

FUJITSU Software BS2000 SDF-P

\*06  
\*06

Version 2.5G  
July 2017

Readme-file

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.

\*06

© 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.

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Program interfaces for the user (manual chapter 13.2 update)</b>	<b>4</b>
2.1	DELVAR	5
<b>3</b>	<b>Predefined functions</b>	<b>7</b>
3.1	Extension of LOWER-CASE () and UPPER-CASE () built-in functions	7
3.2	IS-EMPTY-FILE ()	8
3.3	PROC-Name ()	8
<b>4</b>	<b>Commands</b>	<b>8</b>
4.1	EXECUTE-CMD	8
4.2	ASSIGN-STREAM (Manual chapter 7.3.1 update)	8
4.3	READ-VARIABLE (see page 705 in the manual)	9
4.4	SHOW-VARIABLE (see page 758 in the manual)	9
4.5	SET-PROCEDURE-OPTIONS (see page 734 in the manual)	10
4.6	SELECT-VARIABLE-ELEMENTS (see page 728 in the manual)	12
4.7	CLOSE-VARIABLE-CONTAINER (see page 581 in the manual)	13
4.8	DECLARE-PARAMETER (see page 603 in the manual)	13
4.9	READ-VARIABLE (see page 714 in the manual)	13
<b>5</b>	<b>Messages (manual chapter 17 update)</b>	<b>14</b>

# 1 Introduction

\*05 This README file describes the changes/extensions valid for V2.5A, V2.5B, V2.5C, V2.5D, V2.5E and  
\*05 V2.5F.

\*01 Changes involved with SDF-P V2.5B are marked with \*01 in the left margin.

\*02 Changes involved with SDF-P V2.5C are marked with \*02 in the left margin.

\*03 Changes involved with SDF-P V2.5D are marked with \*03 in the left margin.

\*04 Changes involved with SDF-P V2.5E are marked with \*04 in the left margin.

\*05 Changes involved with SDF-P V2.5F are marked with \*05 in the left margin.

\*06 Changes involved with SDF-P V2.5G are marked with \*05 in the left margin.

The changes/extensions refer to the following manual:

\*03 SDF-P V2.4A

\*03 Edition May 2007

\*03 Order number: U6442-J-Z125-6-76

## Summary of changes/extensions in SDF-P V2.5A

Commands:

ASSIGN-STREAM	new constraints in FHS stream server assignation.
READ-VARIABLE	new operand type for library element version.
SHOW-VARIABLE	new operand type for library element version.

## \*01 Summary of changes/extensions in SDF-P V2.5B

\*01

\*01 Predefined functions:

\*01

\*01 UPPER-CASE new parameter CODED-CHARACTER-SET

\*01 LOWER-CASE new parameter CODED-CHARACTER-SET

\*01

\*01 Commands:

\*01

\*01 SET-PROCEDURE-OPTIONS new operand TRANSLATION-CCS to preset UPPER-CASE and  
\*01 LOWER-CASE built-in functions

## \*02 Summary of changes/extensions in SDF-P V2.5C

\*02

\*02 Commands:

\*02

\*02 SELECT-VARIABLE-ELEMENTS

\*02 HEADER-LINE up to 240 characters

\*02 new operand LENGTH in structure SELECTION-CODE=\*YES  
\*02 to set the width of the selection mark column.

\*02

\*02 SET-PROCEDURE-OPTIONS Note concerning the value \*SPIN-OFF-COMPATIBLE of the  
\*02 operand ERROR-MECHANISM.

## \*03 Summary of changes/extensions in SDF-P V2.5D

\*03

\*03 Predefined function:

\*03

\*03 IS-EMPTY-FILE() support of ISAM files

\*04 Summary of changes/extensions in SDF-P V2.5E  
 \*04  
 \*04 Command:  
 \*04  
 \*04 CLOSE-VARIABLE-CONTAINER  
 \*04 Deletion of variables declared in the variable container.

\*05 Summary of changes/extensions in SDF-P V2.5F  
 \*05  
 \*05 Macro:  
 \*05  
 \*05 DELVAR Deletion of variables from programs running in not-S-  
 \*05 procedures.  
 \*05  
 \*05 Messages:  
 \*05  
 \*05 SDP0201, SDP0203, SDP0204, SDP0207, SDP0215  
 \*05 Report command line number of control structure start.

\*06 Summary of changes/extensions in SDF-P V2.5F  
 \*06  
 \*06 Predefined function:  
 \*06  
 \*06 PROC-NAME() return current procedure name

Only the new, changed or extended operands and operand values are described in the following sections. The position of the operands within the command syntax can however be seen.

## 2 Program interfaces for the user (manual chapter 13.2 update)

\*05 SDF-P provides the assembler programmer with the following interfaces:

Macro	Function
DELVAR	Deletes simple and complex variables

\*05 The PUTVAR, GETVAR, DELVAR, SHOWSSA, TRANSVV and VARINF macros can be used to  
 \*05 address S variables, i.e. variables that can also be addressed via the SDF-P command interface.  
 \*05

\*05 **2.1 DELVAR**

\*05 The DELVAR macro deletes a variable. DELVAR can be used with simple variables or complex  
 \*05 variables, but not with elements of complex variables. I.e. structure or list elements cannot be deleted.

Operation	Operands
DELVAR	MF = E ,PARAM = <name 1..8> / (<integer 1..15> )
	MF = D ,PREFIX = D / prefix
	MF = C ,PREFIX = D / prefix ,MACID = ELV / macid
	MF = L ,NAMLEN = <integer 1..255> ,NAMADR = <name 1..8> ,SCOPE = *VISIBLE / *TASKONLY ,CALLER = USER / SYSTEM

\*05 **Operands**

\*05 **MF = E**  
 \*05 Execute format of the macro call; generates an SVC.

\*05 **PARAM**  
 \*05 Designates the address of the operand list that is evaluated for the macro call (address of macro  
 \*05 call with MF = L).

\*05 **= <name 1..8>**  
 \*05 Designates the symbolic address of the operand list.

\*05 **= (<integer 1..15>)**  
 \*05 Designates the register that contains the address of the operand list.

\*05 **MF = D**  
 \*05 **DSECT format of the macro call; generates a DSECT for the operand list. The names generated**  
 \*05 **begin** with the letter D; they can be modified with PREFIX.

\*05 **PREFIX = D / prefix**  
 \*05 Defines the first character of the generated names.  
 \*05 Default: the generated names begin with the letter D.

\*05 **MF = C**  
 \*05 C format of the macro call; generates an operand list whose symbolic names begin with the string  
 \*05 DELV. They can be changed with PREFIX and MACID.

\*05 **PREFIX = D / prefix**  
 \*05 Defines the first character of the generated names.  
 \*05 Default: the generated names begin with the letter D.

\*05 **MACID = ELV / macid**  
 \*05 A string of up to three characters that replaces characters 2 to 4 of the generated names.  
 \*05 Default: ELV

\*05 **MF = L**  
 \*05 LIST format of the macro call; generates the operand list for the macro call with MF = E (Execute  
 \*05 format); the macro call must be addressable by means of a symbolic address.

\*05 **NAMLEN = <integer 1..255>**  
 \*05 Designates the length of the variable name.

\*05 **NAMADR = <name 1..8>**  
 \*05 Designates the symbolic name of the variable name address.

\*05 **SCOPE**  
 \*05 Defines the scope of the variable.

\*05 **= \*VISIBLE**  
 \*05 The variable is created as a procedure-local variable.

\*05 **= \*TASKONLY**  
 \*05 The variable is created as a task-global variable.

\*05 **Return codes**  
 \*05 The table below lists the return codes in hexadecimal notation.

Subcode2	Subcode1	Maincode	Meaning
00	00	0000	Macro call was successful; no error
01	00	0000	Variable not declared
00	01	0001	Parameter error
00	01	0002	Syntax error in variable name
00	40	0019	no delete possible
00	40	001A	delete failed
00	01	FFFF	Unknown unit or function number
00	02	FFFF	Function not available
00	03	FFFF	Wrong version of operand list

### 3 Predefined functions

#### 3.1 Extension of LOWER-CASE () and UPPER-CASE () built-in functions

Domain: **Conversion functions**

The LOWER-CASE() and UPPER-CASE() built-in functions translate the input string into lower cases or upper cases for the characters defined in the specified coded character set. The scope of the translation can be extended by using the coded character set currently active for the task or a specific coded character set.

**Format**

LOWER-CASE () UPPER-CASE ()
.... ,CODED-CHARACTER-SET = <u>*STD</u> / *EDF03IRV / *CURRENT / string_expression

**Input parameters**

**CODED-CHARACTER-SET =**

Indicates which coded character set (CCS) is used to convert the input string to lower or upper cases. If the CCS is not supported or not known, or if the current CCS is not EDF03IRV and XHCS is not available, the operation is rejected. The CCS must be an 8-bit code. 16-bit codes (UTFE, UNICODE) are not supported.

**CODED-CHARACTER-SET = \*STD**

Default value. The settings defined at procedure level apply (operand TRANSLATION-CCS of command /SET-PROCEDURE-OPTIONS). Outside procedure, \*STD is the compatible \*EDF03IRV value if no external setting changes this into \*CURRENT. The external setting is an optional REP for SDF-P-BASYS. This REP applies globally for all tasks.

**CODED-CHARACTER-SET = \*EDF03IRV**

Only the characters defined in the basic coded character set are translated (compatible behavior)

**CODED-CHARACTER-SET = \*CURRENT**

The characters are translated according to the coded character set currently active for the task.

**CODED-CHARACTER-SET = string\_expression**

The characters are translated according to the specified coded character set.

**Remark**

when the built-in function processes string literals from a procedure, the input has already been translated in EBCDIC compatible format by the system. A coded-character-set specification (explicit name or implicit via \*current) is only meaningful for extended EBCDIC character sets (e.g. EDF041 accented characters) or for values stored in variables which are coded in the specified character set (e.g. using the built-in function FROM-X-LITERAL).

Additional messages relating to CODED-CHARACTER-SET:

- SDP0539 CODED-CHARACTER-SET '(&00)' NOT SUPPORTED
- SDP0540 CODED-CHARACTER-SET '(&00)' INVALID
- SDP0541 CODED-CHARACTER-SET '(&00)' UNKNOWN
- SDP0542 CODED-CHARACTER-SET FUNCTION NOT SUPPORTED BY CURRENT VERSION OF SUBSYSTEM 'XHCS'
- SDP0543 SUBSYSTEM 'XHCS' NOT AVAILABLE

\*03 **3.2 IS-EMPTY-FILE ( )**

\*03

\*03

\*03 Domain: **Environment information**

\*03

\*03 The IS-EMPTY-FILE ( ) function checks whether a SAM or ISAM file is empty i.e. contains no data.

\*06 **3.3 PROC-Name ( )**

\*06

\*06

\*06 Domain: **Environment information**

\*06

\*06 The PROC-NAME ( ) function returns the name of the procedure from which the function was called..

## 4 Commands

### 4.1 EXECUTE-CMD

#### Format

EXECUTE-CMD
CMD = <c-string 0..1800 with-low> / <text 0..1800 with-low>
...

### 4.2 ASSIGN-STREAM (Manual chapter 7.3.1 update)

Before either of these options can be used, the variable stream used by the program must be assigned to FHS, using the SERVER operand of the ASSIGN-STREAM command. The standard stream (i.e. SYSINF, SYSMSG and SYSVAR) could no more be assigned to the FHS server. This limitation exists because FHS might output structured message too and might introduce deadlock.



### 4.3 READ-VARIABLE (see page 705 in the manual)

#### Format

READ-VARIABLE
<pre> ... , INPUT = *LIBRARY-ELEMENT (...)     ...     *LIBRARY-ELEMENT (...)           ELEMENT = &lt;composed-name_1..64_with-underscore&gt; (...)             &lt;composed-name_1..64_with-underscore&gt; (...)                   VERSION = ... / &lt;composed-name_1..24_with-underscore&gt; </pre>

#### Operands

**INPUT = \*LIBRARY-ELEMENT (...)**

**ELEMENT = <composed-name\_1..64\_with-underscore> (...)**

Name of the library element.

**VERSION = \*HIGHEST-EXISTING / <composed-name\_1..24\_with-underscore>**

Version of the library element.

### 4.4 SHOW-VARIABLE (see page 758 in the manual)

#### Format

SHOW-VARIABLE
<pre> ... , OUTPUT = ... / *LIBRARY-ELEMENT (...)     *LIBRARY-ELEMENT (...)           ...           , ELEMENT = &lt;composed-name_1..64_with-underscore&gt; (...)             composed-name_1..64_with-underscore(...)                   VERSION = ... / &lt;composed-name_1..24_with-underscore&gt; </pre>

#### Operands

**OUTPUT = \*LIBRARY-ELEMENT (...)**

**ELEMENT = <composed-name\_1..64\_with-underscore> (...)**

Name of the element.

**VERSION = <composed-name\_1..24\_with-underscore>**

Selects the specified version number.

\*01 **4.5 SET-PROCEDURE-OPTIONS (see page 734 in the manual)**

\*01  
\*01  
\*01

**Format**

\*01  
\*01  
\*01

SET-PROCEDURE-OPTIONS

\*02  
\*02

...  
, ERROR-MECHANISM = \*SPIN-OFF-COMPATIBLE / \*BY-RETURNCODE

\*01  
\*01

...  
, TRANSLATION-CCS = \*STD / \*EDF03IRV / \*CURRENT

\*01  
\*01

**Operands**

\*01  
\*01

**TRANSLATION-CCS =**

Determines the coded-character-set used for translations to lowercase or uppercase performed by SDF-P built-in functions UPPER-CASE and LOWER-CASE.

\*01  
\*01

If the current CCS is not supported or not known, or if the current CCS is not EDF03IRV and the XHCS subsystem is not started, the operation is rejected. The CCS must be a 8-bit code. 16-bit codes are not supported.

\*01  
\*01

**TRANSLATION-CCS = \*STD**

Use the basic EDF03IRV (compatible behavior) if no specific external settings changes this default behavior into \*CURRENT. The external setting is an optional REP for SDF-P-BASYS. This REP is global to all tasks.

\*01  
\*01

**TRANSLATION-CCS = \*EDF03IRV**

Force compatible value (use basic EDF03IRV).

\*01  
\*01

**TRANSLATION-CCS = \*CURRENT**

Use the CCS currently assigned to the input stream.

\*01  
\*01

**Remark**

when the built-in function processes string literals from a procedure, the input has already been translated in EBCDIC compatible format by the system. A coded-character-set specification (explicit name or implicit via \*current) is only meaningful for extended EBCDIC character sets (e.g. EDF041 accented characters) or for values stored in variables which are coded in the specified character set (e.g. using the built-in function FROM-X-LITERAL).

\*01  
\*01

**Additional error messages**

\*01  
\*01  
\*01  
\*01  
\*01

- SDP0539 CODED-CHARACTER-SET '(&00)' NOT SUPPORTED
- SDP0541 CODED-CHARACTER-SET '(&00)' UNKNOWN
- SDP0542 CODED-CHARACTER-SET FUNCTION NOT SUPPORTED BY CURRENT VERSION OF SUBSYSTEM 'XHCS'
- SDP0543 SUBSYSTEM 'XHCS' NOT AVAILABLE

\*02 **ERROR-MECHANISM = \*SPIN-OFF-COMPATIBLE**

\*02 The description of the value \*SPIN-OFF-COMPATIBLE for the operand ERROR-MECHANISM is  
\*02 supplemented by the following note:

\*02 **Note**  
\*02 ERROR-MECHANISM = \*SPIN-OFF-COMPATIBLE does only affect the S procedure in which it is  
\*02 declared. Error propagation into the calling S-procedure is solely governed by Subcode1. This especially  
\*02 means that the implicit exit from a \*SPIN-OFF-COMPATIBLE S- procedure while spin-off is active and  
\*02 Subcode1 == 0 will be considered as non-error in the calling \*SPIN-OFF-COMPATIBLE S-procedure or  
\*02 non-S procedure and the spin-off will become deactivated there.



Explanation: one of following operands or values requires a more recent version of SDF:

```
*02 MESSAGE
*02 TITLE greater than 80 characters
*02 SELECTION-CODE LENGTH greater than 1
*02 To continue processing, the request has been downgraded to display a screen without
*02 message and/or with title truncated to 80 characters and/or with selection code of 1
*02 column.
*02
*02 Action: Contact the system administrator to upgrade the SDF subsystem to a version that
*02 supports these operands or values.
```

#### **4.7 CLOSE-VARIABLE-CONTAINER (see page 581 in the manual)**

```
*04 ...
*04 Note
*04 If a variable container is closed before its scope ceases to exist, the variables declared in this container
*04 are deleted.
```

#### **4.8 DECLARE-PARAMETER (see page 603 in the manual)**

```
*05 ...
*05 Example 1: Demonstration of prompting
*05 ...
*05 /SET-PROCEDURE-OPTIONS
*05 /DECLARE-PARAMETER NAME(INITIAL-VALUE=*PROMPT-
*05 / (PROMPT-STRING='ENTER THE NAME OF THE COLOR TO BE TRANSLATED',-
*05 / DEFAULT-VALUE='ROT'))
*05 /COLOR=TRANSLATE(String=NAME , -
*05 /,WHEN1='ROT', THEN1='RED' , -
*05 /,WHEN2='GRUEN', THEN2='GREEN' , -
*05 /,WHEN3='BLAU', THEN3='BLUE' , -
*05 /,WHEN4='GELB', THEN4='YELLOW' , -
*05 /,WHEN5='SCHWARZ',THEN5='BLACK' , -
*05 /,WHEN6='WEISS', THEN6='WHITE' , -
*05 /,ELSE='UNKNOWN')
*05 /SHOW-VAR NAME
*05 /SHOW-VAR COLOR
```

#### **4.9 READ-VARIABLE (see page 714 in the manual)**

```
*05 ...
*05 Example 5
*05 The data file SDF-P-FILE contains the following records:
*05 ...
```

## 5 Messages (manual chapter 17 update)

\*05 SDP0201 INVALID BLOCK-CLOSING COMMAND USED FOR CONTROL STRUCTURE  
\*05 STARTING AT COMMAND LINE: (&00)  
\*05  
\*05 SDP0203 CURRENT BLOCK NOT AN IF BLOCK FOR CONTROL STRUCTURE STARTING AT  
\*05 COMMAND LINE: (&00)  
\*05  
\*05 SDP0204 /ELSE COMMAND ALREADY PRESENT AT COMMAND LINE: (&00)  
\*05  
\*05 SDP0207 NOT ALL CONTROL STRUCTURES CLOSED FROM CONTROL STRUCTURE  
\*05 STARTING AT COMMAND LINE: (&00)  
\*05  
\*05 SDP0215 LABEL OF BLOCK-TERMINATING COMMAND DOES NOT MATCH LABEL OF  
\*05 BLOCK-INITIATING COMMAND FOR CONTROL STRUCTURE STARTING AT  
\*05 COMMAND LINE: (&00)