility mode* offers the full functionality of the EDT V16.6B including the old L-mode sub-program interface. The extended functions of the Unicode mode are not available in Compatibility Mode.

*4 The Compatibility Mode of EDT V17.0 was only extended by some necessary functions.

- In Compatibility Mode the command @CODENAME had been changed for migration purposes to support a selective change of the character set for non empty work files.
- Compatibility mode is extended by Format 5 of the @IF command, which permits the scanning of and, if necessary, reaction to the current operating mode.
- The new @MODE command is also introduced in the compatibility mode and permits a switchover to Unicode mode.

*7 EDT V17.0D20 Unicode mode behavior is incompatible to version V17.0D10 in 2 points.

*7

*7

*7

*7

*7

*7

*7

- For @STATUS=PAR(procnr) resp. @STATUS=PAR(*) the output line showing the option selected for the line number display is extended by 1 character.
- If an invalid value is specified for @PAR INDEX=, the message EDT3002 is issued instead of message EDT3071.

3.7 Restrictions

*7 There are no restrictions for EDT V17.0D.

3.8 Procedure in the event of errors

- *4 If an error occurs with V17.0, the following error documentation will be required for diagnostic purposes:
 - Runtime log
 - Procedures which caused errors, in file form
 - Used input data in file form
 - User dump
 - The file SYSREP.EDT.170

4 Hardware requirements

- *4 EDT V17.0 is executable on all business servers and terminals supported by BS2000. Only on 9750-compatible terminals in full screen mode is possible. The Unicode function is available on all supported data terminals.