FUJITSU Software
NetCOBOL V11.0.0

Release Notes

Linux(64)

Preface

This book explains Release Note of NetCOBOL.

About the alias of the product

The name of the product described in this book is written for short as follows.

Product Name	Abbreviation
Red Hat(R) Enterprise Linux(R) 7 (for Intel64)	Linux
Red Hat(R) Enterprise Linux(R) 6 (for Intel64)	
Microsoft(R) Windows Server(R) 2012 R2 Datacenter	Windows Server 2012 R2
Microsoft(R) Windows Server(R) 2012 R2 Standard	
Microsoft(R) Windows Server(R) 2012 R2 Essentials	
Microsoft(R) Windows Server(R) 2012 R2 Foundation	
Microsoft(R) Windows Server(R) 2012 Datacenter	Windows Server 2012
Microsoft(R) Windows Server(R) 2012 Standard	
Microsoft(R) Windows Server(R) 2012 Essentials	
Microsoft(R) Windows Server(R) 2012 Foundation	
Microsoft(R) Windows Server(R) 2008 R2 Foundation	Windows Server 2008 R2
Microsoft(R) Windows Server(R) 2008 R2 Standard	
Microsoft(R) Windows Server(R) 2008 R2 Enterprise	
Microsoft(R) Windows Server(R) 2008 R2 Datacenter	
Windows(R) 8.1	Windows 8.1
Windows(R) 8.1 Pro	
Windows(R) 8.1 Enterprise	
Windows(R) 8	Windows 8
Windows(R) 8 Pro	
Windows(R) 8 Enterprise	
Windows(R) 7 Home Premium	Windows 7
Windows(R) 7 Professional	
Windows(R) 7 Enterprise	
Windows(R) 7 Ultimate	
Microsoft(R) Visual C++(R) development system	Visual C++
Microsoft(R) Visual Basic(R) programming system	Visual Basic

Purpose of this documentation

This documentation explains the additional function, the bug fix and the interchangeable information from the old edition.

It has aimed can the smoother shift of the customer who was using the old edition to this product.

Documentation audience

This document is for customers who are using old edition of this products and intended for shift or shifting to this product.

Positioning of this documentation

This document explains only fragments of the additional function, the bug fix, and the interchangeable information from the old edition. For details and to get the overall look of the product, refer to each product manual.

Related manuals

The manual of this product and a related product includes the following manuals besides this document.

- NetCOBOL Language Reference

- NetCOBOL Syntax Samples

- NetCOBOL User's Guide

- NetCOBOL COBOL File Access Subroutines User's Guide

- NetCOBOL Messages

- PowerFORM Runtime Reference

- J Adapter Class Generator User's Guide

- PowerBSORT User's Guide

About the registered trademarks

- NetCOBOL is a trademark or registered trademark of Fujitsu Limited or its subsidiaries in the United States or other countries or in both.

- Microsoft, Windows and Windows Server are trademarks or registered trademarks of Microsoft Corporation.

- Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners. Oracle Solaris might be described as Solaris, Solaris Operating System, or Solaris OS.

- Other brand and product names are trademarks or registered trademarks of their respective owners.

- Trademark indications are omitted for some system and product names described in this manual.

Export Regulation

Exportation/release of this document may require necessary procedures in accordance with the regulations of your resident country and/or US export control laws.

Contents

Chapter 1 Outline of Additional Functionality

New product functionality and content is outlined below by version and level.

1.1 All Products

The following Information applies to all products.

Table 1.1 Outline of additional functionality in NetCOBOL V11

NO.	V/L	Function Name	Content	Location in Manual
1	V11.0.0	Fujitsu Common Tools	Fujitsu Middleware Installation System and FJQSS (Information Collection Tool)	- Overview Software Release Guide - FJQSS User's Guide

Note

The V/L column lists the version and level of NetCOBOL Enterprise Edition.

1.2 NetCOBOL

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

Table 1.2 Outline of additional function of NetCOBOL V11

NO.	V/L	Function name	Content	Location in Manual
1	V11.0.0	UTF-32 support	Encoding form UTF-32 and Shift-JIS can now be used.	NetCOBOL User's Guide - 20 Unicode
2	V11.0.0	Hadoop support	The Hadoop integration function can now be used.	Hadoop Integration Function User's Guide
3	V11.0.0	Named pipes of COBOL File Access Routines	Named pipes of COBOL File Access Routines is supported.	COBOL File Access Subroutines User's Guide - 3.27 Named pipe

Table 1.3 Outline of additional function of NetCOBOL V10

NO.	V/L	Function name	Content	Description place of manual
1	V10.4.0	31-digit support	Up to 31 digits can now be used for numeric items.	NetCOBOL Language Reference - 1.2.3.1 Numeric Literal - 1.7 Operation Mode - 5.4.8 PICTURE Clause - 5.4.16 USAGE Clause - 8.2.2 Host Variable Definitions - Appendix D Intermediate Results NetCOBOL User's Guide

				- 6.8 Other File Functions
				- A.2.2 ARITHMETIC
				NetCOBOL Messages
2	V10.4.0	Recursive calls	Programs can now be called recursively. Data items used by the recursive program are defined in the LOCAL-STORAGE SECTION.	NetCOBOL Language Reference
				- 1.3.5.2 Slack Bit
				- 2.1 Nucleus
				- 2.3 Input-Output Facility
				- Inter-program Communication Module
				- 2.3.3 External Attribute and Internal Attribute
				- 2.3.5 Program Activation
				- 2.3.7 Initial State of a Program
				- 2.3.8 Recursive Attribute of the Program
				- 3.1.1 Program-Id Paragraph
				- 4.3.1.1 ACCESS MODE Clause
				- 4.3.1.3 ASSIGN Clause (Sequential File, Relative File, and Indexed File)
				- 4.3.1.8 FILE STATUS Clause
				- 4.3.1.9 FORMAT clause
				- 4.3.1.10 GROUP clause
				- 4.3.1.23 PROCESSING MODE clause
				- 4.3.1.30 SELECTED FUNCTION clause
				- 4.3.1.32 SYMBOLIC DESTINATION clause
				- 4.3.1.33 UNIT CONTROL clause
				- 5.1 Composition of the Data Division
				- 5.2.9 RECORD clause
				- 5.4 Data Description Entry
				- 5.4.3 CHARACTER TYPE clause
				- 5.4.13 SYNCHRONIZED Clause
				- 5.4.17 VALUE clause
				- 6.4.7 CALL statement
				- 8.2.1 Embedded SQL Declare Section
				- 10.2.3 Data Description Entry

				- Appendix A List of Reserved Words NetCOBOL User's Guide - 8.1 Calling Relationship Types
3	V10.4.0	Unified specification of high-speed file processing	High-speed file processing can be specified for record sequential and line sequential files with one configuration specification as opposed to requiring specification for each file.	NetCOBOL User's Guide - 4.1.1 Runtime Environment - 6.8.1.2 High-Speed File Processing - Batch specification NetCOBOL Messages

4	V10.4.0	Syslog output	The Syslog can be specified as the output destination for DISPLAY statements.	NetCOBOL User's Guide
				
				- 4.1.1 Runtime Environment
				- 9.1 ACCEPT/DISPLAY Function
				- 9.1.10 Programs Using Syslog
				- Appendix E Environment Variable List
				
				  - CBR_DISPLAY_CONSOLE _EVENTLOG_LEVEL
				  - CBR_DISPLAY_CONSOLE _EVENTLOG_SRCNAME
				  - CBR_DISPLAY_CONSOLE _OUTPUT
				  - CBR_DISPLAY_SYSERR_E VENTLOG_LEVEL
				  - CBR_DISPLAY_SYSERR_E VENTLOG_SRCNAME
				  - CBR_DISPLAY_SYSERR_O UTPUT
				  - CBR_DISPLAY_SYSOUT_E VENTLOG_LEVEL
				  - CBR_DISPLAY_SYSOUT_E VENTLOG_SRCNAME
				  - CBR_DISPLAY_SYSOUT_ OUTPUT)

5	V10.4.0	CHECK function suppress option	Runtime options are provided to suppress individual CHECK functions. - nocb - noci - nocn - nocp	NetCOBOL User's Guide - 4.2.2 Specifying Runtime Options - 4.2.2.2 [ccount | { noc | nocb | noci | nocn | nocp }] - 5.3.2 Output Messages
6	V10.4.0	TRACE function suppress option	Runtime option is provided that suppresses the TRACE function. - nor	NetCOBOL User's Guide - 4.2.2 Specifying Runtime Options

N O.	V/L	Function name	Content	Location in Manual
				- 4.2.2.1 [r count | nor] (Set the trace data limit, and suppress the TRACE function)
7	V10.4.0	TRACE information file for separate processes	Separate TRACE information files can be generated for different processes to avoid conflicts in writing to the file.	NetCOBOL User's Guide - 4.1.1 Runtime Environment - 5.2.2 Trace Information

1.3 PowerFORM

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

Table 1.4 Outline of additional function of PowerFORM V11

N O.	V/L	Function name	Content	Location in Manual
1	V11.0.0	31-digit support	Up to 31 digits can now be used for numeric fields in COBOL applications.	PowerFORM Runtime Reference - Output field declarations
2	V11.0.0	Form exporting	The Form can now be outputted to PDF.	PowerFORM Runtime Reference - Chapter 4. Extended Functions - Appendix C. Printer Information File
3	V11.0.0	UTC (Coordinated Universal Time) support	UTC (Coordinated Universal Time) can be specified.	PowerFORM Runtime Reference - Appendix C. Printer Information File
4	V11.0.0	Enhanced print features	The Form Descriptor (free frame) can be used.	PowerFORM Runtime Reference - Chapter 4. Extended Functions
5	V11.0.0	Enhanced print features	The new function added in PowerFORM V11 can be used.	PowerFORM Runtime Reference - What's New in PowerFORM V11?

1.4 PowerBSORT

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

Outline of additional function of PowerBSORT V7

N O.	V/L	Function name	Content	Location in Manual
1	V7.0.0	Data format	The Unicode UTF-32 form was supported as data format.	User's Guide - 1.4 Environment variables and Startup file - 1.6.3 Data format

N O.	V/L	Function name	Content	Location in Manual
				- 1.6.4 Data forms that can be specified in each field
				- 3.2.16 Input code system option (-q)
				- 3.2.26 COBOL file index specification option (-X)
				- 4.2.7.3 icode operand
				- 4.2.8.10 idxkey operand
				- 6.5.1 BSRTPRIM structure
				- 6.5.10 BSIDXKEY structure
2	V7.0.0	Record summation option	The length of data formats that can be specified for the binary file has been enhanced	User's Guide - 1.6.4.2 Data formats that can be specified in the summation field
3	V7.0.0	Record selection option	The length of data formats that can be specified by a literal value has been enhanced.	User's Guide - 1.6.4.3 Data formats that can be specified in the selection field
4	V7.0.0	Record reconstruction option	The length of data formats that can be specified by a literal value has been enhanced.	User's Guide - 1.6.4.4 Data formats that can be specified in the literal value of the reconstruction field
5	V7.0.0	File types	The physical sequential file format supported by COBOL85 that operated on the Fujitsu mainframe was supported.	User's Guide - 1.4.2 Startup file - 1.5 PowerBSORT Input Output Environment - 3.2.3 I/O file system option (-F) - 4.2.5.3 filesys operand - 6.5.8 BSFSYS structure - B.8 Important Notes about the NetCOBOL File system

Outline of additional function of PowerBSORT V6

N O.	V/L	Function name	Content	Location in Manual
1	V6.1.0	Data format	The external decimal number of Micro Focus COBOL form was supported as data format.	User's Guide - 3.2.16 Input code system option (-q) - 4.2.7.3 icode operand - 6.5.1 BSRTPRIM structure

1.5 J Adapter Class Generator

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

Outline of additional function of J Adapter Class Generator V11

NO.	V/L	Function name	Content	Location in Manual
1	V11.0.0	Encoding form Shift-JIS support.	Encoding form Shift-JIS can now be used.	J Adapter Class Generator User's Guide

Chapter 2 Information on interchangeability

Here, information on the interchangeability changed from the version and level before is described.

2.1 NetCOBOL Development Environment

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

2.1.1 Interchangeable information regarding bug fixes

Here, it explains bug fixes corrected by NetCOBOL V10.1.0 and later version in which the operation of the NetCOBOL development environment changes in the following tables.

Table 2.1 Interchangeable information on bug fixes of NetCOBOL Development Environment

NO.	V/L(*)	P number	Content of change
1	V10.1.0 to V10.4.0	PG76651	When all of the following conditions are met while executing a COBOL program, a size comparison of national items varying in length (national language edited item, including the intrinsic function) was incorrect. 1. Compile option RCS (UCS2,LE) or RCS (UTF16,LE) is specified. 2. And, there is a size comparison of a National item and a National item or National language character constant. 3. And, at least one side is an item with the ANY LENGTH clause is specified or reference modification. 4. And, the length of the compared targets differs.

2	V10.1.0 to V10.4.0	PH02265	Under the following conditions, a compilation error may not be generated.
			
			1. Concatenation expression is described in the external name of program, class or method.
			
			2. And, one of the following is described at the beginning of the concatenation expression.
			  - Character constant
			  - National character constant
			
			3. And, when one of the following is described in the concatenation expression.
			  - Hexadecimal character constant
			  - National hexadecimal character constant

3	V10.1.0	PG77383	Under the following conditions, the execution results are incorrect. 1. The OPTIMIZE compiler option is specified (*1). 2. And, there is a statement that sets packed decimal item to numeric edited data item. 3. And, the number of digits for integer part of packed decimal item and numeric edited data item are the same and neither have a decimal-part. 4. And, the numeric edited data item specifies only zero suppression for the edit (Only '9', 'Z', and '*' are used for PICTURE character-string). 5. And, before the statement in 2 (*2), there is a statement that sets data item (or intermediate result) to packed decimal item. 6. And, the relation between number of digits for the data item in 5 (or intermediate result) and number of digits for the packed decimal part in 2 is the following :

NO.	V/L(*)	P number	Content of change
| | | | ```
-----------------------------------------------------
Data item (or intermediate result)    Packed decimal item
------------------------------------  ---------------
            2                                      3
            4                                      5
            6                                      7
            8                                      9
           10                                     11
           12                                     13
           14                                     15
           16                                     17

-----------------------------------------------------
``` |
			7. And, the data item in 5 (or intermediate result) and the packed decimal item in 2 don't have the decimal-part.
			*1: The default is NOOPTIMIZE.
			*2: MOVE statement, COMPUTE statement, and so on.
4	V10.1.0	PG78440	Under the following conditions, the execution results are incorrect.
			1. The BINARY(BYTE) or BINARY(WORD, MLBOFF) compiler option is specified, along with
			2. The following intrinsic functions:
			[A]
			- FUNCTION MAX
			- FUNCTION MIN
			- FUNCTION MEAN
			- FUNCTION MEDIAN
			- FUNCTION RANGE
			[B]
			- FUNCTION ANNUITY
			- FUNCTION NUMVAL
			- FUNCTION NUMVAL-C
			- FUNCTION RANDOM
			3. And, the arguments of the function in 2) above are fixed point of 9 digits or less.
			4. And, the number of the arguments in [A] above is 4 or more.
5	V10.1.0	PG75097	Under the following conditions, the COBOL application running under Red Hat Enterprise Linux (for Intel64) cannot be remotely debugged from the NetCOBOL Studio of Windows edition. "Error occurred in debugger or engine. (65535)" is displayed.
			1. The server is Red Hat Enterprise Linux (for Intel64) or Red Hat Enterprise Linux (for Itanium).
			2. And, installed the following products in the server.
			- Alchemy NetCOBOL Enterprise Edition V10.1.0 for Linux (64bit)
			3. And, gdb used on the server is gdb-7.0(*) or later.
			*: Red Hat Enterprise Linux 5.5 is installed or gdb-7.0 is installed alone.

*: V/L is a scope where the bug exists.

2.2 NetCOBOL Runtime Environment

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

2.2.1 About the BOM of the UTF-8 file

Content

Operation differs when input the UTF-8 file with BOM.

- Before V11.0

    BOM is read as part of the data when it is included in the UTF-8 file.

    When a valid keyword is described at the first line of the input file, but the file is read with BOM added at the beginning of the keyword, the keyword might be invalid.

- After V11.0

    BOM is identified as UTF-8 identification code and it is skipped.

    When a valid keyword is described at the first line of the input file, the keyword is considered as valid as the BOM in the beginning of the file is skipped.

Influence

When the input resource file is UTF-8 with BOM, the content described in the first line is valid after V11.0 however the content described in the first line is invalid before V11.0.

There is no change in the operation of the application when the first line is a comment line or null line. The input resource when executing it becoming an object is as follows.

- Runtime initialization file

- Entry information file

- Class information file

- Print information file

- Printer information file

Action

Please delete or make the first line a comment.

2.2.2 Interchangeable information regarding bug fixes

Here, it explains bug fixes corrected by NetCOBOL V10.1.0 or later version in which the operation of the NetCOBOL development environment changes in the following tables.

Table 2.2 Interchangeable information on bug fixes of NetCOBOL Runtime Environment

NO.	V/L(*)	P number	Content of change
1	V10.1.0 to V10.4.0	PG76651	When all of the following conditions are met while executing a COBOL program, a size comparison of national items varying in length (national language edited item, including the intrinsic function) was incorrect. 1. Compile option RCS (UCS2,LE) or RCS (UTF16,LE) is specified. 2. And, there is a size comparison of a National item and a National item or National language character constant. 3. And, at least one side is an item with the ANY LENGTH clause is specified or reference modification.

NO.	V/L(*)	P number	Content of change
			4. And, the length of the compared targets differs.
2	V10.1.0	PG72597	Under the following conditions, the WRITE AFTER ADVANCING PAGE is not valid at execution time. 1. The line sequential file is used. 2. The line sequential file is linked with the external file handler. 3. After an OPEN statement is executed for file in 2, the WRITE statement of the AFTER ADVANCING PAGE phrase is executed.

3	V10.1.0	PG73346	Under the following conditions, at execution, unnecessary zero string is set to four subordinate position digits of the hexadecimal character of eight digits buried under execution time message JMP0320I-I/JMP0320I-U
			
			1. File operation using COBOL file system (except BSAM specification) is done.
			
			2. Either of the following I/O statements is executed.
			
			   - READ statement
			
			   - WRITE statement
			
			   - REWRITE statement
			
			   - DELETE statement
			
			   - START statement
			
			3. The execution of I/O statement failed.

4	V10.1.0	PG75097	Under the following conditions, the COBOL application running under Red Hat Enterprise Linux (for Intel64) cannot be remotely debugged from the NetCOBOL Studio (Windows) edition. "Error occurred in debugger or engine. (65535)" is displayed. 1. The server is Red Hat Enterprise Linux (for Intel64) or Red Hat Enterprise Linux (for Itanium). 2. Installed the following products in the server. - Alchemy NetCOBOL Enterprise Edition V10.1.0 for Linux (64bit) 3. Gdb used on the server is gdb-7.0(*) or later. *: Red Hat Enterprise Linux 5.5 is installed or gdb-7.0 is installed alone.
5	V10.1.0	PG87250	Under the following conditions, the MOVE result by internal Bool item is incorrect. Note: The move result is different according to the state of the area allocated just behind the sending data item. [Condition 1] 1. Move from internal Boolean item to internal Boolean item. 2. Data start position (*) of the sending data item is different from data start position of the receiving data item. 3. [a] + [b] > 32, or [c] + [d] > 32 a: Data start position (*) of the sending data item b: The length of sending data item or the length of receiving data item, whichever is smaller. c: Data start position (*) of the receiving data item d: The length of receiving data item 4. The receiving data item is subordinate to the group item.

NO.	V/L(*)	P number	Content of change

5. Data start position of the receiving data item is not one byte alignment of the group item in 4.

6. "Bit length of sending data item" < " Bit length from bit data starting location on the receiving side to the first byte boundary "

7. Bit data of sender data item is outside the byte boundary or Byte boundary exist between the start position of the sender's data item of paragraphs 6(*) (bit length till the byte boundary of the first recipient).

*: relative bit position in byte

Example of Condition 1

```
        DATA DIVISION.
        WORKING-STORAGE SECTION.
        01 DATA1.
          02 DATA1-1 PIC 1(5) BIT.
          02 DATA1-2 PIC 1(4) BIT.
          02 DATA1-3 PIC 1(7) BIT.
        01 DATA2.
          02 DATA2-1 PIC 1(1) BIT.
          02 DATA2-2 PIC 1(32) BIT.
        01 ANS-DATA  PIC 1(32) BIT VALUE B"1111".
        PROCEDURE DIVISION.
            MOVE X"FFFF" TO DATA1.
            MOVE DATA1-2 TO DATA2-2.
            IF DATA2-2 = ANS-DATA
              THEN DISPLAY "OK"
              ELSE DISPLAY "NG"  *> B"11111110-00000000"
            END-IF.
```

[Condition 2]

1. Move from internal Boolean item to internal Boolean item.

2. The sending data item or the receiving data item is referred by the subscript.

3. " The length of receiving data item " > 1

4. It is either the following.

   - "The length of sending data item" > 1

   - The receiving data item is referred by the subscript.

   - " The length of receiving data item " > 25

5. There is not the data starting location of the receiving data item on byte boundary.

6. "Bit length of sending data item" < " Bit length from bit data start position of first byte of the receiving data item

7. The bit data of the sending data item overlaps the byte boundary, or there is byte boundary between from start position of the sending data item to "Bit length from bit data start position of the receiving side to the first byte boundary" in 6.

Example of Condition 2

```
        DATA DIVISION.
        WORKING-STORAGE SECTION.
        01 DATA1.
          02 DATA1-1 PIC 1(5) BIT.
          02 DATA1-2 PIC 1(4) BIT.
          02 DATA1-3 PIC 1(7) BIT.
        01 DATA2.
```

NO.	V/L(*)	P number	Content of change

			<pre> 02 DATA2-1 PIC 1(10) BIT OCCURS 5 TIMES.
			01 ANS-DATA PIC 1(10) BIT VALUE B"1111".
			01 CNT PIC 9 VALUE 2.
			PROCEDURE DIVISION.
			    MOVE X"FFFF" TO DATA1.
			    MOVE DATA1-2 TO DATA2-1(CNT).
			    IF DATA2-1(CNT) = ANS-DATA
			      THEN DISPLAY "OK"
			      ELSE DISPLAY "NG" *> B"11111110-00000000"
			    END-IF.</pre>

*: V/L is a scope where the bug exists.

2.3 PowerFORM

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

There is no information on interchangeability.

2.4 PowerBSORT

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

2.4.1 Interchangeable information regarding bug fixes

Here, it explains bug fixes corrected by PowerBSORT V7 and later version in which the operation of the PowerBSORT changes in the following tables.

Table 2.3 Interchangeable information on bug fixes of PowerBSORT

NO.	V/L(*)	P number	Content of change
1	V6.0.0	PG76059	Under the following conditions, PowerBSORT may cause result error, infinite loop or abnormal termination. 1. The bsort command, the bsortex command or the BSORT function is used. 2. And, sort function is used. 3. And, the text file CSV format or text file TSV format is used. 4. And, the record summation option or the record selection option for the output files is specified. 5. And, the summation field or selection field is enclosed with a double quotation.
2	V6.0.0	PG76539	Under the following conditions, PowerBSORT may cause result error or abnormal termination. 1. The bsortex command is used. 2. And, the text file CSV format or text file TSV format is used. 3. And, the record reconstruction option for the output files is used.

*: V/L is a scope where the bug exists.

2.5 J Adapter Class Generator

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

There is no information on interchangeability.

Chapter 3 Program Fix List

In this software, the trouble that occurs by the version and level before is corrected.

This document explains the content of the correction corrected by this software of this version level.

3.1 NetCOBOL Development Environment

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

Table 3.1 Program fix list of NetCOBOL Development Environment

NO.	V/L(*)	P number	Content of change
1	V10.1.0 to V10.4.0	PG76651	When all of the following conditions are met while executing a COBOL program, a size comparison of national items varying in length (national language edited item, including the intrinsic function) was incorrect. 1. Compile option RCS (UCS2,LE) or RCS (UTF16,LE) is specified. 2. And, there is a size comparison of a National item and a National item or National language character constant. 3. And, at least one side is an item with the ANY LENGTH clause is specified or reference modification. 4. And, the length of the compared targets differs.
2	V10.1.0 to V10.4.0	PG88616	When [condition 1] or [condition 2] shown below occurs during the compilation of a COBOL program, there are times when the COBOL compiler outputs a wrong message such as the following. - JMN1019I-W line-no marked area should be '-', '*', '/', 'D' or space. Will be taken as space. - JMN1123I-S line-no not allowed word 'nnn' appeared.Will be disabled till next recognizable clause, paragraph, section or division. - JMN1255I-S line-no 01 without specifying the item level 02 - 49 level item is specified. Will be taken as 01 level. - JMN1356I-W line-no header part incorrect word 'nnn' is specified. Will be disabled till next paragraph or division. line-no: line number nnn: Number The above messages are examples and not a comprehensive list of all the possible messages. [condition 1] 1. The COBOL program that includes the embedded SQL statement is being pre-compiled using the precob command of the Oracle Pro*COBOL., 2. And, the COBOL program is being created from the Pro*COBOL program of 1 above using the INSDBINF command,. 3. And, the COBOL program created in 2 above is compiled by specifying compile option "NUMBER". Or [condition 2] 1. The COBOL program (including the embedded SQL) is being pre-compiled after option "-g" is specified in the sqlpcob command of Symfoware.

NO.	V/L(*)	P number	Content of change
			2. And, the COBOL program created in 1 above is compiled by specifying compile option "NUMBER".
3	V10.1.0 to V10.4.0	PG89665	When all of the following conditions are met, error messages are output during the compilation and the COBOL program cannot be compiled. JMN2206I-S KEY IS clause's data name 'xxxxx' cannot be an address variable item. 1. The KEY IS specification of the OCCURS clause is being described. 2. And, the data name specified in 1 above is at a variable position in the records.
4	V10.1.0 to V10.4.0	PG97116	When all of the following conditions are met during the compilation of the COBOL program, the compiler terminates abnormally and outputs the following message. The compilation process cannot be continued. If other diagnostic messages have been generated, correct those errors and try the compilation again. (substep-name=JMNRAL, module-name=SABEMAC, detailcode=6202, line-number=nnn). 1. Compilation option BINARY(BYTE) is specified. 2. And, MAX function or MIN function with numeric as argument is described. 3. And, the argument of function of 2 above is 3 or less. 4. And, the binary item is being specified in the argument of 2 above. 5. And, the area length of the binary item of 4 above, is 1, 3, 5, 6 or 7 bytes (*1). 6. And, the number of digits (*2) of the temporary data item string in the function value of function of 2 above is greater than 19 and less than 30. 7. And, the constant is not specified in the argument of 2 above. 8. And, in the argument of 2 above, the floating-point data item is not being specified. 9. And, in the argument of 2 above, the item subscripting ALL is not specified. *1: The following binary items apply. <pre> Allocated area length ---------------------- digit count Signed Unsigned -------------------------------------- 1 - 2 1 - 2 1 5 - 6 5 - 7 3 10 - 11 10 - 12 5 12 - 14 13 - 14 6 15 - 16 15 - 16 7</pre> *2: The temporary data item stored in the function value becomes the following digit count. - Integer part line = Maximum value in the integer part of the entire argument - Decimal part digit count = Maximum value in the decimal part of the entire argument
5	V10.1.0 to V10.4.0	PG98620	Under the following conditions when compiling, the COBOL program terminates abnormally without giving an error message. 1. Compiler option OPTIMIZE is specified (*1). 2. And, the CALL statement in which the called program is specified in the identifier is described. 3. And, the USING phrase in the CALL statement of 2. 4. And, the RETURNING phrase in the CALL statement of 2.

NO.	V/L(*)	P number	Content of change
			5. And, pointer data item is set or referred to.
			6. And, an internal program exists.
			7. And, when the USING phrase is omitted in the header of the Procedure Division of 6.
			*1: The compiler option default value differs based on the OS on which product is installed.
			- For Solaris, Linux(x86/x64/IPF) default is OPTIMIZE.
			- For Windows(x86/x64/IPF) default is NOOPTIMIZE.
6	V10.1.0 to V10.4.0	PG98742	When a COBOL program is executed, it might not behave correctly. Or while compiling the COBOL program, the compiler might terminate abnormally. The following conditions help identify when that situation might occur.
			[Condition 1]
			When a data item is declared either in WORKING-STORAGE SECTION, LOCAL-STORAGE SECTION (*1) or CONSTANT SECTION.
			1. When a group item of level-number 01 that has subordinate data item specifying the TYPE clause is defined.
			2. And, in the group item of 1 REDEFINES clause is being specified.
			3. And, when the length of the data item in 1 is smaller than the length of the data item being redefined in 2.
			Or
			[Condition 2]
			When data item is declared in the FILE SECTION.
			1. When record description which includes the TYPE clause is being defined.
			2. And, the file description entry that contains the record description entry of 4 contains two or more data description entries of level-number 01.(*2)
			3. And, when the length of record description items in 4 is smaller than the length of record description items in 5.
			*1: LOCAL-STORAGE SECTION functionality is supported in x64 V10.4 and later.
			*2: The record description item's area is redefined implicitly when more than 2 record description items are mentioned in the same file description items in the file section.
			Example 1)
			When a data item is declared either in WORKING-STORAGE SECTION, LOCAL-STORAGE SECTION or CONSTANT SECTION.

```
      01 D1 PIC X(20).    *> SIZE 20 BYTE   D1 is larger than D2.
      01 D2 REDEFINES D1. *> SIZE  8 BYTE
        02 D21 TYPE T1.   *> SIZE  8 BYTE
      01 T1 TYPEDEF.
        02 TD1 PIC X(8).  *> SIZE  8 BYTE
```

Example 2)

When data items are declared in the FILE SECTION.

```
  FD FILE1.
    01 R1 PIC X(20). *> SIZE 20 BYTE     R1 is larger than R2
    01 R2 TYPE T2.    *> SIZE  8 BYTE    It is implicitly redefine
                  *> and allocated in the same region as R1 and R2
```

NO.	V/L(*)	P number	Content of change
			``` 01 T2 TYPEDEF. 02 TD2 PIC X(8). *> SIZE 8 BYTE ```
7	V10.1.0 to V10.4.0	PG99190	When compiling a program which meets the following conditions, the compiler terminates abnormally after generating the following message (*1). (*1) JMN0124I-U This source file has an invalid character. The compilation is terminated. (substep-name=JMNALO, line-number=0.) 1. Compile option RCS(UTF16), ENCODE(UTF8,UTF16) or ENCODE(UTF8,UTF32) is specified. 2. And, compilation option CHECK(PRM),CHECK(ALL) or SAI is specified. 3. And, the encoding of the COBOL program is Shift_JIS. 4. And, a nonnumeric literal(except National character constant)is specified in the BY CONTENT phrase of CALL Statement or INVOKE Statement. 5. And, a national character is included in the nonnumeric literal of 4.
8	V10.1.0 to V10.4.0	PH01579	Under the following conditions, the COBOL compiler terminates normally without generating an error message. 1. A paragraph name or section name is specified in the USING phrase of CALL statement.
9	V10.1.0 to V10.4.0	PH01893	Under the following conditions, when the cobdepend command is executed, the compile terminates abnormally. 1. The file path length exceeds 4096 bytes. And, [condition 1] 1. The directory is specified with the compiler option "-do" with the environment variables "COBOLOPTS" or cobdepend command argument. 2. The following conditions are met for the length of character string. - The specified directory name ends with a slash (/), and the character string length is 4100 bytes. 　"Directory name" + "File name" (except for the COBOL source file extension - The specified directory name does not end with a slash (/), and the character string length is either 4099 or 4100 bytes. 　"Directory name" + "/" + "File name" (except for the COBOL source file extension Or, [condition 2] 1. The directory is specified with either of the following compiler options with the environment variables "COBOLOPTS" or cobdepend command argument. -R -dr 2. Either the class name paragraph or repository paragraph is described in the COBOL source file. 3. The length of either of the following character strings exceeds 4092 bytes. 　"Directory name (*1)" + "Class name" of the (CLASS-ID) paragraph 　"Directory name (*1)" + "Class identifier" in the repository paragraph

NO.	V/L(*)	P number	Content of change
			Or,
			[condition 3]
			1. The directory is specified with the compiler option "-m" with the environment variables "COBOLOPTS" or cobdepend command argument.
			2. IN/OF XMDLIB is specified in a library at compile time in the COPY statement.
			3. The length of the following character string exceeds 4096 bytes.
			"Directory name (*1)" + "Text-name of COPY statement" + "Extension of form descriptor (*2)"
			Or,
			[condition 4]
			1. The directory is specified with the compiler option "-f" with the environment variables "COBOLOPTS" or cobdepend command argument.
			2. IN/OF XFDLIB is specified in a library at compile time in COPY statement.
			3. The length of the following character string exceeds 4096 bytes.
			"Directory name (*1)" + "Text-name of COPY statement" + "Extension of file descriptor (*3)"
			Or,
			[condition 5]
			1. The directory is specified with the compiler option "-I" with the environment variables "COBOLOPTS" or cobdepend command argument.
			2. Either text-name or text-name literal is specified in the COPY statement.
			3. The length of the following character string exceeds 4096 bytes.
			"Directory name (*1)" + "Text-name of COPY statement" + "Extension of library (*4)"
			"Directory name (*1)" + "Text-name literal of COPY statement"
			Or,
			[condition 6]
			1. "-Xnocall" is not specified for cobdepend command argument.
			2. The following files do not exist.
			3. Program name specified by CALL statement + either of the following extensions
			- .cbl
			- .cob
			4. The directory is specified with the compiler option "-do" with the environment variables "COBOLOPTS" or cobdepend command argument.
			5. The length of the following character string exceeds 4094 bytes.
			"Directory name (*1)" + "Program name specified by CALL statement"
			*1 If the last character of specified directory name in the option is not a slash (/), a slash (/) is added to the directory name.
			*2 If other than "None" is specified in the environment variables "SMED_SUFFIX", this setting value is used.
			Otherwise, the following extensions are used.

NO.	V/L(*)	P number	Content of change
			- .pmd
			- .pxd
			- .smd
			*3 If other than "None" is specified in the environment variables "FFD_SUFFIX", this setting value is used.
			Otherwise, the following extension is used.
			- .ffd
			*4 If other than "None" is specified in the environment variables "COB_LIBSUFFIX", this setting value is used.
			Otherwise, the following extensions are used.
			- .cbl
			- .cob
			- .cobol
10	V10.1.0 to V10.4.0	PH02032	Under the following conditions, while compiling the COBOL program, at times regarding the continuation line of the literal the compiler generates the following message.
			JMN1022I-S The ending quotation mark of the literal is missing. The characters at the end of Area B are assumed to be a literal.
			JMN1021I-S The continuation line of the literal must have a hyphen in the indicator area and must start with a quotation mark. The continuation of the literal is assumed.
			1. REPLACE statement is used.
			2. When there is the following description in the range(*1) where replacement is done by REPLACE statement of 1).
			a) COPY statement is used.
			b) The character(*2) ahead of the COPY statement of (a) is not one of the following characters.
			"(","")",";" and "."
			c) Continuation of lines of the nonnumeric literal(*3) is described after the COPY statement of (a).
			d) Continuation of lines of the COBOL words(*4) is not described between the COPY statement of (a) and the continuation lines of the nonnumeric literal of (c).
			*1: Until the next REPLACE statement appears or the end of the compilation unit
			*2: The space and the comment line are excluded
			*3: Nonnumeric literal, hexadecimal nonnumeric literal, national nonnumeric literal or boolean literal
			*4: PICTURE character-string or COBOL words excluding national language user-defined word
11	V10.1.0 to V10.4.0	PH02168	Under the following conditions, while compiling the COBOL program the following error message is generated.
			JMN1479I-S The concatenation expression is too long. A literal is created using the first parts of the expression so that it does not exceed the maximum length.

NO.	V/L(*)	P number	Content of change
| | | | ```
JMN0124I-U This source file has an invalid character. (substep-
name=JMNxxx, line-number=nn.) The compilation is terminated.
``` |
			1. Encoding of COBOL source is UTF-8. And,
			2. Concatenation expression is described for National character constants (*1)
			3. When the number of characters concatenated at 2 are 54-80 characters.
			(*1) Including named constants and symbolic constant.
12	V10.1.0 to V10.4.0	PH02265	Under the following conditions, a compilation error may not be generated.
			1. Concatenation expression is described in the external name of program, class or method.
			2. And, one of the following is described at the beginning of the concatenation expression.
			- Character constant
			- National character constant
			3. And, when one of the following is described in the concatenation expression.
			- Hexadecimal character constant
			- National hexadecimal character constant
13	V10.1.0	PG73504	Under the following conditions, COBOL compiler terminates abnormally and generates the following message.
| | | | ```
JMN1059I-U There is insufficient memory for processing the library.
``` |
			1. Pro*COBOL is used.
			2. Pro*COBOL program contains the following descriptions.
			a. The EXEC SQL INCLUDE sentence is specified. (The file specified for the EXEC SQL INCLUDE sentence is called INCLUDE file.)
			b. The next line of the EXEC SQL INCLUDE sentence is a comment line. (The relative line number of a comment line is [B].)
			c. The next line of comment line is not a comment line.
			d. The INCLUDE file includes the "END-EXEC". (The relative line number of the last "END-EXEC" is [D].)
			e. [B] is equal to [D].
			3. The COBOL program is created from Pro * COBOL program by using the insdbinf command.
			4. The COBOL program is compiled by COBOL compiler.
| | | | ```
Example
Pro*COBOL
(*1):
 098:000098   EXEC SQL
 099:000099        INCLUDE INC01   *> conditions 2)-a
 100:000100   END-EXEC.
 101:000101*  comment line        *> conditions 2)-b([B]) 2)-e
 102:000102   EXEC SQL             *> conditions 2)-c

INCLUDE file (INC01.cob)
(*1):
 099:000099   EXEC SQL
``` |

NO.	V/L(*)	P number	Content of change
			``` 100:000100 END DECLARE SECTION 101:000101 END-EXEC. *> conditions 2)-d([D]) 2)-e *1:relative line number: ```
14	V10.1.0	PG73470	Under the following conditions, the message (JMN9333I-W (STD) Number of USING phrase in CALL statement must not exceed 5) is generated at compile time. 1. The compile option FLAGSW(STDM) or FLAGSW(STDI) is valid. 2. There is a USING phrase in CALL statement. 3. The number of parameters specified for USING are 5 or less, and the parameters are identifiers. 4. One or more reference modification or subscript is specified for identifier in 3. 5. The identifier is specified to reference modification or subscript in 4. 6. There is a total of 6 or more identifiers in 3 and 5.
15	V10.1.0	PG77217	Debugging cannot begin because interactive debugger terminates abnormally when the execute form file made on the following conditions is debugged using the interactive debugger. 1. Library is included in a program using COPY statement. 2. The included COPY library has only a PROCEDURE DIVISION header. The program is compiled specifying the compiler option TEST.
16	V10.1.0	PG74245	Under the following conditions, the compiler terminates abnormally. 1. PROCEDURE division has 10,000-line (*) or more. 2. Object identifier is used. *: 10,000 lines or more is standard. It is also possible to see this even when there are fewer than 10,000 lines. However, that possibility is extremely low for a standard program of 10,000 lines or more that does not write two or more sentences to one line.

17	V10.1.0	PG77383	Under the following conditions, the execution results are incorrect.
			
			1. The compile option OPTIMIZE is valid. (*1).
			
			2. There is a statement that sets packed decimal item to numeric edited data item.
			
			3. The number of digits for integer part of packed decimal item and numeric edited data item are the same and neither have a decimal-part.
			
			4. The numeric edited data item specifies only zero suppression for the edit (Only '9', 'Z', and '*' are used for PICTURE character-string).
			
			5. Before the statement in 2 (*2), there is a statement that sets data item (or intermediate result) to packed decimal.
			
			6. The relation between number of digits for the data item in 5 (or intermediate result) and number of digits for the packed decimal part in 2 is the following :
			
			``
			--- 	
			Data item (or intermediate result) Packed decimal item
			--- 	
			            2 3
			            4 5
			            6 7
			            8 9
			           10 11
			           12 13
			           14 15
			``

NO.	V/L(*)	P number	Content of change
| | | | ```
                    16                                               17
-------------------------------------------------------
``` |
| | | | 7. The data item in 5 (or intermediate result) and the packed decimal item in 2 do not have the decimal-part.

*1: NOOPTIMIZE is the default.

*2: MOVE statement, COMPUTE statement, and so on. |
| 18 | V10.1.0 | PG75097 | Under the following conditions, the COBOL application running under Red Hat Enterprise Linux (for Intel64) cannot be remotely debugged from the NetCOBOL Studio for Windows edition. "Error occurred in debugger or engine. (65535)" is displayed.

1. The server is Red Hat Enterprise Linux (for Intel64) or Red Hat Enterprise Linux (for Itanium).

2. Installed the following products in the server.

   - Alchemy NetCOBOL Enterprise Edition V10.1.0 for Linux (64bit)

3. Gdb used on the server is gdb-7.0(*) or later.

*: Red Hat Enterprise Linux 5.5 is installed or gdb-7.0 is installed alone. |
| 19 | V10.1.0 | PG77463 | Under the following conditions, COBOL compiler terminates abnormally and generates the following message.<br><br>```
PROGRAM-NAME xxx 0: JMN0102I-U  The compile process could not
continue. If other diagnostic messages have been generated,
correct those errors and try the compilation again. (substep-
name=JMNRAL, module-name=SABEMAC, detail-code=6202, line-
number=yy)
```<br><br>1. The presentation file module is used.<br><br>2. Special register (*) in presentation file is specified for the relational condition operand.<br><br>3. The special register in 2 is modified by national item.<br><br>* : Special register of presentation file<br><br>  - EDIT-MODE<br><br>  - EDIT-OPTION<br><br>  - EDIT-OPTION2<br><br>  - EDIT-OPTION3<br><br>  - EDIT-COLOR<br><br>  - EDIT-STATUS<br><br>  - EDIT-CURSOR |
| 20 | V10.1.0 | PG78440 | Under the following conditions, the execution results are incorrect.

1. The BINARY(BYTE) or BINARY(WORD, MLBOFF) compiler option is specified, along with

2. The following intrinsic functions:

[A]

  - FUNCTION MAX

  - FUNCTION MIN

  - FUNCTION MEAN |

NO.	V/L(*)	P number	Content of change

			- FUNCTION MEDIAN
			
			   - FUNCTION RANGE
			
			  [B]
			
			   - FUNCTION ANNUITY
			
			   - FUNCTION NUMVAL
			
			   - FUNCTION NUMVAL-C
			
			   - FUNCTION RANDOM
			
			3. The arguments of the function in 2) above are fixed point of 9 digit or less.
			
			4. The number of the arguments in [A] above is 4 or more.

21	V10.1.0	PG78833	Under the following conditions, COBOL compiler terminates abnormally and generates the following message. ``` PROGRAM-NAME xxx 0: JMN0102I-U The compile process could not continue. If other diagnostic messages have been generated, correct those errors and try the compilation again. (substep- name=JMNRAL, module-name=SABEMAC, detail-code=6202, line- number=yy) ``` 1. The compiler option BINARY(BYTE) is valid. 2. The compiler option CHECK(NUMERIC) or CHECK(ALL) is valid. 3. The statement with subscript or reference modification in PROCEDURE DIVISION. 4. The binary data item of USAGE COMP-5 is specified for either subscript, high-order end character position of a reference modification or the length of reference modification. 5. The number of digits of binary data item in 4 is 10 to 16 or 1. Compiler option BINARY(BYTE) is valid. 2. There is CALL statement with the USING BY VALUE specification that calls the subprogram. 3. Data item of CALL statement in 1 is a COMP-5binary item. 4. The number of digits of binary data item in 3 is 10 to 16.
22	V10.1.0	PG79472	Under the following conditions, the program terminates abnormally with producing the message. ``` PROGRAM-NAME xxx 0: JMN0102I-U The compile could not continue. If other diagnostic messages have been generated, correct those errors and try the compile again. (substep-name=JMNRAL, module- name=SABEMAC, detail-code=5460, line-number=yy) ``` 1. There is binary arithmetic operator (addition or subtraction). 2. Identifier of the operand in 1 is Signed packed-decimal item or Signed zoned decimal item. 3. Number of digits is the same as identifier of the operand in 1. 4. The identifier for the result is same identifier as the operand in side 1. ``` Example) 01 A PIC S9(4) COMP-3. ```

NO.	V/L(*)	P number	Content of change

			<pre>01 B PIC S9(4) COMP-3.
			01 C PIC S9(4) DISPLAY.
			01 D PIC S9(4) DISPLAY.
			        :
			     COMPUTE A = A + B.
			     SUBSTRUCT C FORM D.</pre>

23	V10.1.0	PG79150	Under the following conditions, the program terminates abnormally without generating a message.
			
			  1. The compiler option OPTIMIZE is valid.
			
			  2. Following identifier is used in PROCEDURE DIVISION. (*)
			
			    - Data item with subscripting
			
			    - Data item with reference modification
			
			    - variable occurrence data item (DEPENDING ON phrase)
			
			    - Data item included in same group item as variable occurrence data item
			
			* : It will likely not terminate even under this condition because the error occurs depending on the state of memory at compile time.

24	V10.1.0	PG81352	Under the following conditions, NetCOBOL Studio or Interstage Studio for Windows edition cannot be connected with the Linux(64) server. "Connecting to server did not succeed." is displayed.
			
			  1. IPv6 is invalidated on the Linux(64) system.
			
			  2. The enable-rds.sh is executed on the Linux(64) system, and NetCOBOL remote development service starts.
			
			  3. NetCOBOL Studio or Interstage Studio for Windows edition connects the Linux(64) system .

25	V10.1.0	PG85632	Under the following conditions, COBOL compiler terminates abnormally or the line-numbers generated by the compiler are incorrect.
			
			  1. Pro*COBOL is used.
			
			  2. The insdbinf command is used.
			
			  3. The EXEC SQL INCLUDE sentence is specified in the COBOL program.
			
			  4. The line-number is set to sequence number area of the INCLUDE file.

26	V10.1.0	PG85966	Under the following conditions, a wrong debugging information file is created, and the COBOL program cannot be debugged.
			
			  1. The compiler option TEST is valid.
			
			  2. The text-name literal is specified for the COPY statement.
			
			  3. The library (*1) taken by the COPY statement of 2 contains the COPY statement for which text name is specified.
			
			  4. The compiler terminates abnormally.(*2)
			
			*1: When the library is nested, any one of the nested libraries.
			
			*2: COBOL compiler returns a code other than 0-3 without giving a compiler message.

27	V10.1.0	PG86728	Under the following conditions, the compiler generates an erroneous compile message.
			
			<pre>The interface for method 'mmmmm' must conform to that of the
			overridden method. Method 'mmmmm' is ignored.</pre>

NO.	V/L(*)	P number	Content of change

			1. The OVERRIDE clause is specified in the Method definition.
			
			2. The method in (1) and the method of the same name are in the parent class. When the USING or RETURNING parameter specified for each method encounters the following requirements.
			
			    a. The parameter of one method is a group item and the parameter of the other method is group item or an alphanumeric item.
			
			    b. The size of the parameter is the same.

28	V10.1.0	PG75097	Under the following conditions, the COBOL application running under Red Hat Enterprise Linux (for Intel64) cannot be remotely debugged from the NetCOBOL Studio for Windows edition. "Error occurred in debugger or engine. (65535)" is displayed. 1. The server is Red Hat Enterprise Linux(for Intel64) or Red Hat Enterprise Linux (for Itanium). 2. Installed the following products in the server. - Alchemy NetCOBOL Enterprise Edition V10.1.0 for Linux (64bit) 3. Gdb used on the server is gdb-7.0(*) or later. *: Red Hat Enterprise Linux 5.5 is installed or gdb-7.0 is installed alone.
29	V10.1.0	PG88107	Under the following conditions, COBOL compiler terminates abnormally and generates the following message. <pre>JMN0102I-U The compile could not continue. If other diagnostic messages have been generated, correct those errors and try the compilation again. (substep-name=JMN300, module-name=SC30MKTT, detailcode=3209, line-number=nnn)</pre> When there is either of the following statements: - Statement that moves a numeric item that doesn't have integer digits to a BINARY-CHAR/SHORT/LONG/DOUBLE data item - Statement that moves an intermediate result that doesn't have integer to a BINARY-CHAR/SHORT/LONG/DOUBLE data item
30	V10.1.0	PG88615	Under the following conditions, the compiler gives an erroneous compile message (JMN1477I-I). 1. The SPECIAL-NAMES paragraph is declared. 2. The severity code S code is found in the special-names paragraph. Note: Even if it satisfies all of the conditions, this event might not occur according to the state of the memory.

31	V10.1.0	PG88850	Under the following conditions, the execution results were incorrect.
			
			1. Compiler option OPTIMIZE is valid.
			
			2. The following data items are declared.
			
			    a. A data item in the following records:
			
			      - A record of file of object definition and factory definition
			
			      - A record is specified in the WORKING-STORAGE section of the object definition and factory definition
			
			    b. The data item is either of the following attributes:
			
			      - Binary item of 18 digits or less

NO.	V/L(*)	P number	Content of change
			- Index data item c. The OCCURS phrase is not specified in the data item. d. The data item is not subordinate to the group item in which the OCCURS phrase is specified. e. The data item is not declared after the item in which the OCCURS DEPENDING ON phrase is specified. 3. In the Procedure Division, other data item which storage is overlapped the data item in 2) is not used. 4. In the Procedure Division, the value is set to the data item in 2).

*: V/L is a scope where the bug exists.

3.2 NetCOBOL Runtime Environment

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

Table 3.2 Program fix list of NetCOBOL Runtime Environment

NO.	V/L(*)	P number	Content of change
1	V10.1.0 to V10.4.0	PG76651	When all of the following conditions are met while executing a COBOL program, a size comparison of national items varying in length (national language edited item, including the intrinsic function) was incorrect. 1. Compile option RCS (UCS2,LE) or RCS (UTF16,LE) is specified. 2. And, there is a size comparison of a National item and a National item or National language character constant. 3. And, at least one side is an item with the ANY LENGTH clause is specified or reference modification. 4. And, the length of the compared targets differs.
2	V10.1.0 to V10.4.0	PG88590	Under the following conditions, the SORT statement may cause an infinite loop. 1. PowerBSORT is not installed. 2. The record length of the SORT file or MERGE file is 21477 to 21484 bytes. 3. The SORT statement or MERGE statement is executed.
3	V10.1.0 to V10.4.0	PG96520	Under the following conditions, the message that should be written to a trace information file is written to a file that is not a trace information file (a socket is included). 1. A TRACE function is used. 2. The same trace information file name is used for two or more processes.
4	V10.1.0	PG72597	Under the following conditions, the WRITE AFTER ADVANCING PAGE is not valid at execution time. 1. The line sequential file is used. 2. The line sequential file is linked with the external file handler. 3. After an OPEN statement is executed for file in 2, the WRITE statement of the AFTER ADVANCING PAGE phrase is executed.
5	V10.1.0	PG73346	Under the following conditions, While executing the COBOL program, from 16 hexadecimal characters of 8 digits that are embedded in the runtime message JMP0320I-I/U, in the low end 4 digits zero column is set.

NO.	V/L(*)	P number	Content of change
			1. File operation using COBOL file system(*1) is performed.
			2. Either of the following I/O statements is executed.
			- READ statement
			- WRITE statement
			- REWRITE statement
			- DELETE statement
			- START statement
			3. Execution of the I/O statement failed.
			*: Specifying high speed (BSAM) file processing will not be effective.
6	V10.1.0	PG75097	Under the following conditions, the COBOL application running under Red Hat Enterprise Linux (for Intel64) cannot be remotely debugged from the NetCOBOL Studio for Windows edition. "Error occurred in debugger or engine. (65535)" is displayed.
			1. The server is Red Hat Enterprise Linux (for Intel64) or Red Hat Enterprise Linux (for Itanium).
			2. Installed the following products in the server.
			- Alchemy NetCOBOL Enterprise Edition V10.1.0 for Linux (64bit)
			3. Gdb used on the server is gdb-7.0(*) or later.
			*: Red Hat Enterprise Linux 5.5 is installed or gdb-7.0 is installed alone.
7	V10.1.0	PG77612	Under the following conditions, the SORT statement or the MERGE statement fails.
			1. PowerSORT is installed.
			2. COBOL programs are executed in multithread environments.
			3. The SORT statement or the MERGE statement is executed at the same time between different threads.
8	V10.1.0	PG83883	Under the following conditions, the file OPEN fails while executing the COBOL file access routing.
			- Error Number: FA_EFNAME(114)
			- I-O Status : 90
			1. The COBOL file access subroutine reads and writes the index file.
			2. As for API function (*1), the following are specified.
			- The file name is enclosed in double quotes and
			- The NULL pointer is specified for the record key list. (*2)
			3. The API function in 2 is executed.
			*1: cobfa_open
			*2: If the NULL pointer is specified for the record key list to open an indexed file, this function automatically identifies the key list structure, record format, and record length of an existing file, and opens the file.
9	V10.1.0	PG84545	Under the following conditions, COBOL file utility generates the following message and recovery fails.
| | | | ```
System error occurred. code = 0x44
``` |
| | | | 1. cobfrcov command is used. |

NO.	V/L(*)	P number	Content of change
			2. The path name including the comma (,) is specified for the unrecoverable-data-file-name.

10	V10.1.0	PG84546	Under the following conditions, while executing the cobfrcov command the recovery processing fails and one of the following messages is generated.
			
			1. Insufficient memory/disk space to write file.
			
			2. System error occurred. Code = 0x06
			
			3. Recovery file does not exist.
			
			Or, even if recover processing is successful, the following abnormality occurs.
			
			1. File with unintentional/accidental name is generated.
			
			    1. cobfrcov command is used.
			
			    2. And, one of the following is specified.
			
			        2-1. Large file is specified to the index file to be recovered.(*1)
			
			        2-2. Specified Index file to be recovered contains space or comma in file
			
			        2-3. Specified index file which need not to be recovered contains space or comma in data file name

11	V10.1.0	PG87520	Under the following conditions, the MOVE result by internal Bool item is incorrect.
			
			Note: The move result is different according to the state of the area allocated just behind the sending data item.
			
			[Condition 1]
			
			1. Move from internal Boolean item to internal Boolean item.
			
			2. Data start position (*) of the sending data item is different from data start position of the receiving data item.
			
			3. [a] + [b] > 32, or [c] + [d] > 32
			
			    a: Data start position (*) of the sending data item
			
			    b: The length of sending data item or the length of receiving data item, whichever is smaller.
			
			    c: Data start position (*) of the receiving data item
			
			    d: The length of receiving data item
			
			4. The receiving data item is subordinate to the group item.
			
			5. Data start position of the receiving data item is not one byte alignment of the group item in 4.
			
			6. "Bit length of sending data item" < " Bit length from bit data starting location on the receiving side to the first byte boundary "
			
			7. Bit data of sender᾽s data item is outside the byte boundary. Or, Byte boundary exist between the start position of the sender᾽s data item of paragraphs 6(*) (bit length till the byte boundary of the first recipient).
			
			*: relative bit position in byte
			
			Example of Condition 1
			
			<pre> DATA DIVISION.
			        WORKING-STORAGE SECTION.
			        01 DATA1.
			          02 DATA1-1 PIC 1(5) BIT.
			          02 DATA1-2 PIC 1(4) BIT.</pre>

NO.	V/L(*)	P number	Content of change

```
      02 DATA1-3 PIC 1(7) BIT.
01 DATA2.
  02 DATA2-1 PIC 1(1) BIT.
  02 DATA2-2 PIC 1(32) BIT.
01 ANS-DATA  PIC 1(32) BIT VALUE B"1111".
PROCEDURE DIVISION.
    MOVE X"FFFF" TO DATA1.
    MOVE DATA1-2 TO DATA2-2.
    IF DATA2-2 = ANS-DATA
      THEN DISPLAY "OK"
      ELSE DISPLAY "NG"  *> B"11111110-00000000"
    END-IF.
```

[Condition 2]

1. Move from internal Boolean item to internal Boolean item.

2. The sending data item or the receiving data item is referred by the subscript.

3. " The length of receiving data item " > 1

4. It is either the following.

   - "The length of sending data item" > 1

   - The receiving data item is referred by the subscript.

   - " The length of receiving data item " > 25

5. There is not the data starting location of the receiving data item on byte boundary.

6. "Bit length of sending data item" < " Bit length from bit data start position of first byte of the receiving data item

7. The bit data of the sending data item overlaps the byte boundary, or there is byte boundary between from start position of the sending data item to "Bit length from bit data start position of the receiving side to the first byte boundary" in 6.

Example of Condition 2

```
DATA DIVISION.
WORKING-STORAGE SECTION.
01 DATA1.
  02 DATA1-1 PIC 1(5) BIT.
  02 DATA1-2 PIC 1(4) BIT.
  02 DATA1-3 PIC 1(7) BIT.
01 DATA2.
  02 DATA2-1 PIC 1(10) BIT OCCURS 5 TIMES.
01 ANS-DATA  PIC 1(10) BIT VALUE B"1111".
01 CNT       PIC 9 VALUE 2.
PROCEDURE DIVISION.
    MOVE X"FFFF" TO DATA1.
    MOVE DATA1-2 TO DATA2-1(CNT).
    IF DATA2-1(CNT) = ANS-DATA
      THEN DISPLAY "OK"
      ELSE DISPLAY "NG"  *> B"11111110-00000000"
    END-IF.
```

*: V/L is a scope where the bug exists.

3.3 PowerFORM

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

Table 3.3 Program fix list of PowerFORM

NO.	V/L(*)	P number	Content of change
1	V10.1.0 to V10.4.0	PG78189	It may abort logging for the job despite not reaching to the maximum size of the log for form features per job under the following condition: 1. Trace log storage folder (MEFTLOGROOT) is specified. And, 2. The size of at least one of media data, overlay descriptor, form descriptor, or print information file is greater than or equal to 1048576 bytes. And, 3. The file size value makes the following inequation true. "FLOOR((File_Size_In_Bytes MOD 65536) / 1024) * 1024 > Maximum_Size_Of_The_Log_Per_Job[MEFTLOGSIZEP]"

*: V/L is a scope where the bug exists.

3.4 PowerBSORT

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

Table 3.4 Program fix list of PowerBSORT

NO.	V/L(*)	P number	Content of change
1	V6.0.0	PG76059	Under the following conditions, PowerBSORT may cause result error, infinite loop or abnormal termination. 1. The bsort command, the bsortex command or the BSORT function is used. 2. Sort function is specified. 3. The text file CSV format or text file TSV format is specified. 4. The record summation option or the record selection option for the output files is specified. 5. The summation field or selection field is enclosed with a double quotation.
2	V6.0.0	PG76539	Under the following conditions, PowerBSORT may cause result error or abnormal termination. 1. The bsortex command is used. 2. The text file CSV format or text file TSV format is specified. 3. The record reconstruction option for the output files is specified.
3	V6.0.0	PG84172	Under the following conditions, PowerBSORT outputs the error message "A necessary BSORT working area cannot be secured." and terminates abnormally. 1. Sort function is specified. 2. The input file system is a native file system of the system. 3. The input file is a binary file. 4. The input record length is very large. 5. The size of the input file is a small.
4	V6.0.0	PG83119	Under the following conditions, PowerBSORT outputs the error message "The mistake (BSORT_MSGLEVEL) is found in startup file." and terminates abnormally.

NO.	V/L(*)	P number	Content of change
			1. The bsortex command is used.
			2. BSORT_MSGLEVEL is specified in the startup file.
			3. N, E, W or I is specified for BSORT_MSGLEVEL.
5	V6.0.0	PG90774	Under the following conditions, PowerBSORT outputs the error message "First/last option is specified without summation/suppress function." and terminates abnormally. 1. The bsortex command is used. 2. The suppress operand (suppression option) of -summary option is specified. 3. The first operand or last operand of -summary option is specified. 4. The fifo operand of -option option is specified.

*: V/L is a scope where the bug exists.

3.5 J Adapter Class Generator

Information described here is applied to the following products.

- NetCOBOL Enterprise Edition for Linux (64bit)

Program fix list of J Adapter Class Generator

NO.	V/L(*)	P number	Content of change
1	V10.1.0 to V10.4.0	PH02733	Under either of the following conditions, the last character of the class-name or interface-name of COBOL source generated by the J adapter class generator command (java2cob) will be hyphen "-" and the following compiler error occurs. ```\nJMN1123I-S The word "-" is not allowed. This COBOL source disabled\nuntil the next clause, paragraph and section or division.\n``` 1. The 28th character of the class-name except the package-name of Java that generates the adapter class is period (.) or dollar ($). 2. The 28th character of the interface-name except the package-name of Java that generates the adapter class is period (.) or dollar ($).

*: V/L is a scope where the bug exists.

Chapter 4 Notes to consider when a Japanese native application is converted to run globally

This chapter describes what is involved in taking a Japanese native application global.

4.1 Guideline

The program resources and run time code-set are made by Unicode.

📚 See
....
Refer to "Unicode" of "NetCOBOL User's Guide" for details.
....

4.2 Languages

Language used in the messages for compiler, runtime, tools etc., can be selected in accordance with the current locale setting.

Setting of the current locale is valid for the value of the first environment variable detected in the following order.

1. LC_ALL

2. LC_MESSAGES

3. LANG

Messages for the following Japanese locales are provided with this product.

- ja_JP.UTF-8

Messages are generated in English if the specified locale is not supported.

4.3 Feature difference with Japanese version

In NetCOBOL, there are a Japanese version and a Global version. This product is a Global version.

This section explains the feature differences between a Japanese version and a Global version.

4.3.1 Related products

The following NetCOBOL family products are not supported.

- FORM

- FORM Overlay Option

- MeFt/Web

- SIMPLIA

The following Fujitsu products are not supported.

- Interstage Charset Manager

- Interstage List Works

- Interstage Business Application Server

- Interstage JOB Workload Server

- Interstage Big Data Parallel Processing Server

- PrintWalker/LXE

- PrintWalker/LXE-EX

- INTARFRM

🛑 Note

For the following products, the name is different with the Japanese version.

Global version	Japanese version
PowerFORM RTS	MeFt
PowerBSORT	PowerSORT

4.3.2 Specifications

The following specifications are different with the Japanese version.

4.3.2.1 Currency symbol

In this compiler, $ (x'24') is used as currency symbol.

When any other character is used, the following method is used.

- Specify with the compile option CURRENCY.

📑 See

Please refer "A.2.10 CURRENCY (currency symbol handling)" of NetCOBOL User's Guide.

- Specify in the source program by using CURRENCY SIGN clause.

📑 See

Please refer 4.2.3.7 CURRENCY SIGN clause" of "COBOL Language Reference".

🛑 Note

- Character composed of multiple bytes cannot be used as currency symbol. Regarding characters that can be used as currency symbol, please refer "4.2.3.7 CURRENCY SIGN clause" of COBOL Language Reference.

- If data is passed between compilation units having different currency symbol, the result might differ from the intended/expected result. Please specify compile option CURRENCY or CURRENCY SIGN clause and use same currency symbol character.

4.3.2.2 Handling of National item spaces

In this compiler, National item space (Trailing Blank and Figurative Constant SPACE) of Unicode encoding is handled as alphabetic spaces (U+0020).

Specify compilation option NSP, if national spaces are to be changed.

📑 See

Refer to "NSP(Handling of spaces related to national item)" of "NetCOBOL User's Guide" for details.

Note

If data is passed between compilation units with different National item spaces, results might not be as expected. Specify the compilation option NSP, and use the same national item space.

4.3.2.3 Printing

It is recommend that PowerFORM Runtime use PDF file output by using the form descriptor created with PowerFORM when developing both a Japanese version and a global version.

The difference between a Japanese version and a global version is shown as follows.

Please consult "4.3.1 Related products" regarding supported situations for the related products.

Data stream type

Data stream type UVPI is not supported in a global version however it is supported in a Japanese version.

See

Refer to "4.3.1 Related products".

Paper size

The default value of the size of the form is A4 in a Japanese version. LETTER is used in a global version.

When the size of the form is changed, it is specified on the print information file or I control record.

Font for print

The default value of the print font is Minchou/Gothic in a Japanese version. COURIER is used in a global version.

When the print font is changed, the font table is used.

Characters that can be written

It is a range of Unicode and SJIS in a Japanese version. ASCII is used in a global version.

Form descriptor

It corresponds to SMD and PMD in a Japanese version. Only PMD is used in a global version.

See

Refer to "4.3.1 Related products".

4.3.3 PowerFORM RTS

When the Form is output in a Global version, there are the following notes.

- Only ASCII character can be used.

- When the paper size is omitted, it is printed on LETTER size.

- Only the print to the PostScript printer and the PDF output are possible.