# FUJITSU

# FUJITSU Software
# NetCOBOL V12a (V12.2.0A)

# Release Notes

Windows(64)

# Preface

This book explains Release Note of NetCOBOL.

## Abbreviations

The following abbreviations are used in this manual:

| Product Name | Abbreviation |
|---|---|
| Microsoft® Windows Server® 2019 Datacenter<br>Microsoft® Windows Server® 2019 Standard<br>Microsoft® Windows Server® 2019 Essentials | Windows Server 2019 |
| Microsoft® Windows Server® 2016 Datacenter<br>Microsoft® Windows Server® 2016 Standard<br>Microsoft® Windows Server® 2016 Essentials | Windows Server 2016 |
| Microsoft® Windows Server® 2012 R2 Datacenter<br>Microsoft® Windows Server® 2012 R2 Standard<br>Microsoft® Windows Server® 2012 R2 Essentials<br>Microsoft® Windows Server® 2012 R2 Foundation | Windows Server 2012 R2 |
| Microsoft® Windows Server® 2012 Datacenter<br>Microsoft® Windows Server® 2012 Standard<br>Microsoft® Windows Server® 2012 Essentials<br>Microsoft® Windows Server® 2012 Foundation | Windows Server 2012 |
| Microsoft® Windows Server® 2008 R2 Datacenter<br>Microsoft® Windows Server® 2008 R2 Enterprise<br>Microsoft® Windows Server® 2008 R2 Standard<br>Microsoft® Windows Server® 2008 R2 Foundation | Windows Server 2008 R2 |
| Windows® 10 Education<br>Windows® 10 Home<br>Windows® 10 Pro<br>Windows® 10 Enterprise | Windows 10(x64) |
| Windows® 8.1<br>Windows® 8.1 Pro<br>Windows® 8.1 Enterprise | Windows 8.1(x64) |
| Windows® 7 Home Premium<br>Windows® 7 Professional<br>Windows® 7 Enterprise<br>Windows® 7 Ultimate | Windows 7(x64) |
| Red Hat(R) Enterprise Linux(R) 5(for Intel64)<br>Red Hat(R) Enterprise Linux(R) 6(for Intel64) | Linux(64) |
| Red Hat(R) Enterprise Linux(R) 5(for Intel Itanium) | Linux(Itanium) |
| Red Hat(R) Enterprise Linux(R) 5(for x86) | Linux |
| Java(TM) Platform, Standard Edition Development Kit | JDK |

| Product Name | Abbreviation |
|---|---|
| Java(TM) Platform, Standard Edition Runtime Environment | JRE |
| Oracle Solaris | Solaris |

- In this manual, when all the following products are indicates, it is written as "Windows" or "Windows(x64)".

    - Windows Server 2019

    - Windows Server 2016

    - Windows Server 2012 R2

    - Windows Server 2012

    - Windows Server 2008 R2

    - Windows 10(x64)

    - Windows 8.1(x64)

    - Windows 7(x64)

## Purpose of this book

This book explains the function addition, the trouble correction from the old edition, and information on interchangeability according to them.

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

## Object reader in this book

An old product is introduced, and it is targeted for the shift to this product to be examined, and to be shifting.

## Location of this book

Please refer to the manual of each program for a whole image or detailed information because it is fragmentary information though this book explains the function addition, the trouble correction, and information on interchangeability according to them.

## 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 either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

- Oracle and Java are registered trademarks of Oracle and/or its affiliates.

- Linux(R) is the registered trademark of Linus Torvalds in the U.S. and other countries.

- Red Hat and Red Hat Enterprise Linux are trademarks of Red Hat, Inc. in the U.S. and other countries.

- Intel and Itanium are trademarks of Intel Corporation or its subsidiaries in the U.S. and/or other countries.

- MICRO FOCUS and Micro Focus COBOL are trademarks or registered trademarks of Micro Focus (IP) Limited or its subsidiaries in the United Kingdom, United States and other countries.

- All other trademarks are the property of their respective owners.

## 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.

June 2022

# 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 | V12.2.0 | OS support | Windows Server 2019 is supported. | - |
| 2 | V11.1.0 | OS support | Windows Server 2016 is supported. | - |
| 3 | V11.0.1 | OS support | Windows 10 is supported. | - |
| 4 | V11.0.0 | OS support | Windows 8.1 and Windows Server 2012 R2 are supported. | - |
| 5 | V11.0.0 | Fujitsu Common Tools | Fujitsu Middleware Installation System and FJQSS (Information Collection Tool) | - Overview Software Release Guide<br>- FJQSS User's Guide |

## Note

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

## 1.2 NetCOBOL

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

- NetCOBOL Enterprise Edition Server Runtime (64bit) for Windows

Table 1.2 Outline of additional functionality in NetCOBOL V12

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 1 | V12.2.0 | Pre-compilation source convert function | The pre-compilation source convert function converts the source program written in the third-party COBOL syntax, into a source program which can be compiled by NetCOBOL.<br><br>Using this function, it can automatically convert source programs and can perform migration efficiently. | NetCOBOL User's Guide (Third-party COBOL Resource Migration)<br><br>NetCOBOL User's Guide<br>- 3.5.2.1 -CV(Specify use of the pre-compiler source conversion function)<br>- A.2.18 FLAG(diagnostic message level)<br>- A.2.41 PRECONV(Determines whether to use the pre-compiler source conversion function)<br>- I.5 COBPRECONV Command |

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 2 | V12.2.0 | Micro Focus COBOL File Migration Tool | Micro Focus COBOL file migration tool converts the COBOL file created by an application written in Micro Focus COBOL syntax into a format accessible by NetCOBOL.<br><br>Using this tool, you can automatically convert COBOL files and can perform migration efficiently. | NetCOBOL User's Guide (Third-party COBOL Resource Migration)<br><br>- Chapter 3 Migration COBOL File |
| 3 | V12.2.0 | Program Repair Support Function | In the program repair, you can identify the source file used to create an executable file when multiple source files may have been used during development.<br><br>In build, you can skip the option to re-compile programs that are not affected by the change. | NetCOBOL User's Guide<br><br>- Chapter 23 Program Repair Support Function<br><br>- A.2.39 OBJECT(whether an object program should be output)<br><br>- I.6 COBCMPO Command |
| 4 | V12.2.0 | Dump function | Support for dump information written from the executable file and object program. This allows you to check the following information:<br><br>- Compiler options specified when creating an object program.<br>- Object programs constituting an executable file.<br>- Recommended compiler options to be unified between programs. | NetCOBOL User's Guide<br><br>- I.7 COBDUMP Command |
| 5 | V12.2.0 | Data item boundary specification support | Compiler can align the data item with SYNCHRONIZED clause specified on NetCOBOL native boundary or system boundary. | NetCOBOL User's Guide<br><br>- A.2.59 SYNC compiler option(Handling of data item boundary with SYNCHRONIZED clause specified) |
| 6 | V12.2.0 | Cursor operation support at transaction completion | In database (SQL) function, the cursor operation can be specified at transaction completion. | NetCOBOL User's Guide<br><br>- 17.2.8.1.2 Creating an ODBC Information File |
| 7 | V12.2.0 | Pipe support for standard input | Support for reading data from pipe as standard input. | NetCOBOL User's Guide<br><br>- 10.1.4.2 Command Prompt Windows |
| 8 | V12.0.0 | COMP-6 support | The data of packed decimal data item without sign half-byte of the Micro Focus native can be used by describing USAGE IS COMPUTATIONAL-6. | Language Reference<br><br>- 10.7 Packed decimal data item without sign half-byte |
| 9 | V12.0.0 | Fujitsu mainframe floating-point support | The data of Fujitsu mainframe format floating-point format can be operated by using the Fujitsu mainframe floating-point arithmetic emulator. | NetCOBOL User's Guide<br><br>- A.2.20 FLOAT (Internal format of internal floating-point item)<br><br>- A.2.21 FLOATCHK (Mixed check of internal format of |

| NO. | V/L | Function Name | Content | Location in Manual |
|-----|-----|---------------|---------|--------------------|
| | | | | internal-floating point items at execution time)<br><br>- Appendix M Format of floating-point |
| 10 | V12.0.0 | LE subroutine support | The LE subroutines of IBM can be used. | LE Subroutines User's Guide |
| 11 | V12.0.0 | NetCOBOL Studio | Eclipse 4.6 workbench can be used in NetCOBOL Studio.<br><br>Under this environment, the following operability is improved.<br><br>- Split editor view<br><br>- Enlarging or reducing font size by shortcut key<br><br>- The icon of the tool bar is enlarged even on high resolution screen. | NetCOBOL Studio User' s Guide |
| 12 | V12.0.0 | NetCOBOL Studio | Support the break using the conditional expression at debugging time. This enables efficient debugging. | NetCOBOL Studio User's Guide<br><br>- Chapter 7 Debugging Function |
| 13 | V12.0.0 | Enhancement of development environment | The following functions were enhanced.<br><br>- Extensions other than .cob, .cbl, and .cobol can be treated as extensions of COBOL source files.<br><br>- For remote development, makefile creation and remote debugging, the following files are possible.<br><br>  - The COBOL libraries with extensions other than .cbl.<br><br>  - The descriptor files with extensions other than .smd and .pmd.<br><br>- NetCOBOL Studio can treat files stored in a subfolder under the workspace.<br>For remote development, NetCOBOL can create a subdirectory with the same name as subfolder on the server side and send the following files.<br><br>  - The COBOL source files, the COBOL library files, the descriptor files stored in subfolder under the workspace. | NetCOBOL Studio User's Guide<br><br>- 4.7 File content |
| 14 | V12.0.0 | Switch of compiler diagnostic message format | Compiler diagnostic message format can be switched. The message is displayed in the standard error output. | NetCOBOL User's Guide<br><br>- 1.2.1 Setting Environment Variables |

Table 1.3 Outline of additional functionality in NetCOBOL V11

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 1 | V11.1.0 | Enhanced file-identifier and file-identifier literal | Up to 30 characters can now be specified for file-identifier.<br><br>The file-identifier literal can be treated as an environment variable name. | NetCOBOL User's Guide<br><br>- A.2.17 FILELIT(file-identifier literal handling)<br><br>NetCOBOL Language Reference<br><br>- 4.3.1.3 ASSIGN Clause (Sequential File, Relative File, and Indexed File) |
| 2 | V11.1.0 | Micro Focus synonym support | The Micro Focus native synonyms can be used in NetCOBOL by specifying the compiler option MF. | NetCOBOL User's Guide<br><br>- A.2.30 MF (Micro Focus compatible mode)<br><br>NetCOBOL Language Reference<br><br>- 10.8 Synonym Compatibility Mode |
| 3 | V11.1.0 | PRINTER_n support | In the ASSIGN clause, PRINTER_n phrase is supported. n for PRINTER_n is an integer from 1 through 99. | NetCOBOL Language Reference<br><br>- 4.3.1.3 ASSIGN Clause (Sequential File, Relative File, and Indexed File)<br><br>- 11.8.6.17 The WRITE Statement (Object-Oriented Programming) |
| 4 | V11.1.0 | Enhanced cobmkmf command | The extensions other than default (*.cob, *.cobol) can be used as an extension of the source files that the cobmkmf command handles.<br><br>Moreover, the files in the subdirectory can be included in the build target. | NetCOBOL User's Guide<br><br>- I.3 cobmkmf Command |
| 5 | V11.1.0 | Supports specification of data area allocation method | Specify whether the memory which allocates the program data area is acquired statically or dynamically. | NetCOBOL User's Guide<br><br>- 9.1.3.3 Program data area allocation<br><br>- A.2.11 DATAAREA(Specifies the method of data area allocation) |
| 6 | V11.0.1 | Eclipse 4.3 workbench support | Eclipse 4.3(*) workbench can now be used in NetCOBOL Studio, in which view layout can be changed and recent plug-in can be used.<br><br>*: Using custom installation, NetCOBOL Studio based on Eclipse 3.4 can be installed instead of NetCOBOL Studio based on Eclipse 4.3. However, NetCOBOL Studio based on Eclipse 3.4 does not support operation on Windows 10. If you use NetCOBOL Studio on Windows 10, you use NetCOBOL Studio based on Eclipse 4.3. | NetCOBOL Studio User's Guide |

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 7 | V11.0.1 | Free format COBOL editor support<br><br>(Only NetCOBOL Studio (Eclipse 4.3) ) | Free format style COBOL editor can now be used in NetCOBOL Studio. | NetCOBOL Studio User's Guide<br><br>- 5.10 Reference Formats<br><br>- A.1.45 SRF compile option |
| 8 | V11.0.1 | Security enhancement in remote development<br><br>(Only NetCOBOL Studio (Eclipse 4.3) ) | SSH Port Forwarding enhances the security in remote developing on Linux(x64). | NetCOBOL Studio User's Guide<br><br>- 9.2.1.1 NetCOBOL Remote Development Service |
| 9 | V11.0.0 | UTF-32 support | Encoding form UTF-32 can now be used. | User's Guide<br><br>- 20 Unicode |
| 10 | V11.0.0 | COBOL Resource Project function support | A "COBOL resource project" is used for management of the library file and the descriptor file on the NetCOBOL Studio. | NetCOBOL Studio User's Guide<br><br>- 3.1.2.2 COBOL Resource Project |
| 11 | V11.0.0 | COBOL Solution Project function support | A "COBOL solution project" is used for management of multiple projects on the NetCOBOL Studio. | NetCOBOL Studio User's Guide<br><br>- 3.1.2.3 COBOL Solution Project |
| 12 | V11.0.0 | Project configuration conversion command | Provide the project configuration conversion command as a transfer support tool in order to convert the project in Project Manager to the project in NetCOBOL Studio. | NetCOBOL Studio User's Guide<br><br>- E.3 Transition of Project According to Project Configuration Conversion Command |
| 13 | V11.0.0 | The default value of paper size and printing format specification support | Paper size and printing format can now be specified in print information file. | NetCOBOL User's Guide<br><br>- 7.1.9 Print Information File |

Table 1.4 Outline of additional functionality in NetCOBOL V10

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 1 | V10.4.0 | 31-digit support | Up to 31 digits can now be used for numeric items. | NetCOBOL Language Reference<br><br>- 1.2.3.1 Numeric Literal<br><br>- 1.7 Operation Mode<br><br>- 5.4.8 PICTURE Clause<br><br>- 5.4.16 USAGE Clause<br><br>- 8.2.2 Host Variable Definitions<br><br>- Appendix D Intermediate Results<br><br>NetCOBOL User's Guide<br><br>- 6.9 How to Use Other File Systems<br><br>- A.2.2 ARITHMETIC<br><br>NetCOBOL Messages |
| 2 | V10.4.0 | Recursive calls | Programs can now be called recursively. Data items used by the | NetCOBOL Language Reference |

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| | | | recursive program are defined in the LOCAL-STORAGE SECTION. | - 2.3.8 Recursive Attribute of the Program<br><br>- 3.1.1 Program-Id Paragraph<br><br>- 6.4.7 CALL statement<br><br>NetCOBOL User's Guide<br><br>- 8.1.1 Calling Relationship Forms |
| 3 | V10.4.0 | FETCH FIRST/LAST Statement support | FIRST and LAST can now be specified for embedded SQL FETCH statements. | NetCOBOL Language Reference<br><br>- 8.6.4 FETCH Statement |
| 4 | V10.4.0 | Extension of the file size limit | The maximum file size of a record sequential file, a line sequential file and an index file can now be extended to the system limit. | NetCOBOL User's Guide<br><br>- 6.9 How to Use Other File Systems |
| 5 | V10.2.0 | FETCH NEXT/PRIOR Statement support | NEXT and PRIOR can be specified for the embedded SQL FETCH Statement. | Language Reference<br><br>- 8.6.4 FETCH Statement<br><br>NetCOBOL User's Guide<br><br>- Acquisition of data by FETCH PRIOR statement |
| 6 | V10.2.0 | IBM DB2 SQLCA compatibility | SQLSTATE, SQLCODE, and SQLERRD included in SQLCA can be compiled. | Language Reference<br><br>- 8.2.4 SQLSTATE/ SQLCODE<br><br>- 8.2.6 SQLERRD |
| 7 | V10.2.0 | 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<br><br>- High-Speed File Processing |
| 8 | V10.2.0 | Event Log output | The Windows Event Log can be specified as the output destination for DISPLAY statements. | NetCOBOL User's Guide<br><br>- ACCEPT/DISPLAY Function<br><br>- @CBR_DISPLAY_CONSO LE_EVENTLOG_LEVEL (Specify the Event Type for DISPLAY UPON CONSOLE output to the Event Log)<br><br>- @CBR_DISPLAY_CONSO LE_EVENTLOG_SRCNAM E (Specify the Event Source Name for DISPLAY UPON CONSOLE output to the Event Log)<br><br>- @CBR_DISPLAY_CONSO LE_OUTPUT (Specify the Event Log for DISPLAY UPON CONSOLE output) |

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| | | | | - @CBR_DISPLAY_SYSER R_EVENTLOG_LEVEL (Specify the Event Type for DISPLAY UPON SYSERR output to the Event Log) |
| | | | | - @CBR_DISPLAY_SYSER R_EVENTLOG_SRCNAME (Specify the Event Source Name for DISPLAY UPON SYSERR output to the Event Log) |
| | | | | - @CBR_DISPLAY_SYSER R_OUTPUT (Specify the Event Log for DISPLAY UPON SYSERR output) |
| | | | | - @CBR_DISPLAY_SYSOU T_EVENTLOG_LEVEL (Specify the Event Type for DISPLAY UPON SYSOUT output to the Event Log) |
| | | | | - @CBR_DISPLAY_SYSOU T_EVENTLOG_SRCNAME (Specify the Event Source Name for DISPLAY UPON SYSOUT output to the Event Log) |
| | | | | - @CBR_DISPLAY_SYSOU T_OUTPUT (Specify the Event Log for DISPLAY UPON SYSOUT output) |
| 9 | V10.2.0 | CHECK function suppress option | Runtime options are provided to suppress individual CHECK functions. <br> - nocb <br> - noci <br> - nocl <br> - nocn <br> - nocp | NetCOBOL User's Guide <br> - Format of Runtime Options |
| 10 | V10.2.0 | TRACE function suppress option | Runtime option is provided that suppresses the TRACE function. <br> - nor | NetCOBOL User's Guide <br> - Format of Runtime Options |
| 11 | V10.2.0 | 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 Debugging Guide <br> - Trace Information |
| 12 | V10.2.0 | Simpler identification of application errors in COBOL Error Report | When an application error occurs and COBOL Error Report starts, message JMW0099I-U is output. This makes such application errors easy to identify in the output. | NetCOBOL Messages <br> - COBOL Error Report Messages |

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 13 | V10.2.0 | Suppression of SYSERR of DISPLAY statement | The message output to SYSERR of the DISPLAY statement is suppressed. | NetCOBOL User's Guide<br><br>- @NoMessage(Set to Suppress Runtime Messages) |
| 14 | V10.2.0 | NetCOBOL Studio | NetCOBOL Studio includes development features for COBOL programming based on Eclipse, an open-source GUI development environment. | NetCOBOL Studio User's Guide |
| 15 | V10.2.0 | Remote development functions | COBOL programs for the Solaris, Linux(Itanium), Linux(x64), or Windows(x64) servers can be developed remotely. After performing standalone tests on a local computer, add the settings for remote development to the project for a smooth transition to test builds and links on a server. | NetCOBOL Studio User's Guide<br><br>- Remote Development Function |
| 16 | V10.2.0 | Debug functions | The interactive debugger can help programmers detect logic errors in program processing. It can be used to set breakpoints to halt execution of a program, and to verify execution of a program by confirming data item values. The debugging of multithreaded programs is supported; however, the debugger does not support multiple debug sessions in a single workspace. | NetCOBOL Studio User's Guide<br><br>- Debugging Function |
| 17 | V10.2.0 | Screen Handling Module | Screen handling module is supported. | Language Reference<br><br>- 2.8 Screen Handling Module<br><br>NetCOBOL User's Guide<br><br>- Using Screens |

## Note

The Location in Manual column shows which sections of the documentation are impacted by the change.

# 1.3 J Adapter Class Generator

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

- NetCOBOL Enterprise Edition Server Runtime (64bit) for Windows

Table 1.5 Outline of additional functionality in J Adapter Class Generator V12

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 1 | V12.1.0<br>(V12.2.0) | Explicit specification of JavaVM path | New function to specify JavaVM (jvm.dll) path explicitly. This function can be used in the new Java environment. | J Adapter Class Generator User's Guide |

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| | | | | - 4.1.2 Options<br>-vm JavaVM-path<br><br>- 5.3.1 JVM-INIT Method (factory method) Environment variable COBJNI_JAVA_VM |

Table 1.6 Outline of additional functionality in J Adapter Class Generator V11

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 1 | V11.0.0<br><br>(V11.0.0) | Unicode character for file path character string | The Unicode character can be used for the file path character string specified for a command line argument and an optional file. | - |

Table 1.7 Outline of additional functionality in J Adapter Class Generator V10

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 1 | V10.0.0<br><br>(V10.2.0) | J Adapter Class Generator | J Adapter Class Generator is a tool that generates a COBOL class (adapter class) used to call a Java class. Using the generated adapter class makes Java class libraries available from COBOL. | J Adapter Class Generator User's Guide |

NOTE: The version in parentheses shows the version level in the NetCOBOL series.

# 1.4 PowerFORM

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

- NetCOBOL Enterprise Edition Server Runtime (64bit) for Windows

Table 1.8 Outline of additional functionality in PowerFORM V12

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 1 | V12.2.0 | Enhanced print features | New era (Reiwa) is supported. | - |
| 2 | V12.0.0 | Enhanced print features | The era name of Japanese calendar can be customized. | PowerFORM Runtime Reference<br><br>- C.6.19 GENGO (Japanese era name) |

Table 1.9 Outline of additional functionality in PowerFORM V11

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 1 | V11.1.0 | Enhanced print features | Printer paper can now be specified when the specified paper was not supported with the output printer. | PowerFORM Runtime Reference<br><br>- Appendix C.<br>Printer Information File |
| 2 | V11.0.0 | Unicode(UTF-32) support | UTF-32 data can now be used in COBOL applications. | PowerFORM Runtime Reference<br><br>- How to Use PowerFORM RTS |
| 3 | V11.0.0 | Encoding systems for print information file | Print Information File in UTF-8 with BOM can now be used in user created Unicode COBOL applications. | PowerFORM Runtime Reference<br><br>- How to Use PowerFORM RTS |

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| | | | | - Chapter 5.<br>Printer Information File<br><br>- Appendix A.<br>Troubleshooting Guide<br><br>- Appendix E.<br>Error Messages |
| 4 | V11.0.0 | 31-digit support | Up to 31 digits can now be used for numeric fields in COBOL applications. | PowerFORM Runtime Reference<br><br>- Output field declarations |
| 5 | V11.0.0 | Form exporting | The Form can now be outputted to PDF. And The character (surrogate pair) added by character-code standard "JIS X 0213:2004(JIS2004)" can be output. | PowerFORM Runtime Reference<br><br>- Chapter 4.<br>Extended Functions<br><br>- Appendix C.<br>Printer Information File |
| 6 | V11.0.0 | Enhanced print features | Print options are provided to define the behavior when output printer is omitted and when it fails in opening the specified output printer. | PowerFORM Runtime Reference<br><br>- Appendix C.<br>Printer Information File |
| 7 | V11.0.0 | Enhanced print features | Print options are provided to define the behavior when the specified paper was not supported with the printer device. | PowerFORM Runtime Reference<br><br>- Appendix C.<br>Printer Information File |
| 8 | V11.0.0 | Enhanced print features | Print options are provided to define the behavior when the specified paper feed was not supported with the printer device. | PowerFORM Runtime Reference<br><br>- Appendix C.<br>Printer Information File |
| 9 | V11.0.0 | Enhanced print features | The processing time at expansion/ reduction print and the print preview can be shortened. | PowerFORM Runtime Reference<br><br>- Appendix C.<br>Printer Information File |
| 10 | V11.0.0 | Enhanced print features | The processing time of the Form output using the fixed-width font can be shortened. | PowerFORM Runtime Reference<br><br>- Appendix C.<br>Printer Information File |
| 11 | V11.0.0 | Support locale | The following 4 locales outputs are supported.<br><br>- Japanese<br><br>- English<br><br>- Chinese (simplified Chinese)<br><br>- Portuguese | Release Notes<br><br>- 4.4.4 PowerFORM RTS |
| 12 | V11.0.0 | UTC (Coordinated Universal Time) support | UTC (Coordinated Universal Time) can be specified. | PowerFORM Runtime Reference<br><br>- Appendix C.<br>Printer Information File |
| 13 | V11.0.0 | Enhanced print features | When the user character-code is UNICODE, the table of full-width/half-width character used by outputting the UNICODE character can be specified, and customize at the character-code level can be specified. | PowerFORM Runtime Reference<br><br>- Appendix C.<br>Printer Information File |

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 14 | V11.0.0 | Enhanced print features | The Form Descriptor (free frame) can be used. | PowerFORM Runtime Reference<br><br>- Chapter 4.<br>  Extended Functions |
| 15 | V11.0.0 | Enhanced print features | The new function added in PowerFORM V11 can be used. | PowerFORM Getting Started<br><br>- What's New in PowerFORM V11? |

## 1.5 Fujitsu mainframe floating-point arithmetic emulator

The Fujitsu mainframe floating-point arithmetic emulator is provided from V12 or later.

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

Table 1.10 Outline of additional functionality in Fujitsu mainframe floating-point arithmetic emulator V12

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 1 | V12.0.0 | Fujitsu mainframe format floating-point support | The data of Fujitsu mainframe format floating-point format can be operated by using the Fujitsu mainframe floating-point arithmetic emulator. | NetCOBOL User's Guide<br><br>- A.3.20 FLOAT (Internal format of internal floating-point item)<br><br>- A.3.21 FLOATCHK (Mixed check of internal format of internal-floating point items at execution time)<br><br>- Appendix M Format of floating-point |

## 1.6 PowerBSORT

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

- NetCOBOL Enterprise Edition Server Runtime (64bit) for Windows

Table 1.11 Outline of additional functionality in PowerBSORT V8

| NO. | V/L | Function Name | Content | Location in Manual |
|---|---|---|---|---|
| 1 | V8.0.1<br><br>(V12.2.0) | Data format | The Fujitsu mainframe format floating-point was supported with NetCOBOL. | PowerBSORT User's Guide<br><br>- 1.6.3 Data format<br><br>- 1.6.4 Data forms that can be specified in each field |
| 2 | V8.0.0<br><br>(V12.0.0) | Data format | The COMP-6 format was supported with NetCOBOL. | PowerBSORT User's Guide<br><br>- 1.6.3 Data format<br><br>- 1.6.4 Data forms that can be specified in each field |

NOTE: The version in parentheses shows the version level in the NetCOBOL series.

# 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 applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

### 2.1.1 Using JRE by NetCOBOL Studio

**Content**

- V12a (V12.2.0A) or later

  Before using NetCOBOL Studio, you must obtain and install OpenJDK 8 JRE for Windows(x86) provided by Adoptium.

**Action**

Before using NetCOBOL Studio, obtain and install OpenJDK 8 JRE for Windows(x86) provided by Adoptium.

For details on how to obtain, install, and for notes about the OpenJDK 8 JRE for Windows (x86) provided by Adoptium, refer to "OpenJDK/JRE Used by NetCOBOL Studio" in "NetCOBOL Software Release Guide."

### 2.1.2 Change the file association by using PRINTER_n phrase

**Content**

- V11.0 or earlier

  The ASSIGN clause with PRINTER_n phrase associates file-name with a physical file.

- V11.1 or later

  The ASSIGN clause with PRINTER_n phrase associates file-name with a printing device.

**Action**

Change PRINTER_n to another name.

### 2.1.3 Change default link option /SUBSYSTEM

**Content**

The default of link option /SUBSYSTEM was changed.

**Impact**

When the following functions are used, the output character string might not be displayed in the window with this link option.

- COBOL Console Window

- Screen Function

**Action**

If you use COBOL console window or screen function, please specify following LINK option when you link main programs.

- When compile option MAIN(WINMAIN) is specified for the main program, it is WinMain type.

  When compile option MAIN(MAIN) is specified for the main program, it is main type.

  [WinMain type]

/SUBSYSTEM:WINDOWS and 5.02

[main type]

/SUBSYSTEM:CONSOLE and 5.02

- If you use screen function it is possible to display it correctly by specifying environment variable @ScrnSize and changing size of logical screen.

## 2.1.4  Reference format setting in NetCOBOL Studio

**Content**

In NetCOBOL Studio V11.0.1 or late (the Eclipse 4.x workbench), SRF and TAB compile option setting is not consistent with the applicable editor setting.

**Action**

To restore the consistency to NetCOBOL Studio V11.0.1 or earlier (the Eclipse 3.4 workbench), select "SRF and TAB compile option setting to be consistent with the applicable editor setting" checkbox.

## 2.1.5  Runtime code-set when compilation option ENCODE is specified

**Content**

Added the compilation option ENCODE to define the encoding form of the data item.

With the specification of compilation option ENCODE, the runtime code set is changed.

- When the compilation option RCS is explicitly specified.

    - The runtime code-set becomes the code-set that is specified with the RCS compilation option.

- When the compilation option RCS is not specified explicitly.

    - When the compilation option ENCODE is specified explicitly, then the runtime code-set becomes the Unicode.

    - When the compilation option ENCODE is not specified explicitly, then the runtime code-set becomes the ANSI code page.

**Conditions**

1. In compiler versions prior to 10.5.0, the compilation option RCS (SJIS) is specified explicitly or implicitly and it is a program asset.

2. And, in compiler versions 11.0.0 and forward, without specifying the compilation option RCS (SJIS) explicitly, for the program assets of 1 above, the compilation option ENCODE(SJIS[,SJIS]) is described explicitly and recompiled.

**Impact**

The impact corresponding to the above conditions is as follows.

- An error (JMP0081I-U) occurs in the following situations.

    - When the recompiled program is called from a program that is not recompiled.

    - When a program that is not recompiled is called from a program that is recompiled.

- During execution after the main program is recompiled, the source becomes the Unicode.

For details, refer to "Unicode" in the "NetCOBOL User's Guide".

**Action**

Create the target program without explicitly specifying the compilation option ENCODE.

Moreover, in regards to the repository, keep the compilation option RCS and the compilation option ENCODE specified on the reference and remote side the same.

## 2.1.6 Location of SQLCODE/SQLMSG/SQLERRD definition

SQLCODE, SQLMSG, and SQLERRD can now be defined outside of the SQL declaration section. Under the conditions listed below, SQL statement runtime information was not stored in V10.1.0. In V10.2.0 and later, runtime information is stored in SQLCODE, SQLERRD, and SQLMSG.

1. SQLSTATE is described inside the SQL declaration section, and

2. SQLCODE, SQLERRD, or SQLMSG are described outside the SQL declaration section, and

3. SQL statements are executed, and

4. SQLCODE, SQLERRD, or SQLMSG are referred to in the PROCEDURE DIVISION, and

5. Conditions 1 through 4 are occurred in the same compilation unit.

In V10.2.0 and later, SQLCODE, SQLERRD, and SQLMSG are reserved names used to fetch SQL statement runtime information. If you have variables by those names that are used for any other purpose, the results are unpredictable. In this case, change the names of those variables.

## 2.1.7 Result of addition and subtraction of zoned decimal item outside regulations

Under the following conditions, the execution result of V10.1.0 and that of V10.2.0 and later are different.

1. There is a binary operation of addition or the subtraction.

2. The number of identifier of the result is 1.

3. [NOT] ON SIZE ERROR is not specified.

4. ROUNDED is not specified.

5. Two operands are zoned decimal and numbers of digits are from 16 to 18.

6. The accuracy of intermediate result (*) of binary operation is from 17 to 19 digits.

7. Either of representation of two operands is wrong as zoned decimal.

    *: Refer to "COBOL Language Reference" about the accuracy of intermediate result.

### Example

```
        WORKING-STORAGE SECTION.
        01 DATA-ZONE1    PIC S9(18).
        01 DATA-ZONE2    PIC S9(18).
        01 DATA-2 REDEFINES  DATA-ZONE2 PIC X(18).
        01 DATA-ZONE3    PIC S9(18).
        01 DATA-3 REDEFINES  DATA-ZONE3 PIC X(18).

        PROCEDURE DIVISION.
            MOVE X"FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF" TO DATA-2  DATA-3  .....(a)
            COMPUTE DATA-ZONE1 = DATA-ZONE2 + DATA-ZONE3
            DISPLAY DATA-ZONE1
```

In the above program, save area of alphanumeric item is allocated in zoned decimal. When (a) is executed, wrong representation (0xFFFF...FFFF) is set to DATA-ZONE2 and DATA-ZONE3 as zoned decimal.

- V10.1.0

```
    +333333333333333330
```

- V10.2.0 and later

```
    +545555555455555554
```

The result of the arithmetic statement including the operand to which wrong internal format is set as zoned decimal is not provided for. Please correct the part where wrong internal format is set.

The outline of the representation that zoned decimal is correct is as follows.

| Data item | Zone part | Numeric part | Sign part |
|---|---|---|---|
| Zoned decimal (SEPARATE) | 3 | 0 to 9 | 2B,2D |
| Zoned decimal (no SEPARATE) | 3 | | 4,5 |

## 2.1.8 Interchangeable information regarding bug fixes

Here, it explains bug fixes corrected by NetCOBOL V11 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 |
|---|---|---|---|
| 1 | V10.1.0 to V12.0.0 | PH14815 | Under the following conditions, when a COBOL program is executed, it might not behave correctly.<br><br>1. The compiler option INITVALUE is specified, and a value other than 00 is specified as the value of INITVALUE.<br><br>2. And, a data item is defined without an initial value in the working-storage section of the factory data or object data.<br><br>3. And, reference without setting a value in the data item of 2. |
| 2 | V10.0.1 to V11.0.0 | PH05861 | Under the following conditions, the result of the move is incorrect.<br><br>The reference modification range of the sending side is shifted one digit to the left.<br><br>1. The following MOVE sentences (*1) are described.<br>  - The sending side is a signed zoned decimal item.<br>  - And, the receiving side is a numeric edited data item or a floating point item.<br><br>2. And, the SIGN IS TRAILING SEPARATE CHARACTER clause is specified in the sending data item.<br><br>3. And, there is a reference modification of the sending data item.<br><br>4. And, the length of reference modification of (3) is specified with a numeric literal.<br><br>*1: Includes the implicit MOVE statement. |
| 3 | V10.1.0 to V11.0.0 | PH05538 | Under either of the following conditions, the COBOL program execution result is incorrect or terminates abnormally (ACCESS VIOLATION).<br><br>[Condition 1]<br><br>1. The data item declared to based-storage section is used by either of the following methods.<br>  - There is a subscript.<br>    or<br>  - There is a reference modifier. The high-order-end-character-position of the reference modifier is a variable.<br><br>2. And, compile option OPTIMIZE(*A) is effective<br><br>*A: Default is NOOPTIMIZE.<br><br>[Condition 2] |

| No. | V/L(*) | P number | Content |
|---|---|---|---|
|  |  |  | 1. In the based-storage section, the declared data items are specified in any of the following statements.<br><br>  - INSPECT statement<br><br>  - STRING statement (Nucleus)<br><br>  - UNSTRING statement (Nucleus)<br><br>[Condition 3]<br><br>1. In the based-storage section, the data items specified the OCCURS clause are being declared.<br><br>And, the data items of 1 are being specified in the INITIALIZE statement. |
| 4 | V11.0.0 | PH04570 | Under the following [Conditions], the following [Situation 1] or [Situation 2] might occur.<br><br>[Situation 1]<br><br>The COBOL compiler terminates normally (*A) without giving an error message.<br><br><pre>JMN2038I-S Length of the literal in the VALUE clause must not exceed the length of the item. The extra characters at the right end of the literal are truncated</pre><br>*A: In the created object program, the following literal values are set to the data item of the object program.<br><br>  - The extra characters at the right end of the literal are truncated to match the item length<br><br>[Situation 2]<br><br>The COBOL compiler outputs the following error message to a correct syntax.<br><br><pre>JMN2106I-S The value of the literal following 'THROUGH' in the VALUE clause of the condition-name must be greater than the value of the literal preceding 'THROUGH'.</pre><br>[Conditions]<br><br>1. There is a national item or national edited item that either encodes the following.<br><br>  - UTF32<br><br>  - UTF32LE<br><br>  - UTF32BE<br><br>2. And, the VALUE phrase is specified as follows.<br><br>  - data item of (1), or<br><br>  - Condition-name of which conditional variable is data item of (1)<br><br>3. And, in the VALUE phrase of (2), the national nonnumeric literal (*B) is specified to meet one of the following.<br><br>  - National nonnumeric literal is larger than the sizes specified by the PICTURE phrase<br><br>  - The value of the national nonnumeric literal following 'THROUGH' is greater than the value of the national nonnumeric literal preceding 'THROUGH'<br><br>4. And, national nonnumeric literal of (3) are from 41 to 80 characters.<br><br>*B: The concatenation expression of national nonnumeric literal is contained. |
| 5 | V10.1.0<br>to | PH02265 | Under the following [Conditions 1] or [Condition 2], even if there is an error in the program, the compiler may not output the error messages (*A). |

| No. | V/L(*) | P number | Content |
|---|---|---|---|
| | V11.0.0 | | (*A) |

```
JMN1775I-S A nonnumeric literal or a national literal must be
specified after an AS phrase. The compiler skips to the next
paragraph or division.
JMN1107I-S An invalid character-string is specified in the program-
id, or the program-id is missing. The program-name of the next
paragraph or division is used.
JMN1292I-S An invalid character-string is specified as the program-
name, or the program-name is missing.
JMN5526I-S The method-name specified in the INVOKE statement must be
an identifier, nonnumeric literal, or national character literal.
The INVOKE statement is ignored.
JMN5561I-S The method-name specified in an in-line method invocation
must be a nonnumeric literal or a national character literal. The in-
line method invocation is ignored.
```

The object program that has been created can be executed correctly.

[Conditions 1]

    1. The concatenation expression is described in the following.

      - AS phrase of program-name

      - AS phrase of class-name

      - AS phrase of method-name

      - AS phrase of property-name

      - AS phrase of program-name-literal

      - Method-name specified for INVOKE statement or in-line method invocation

    2. And, the high order end of concatenation expression of (1) is a nonnumeric literal.

    3. And, the concatenation expression of (1) contains a hexadecimal nonnumeric literal.

    4. And, the COBOL source program code set and the runtime code set are the following combinations.

      a. The COBOL source program code set is SJIS and the runtime code set is SJIS.

      b. Or, the COBOL source program code set is UTF-8 and the runtime code set is Unicode.

[Conditions 2]

    1. The concatenation expression is described in the following.

      - AS phrase of program-name

      - AS phrase of class-name

      - AS phrase of method-name

      - AS phrase of property-name

      - Method-name specified for INVOKE statement or in-line method invocation

    2. And, the high order end of concatenation expression of (1) is a national language nonnumeric literal.

    3. And, the concatenation expression of (1) contains a national hexadecimal nonnumeric literal.

| No. | V/L(*) | P number | Content |
|---|---|---|---|
| | | | 4. And, the COBOL source program code set and the runtime code set are the following combinations.<br><br>    a. The COBOL source program code set is SJIS and the runtime code set is SJIS.<br><br>**Supplementary Explanation:**<br><br>A literal that specifies in the following must be a nonnumeric literal or a national language nonnumeric literal.<br><br>- AS phrase of program-name<br><br>- AS phrase of class-name<br><br>- AS phrase of method-name<br><br>- AS phrase of property-name<br><br>- AS phrase of program-name-literal<br><br>- Method-name specified for INVOKE statement or in-line method invocation<br><br>The above conditions are in violation of the syntax rules. |
| 6 | V10.1.0 | PG78440 | Under the following conditions, the execution results are incorrect.<br><br>1. The BINARY(BYTE) or BINARY(WORD, MLBOFF) compiler option is specified, along with<br><br>2. The following intrinsic functions:<br><br>[A]<br><br>  - FUNCTION MAX<br><br>  - FUNCTION MIN<br><br>  - FUNCTION MEAN<br><br>  - FUNCTION MEDIAN<br><br>  - FUNCTION RANGE<br><br>[B]<br><br>  - FUNCTION ANNUITY<br><br>  - FUNCTION NUMVAL<br><br>  - FUNCTION NUMVAL-C<br><br>  - FUNCTION RANDOM<br><br>3. The arguments of the function in 2) above are fixed point of 9 digits or less.<br><br>The number of the arguments in [A] above is 4 or more. |
| 7 | V10.1.0 | PG77383 | Under the following conditions, the execution results are incorrect.<br><br>1. The OPTIMIZE compiler option is specified (*1).<br><br>2. There is a statement that sets packed decimal item to numeric edited data item.<br><br>3. The number of digits for integer part of packed decimal item and numeric edited data item are the same and neither both of them doesn't have decimal-part.<br><br>4. The numeric edited data item specifies only zero suppression for the edit (Only '9', 'Z', and '*' are used for PICTURE character-string).<br><br>5. Before the statement in 2 (*2), there is a statement that sets data item (or intermediate result) to packed decimal item. |

| No. | V/L(*) | P number | Content |
|---|---|---|---|
| | | | 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                16                                    17  --------------------------------------------------- ``` |
| | | | 7. 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. |
| 8 | 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.<br><br>1. Compile option RCS (UCS2,LE) or RCS (UTF16,LE) is specified.<br><br>2. And, there is a size comparison of a National item and a National item or National language character constant.<br><br>3. And, at least one side is an item with the ANY LENGTH clause is specified or reference modification.<br><br>4. And, the length of the compared targets differs. |

*: 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 Developer (64bit) for Windows

- NetCOBOL Enterprise Edition Server Runtime (64bit) for Windows

## 2.2.1 Character-code conversion result of data and imperfect character outside character-code range

**Content**

The result of the character-code conversion of the data stored in the data item of the DISPLAY statement, the STRING statement (writing 2), the UNSTRING statement (writing 2), and the WRITE statement of the print file changed to prevent an unjustified data error.

- Character-code conversion of data beyond the limits of character-code set conversion origin

    - Before V11.0

    It is converted into an alternative character.

    - After V11.0

    It becomes a code conversion error when executing it, and the following messages (detailed code 42 (0x2a)) are output.

- DISPLAY statement: JMP0086I-W

- STRING statement (writing 2) and the UNSTRING statement (writing 2): An overflow condition has occurred. JMP0260I-U when ON OVERFLOW is not specified

- WRITE statement of the print file: JMP0310I-I/U and JMP0320I-I/U (It is executed along with the execution result when the input-output error of the file occurs).

- Character-code conversion of imperfect character

When a surrogate pair of the character-code is a high rank or only one of the subordinate positions is stored or the multi byte character is lacked. The result of converting such an imperfect character is different.

- Before V11.0

The character is converted into the replacement character.

- After V11.0

It becomes a code conversion error during execution, and the following messages (detail code 22 (0x16) or 42 (0x2a)) are output.

- DISPLAY statement: JMP0086I-W

- STRING statement (writing 2) and the UNSTRING statement (writing 2): An overflow condition has occurred. JMP0260I-U when ON OVERFLOW is not specified.

- WRITE statement of the print file: JMP0310I-I/U and JMP0320I-I/U (It is executed along with the execution result when the input-output error of the file occurs).

## Action

Correct to store a correct character-code in the data item that refers to a detailed message code when executing it.

Specify the execute environment variable @CBR_CONVERT_CHARACTER=SYSTEM expressly when you want to obtain the result similar to before V11.0.

However, it is not possible to specify it by the program using the specification after V11.0.

Refer to environment variable information @CBR_CONVERT_CHARACTER on the user's guide for details.

# 2.2.2 Replacement character used by code conversion

## Content

The replacement character used has changed when the character-code conversion corresponding to the character-code in the conversion origin does not exist.

- Before V10.4

Normal-width underscore "_"

- After V11.0

Normal-width question "?"

## Action

Specify the execute environment variable information @CBR_CONVERT_CHARACTER=SYSTEM expressly.

However, it is not possible to specify it by the program using the specification after V11.0.

Refer to environment variable information @CBR_CONVERT_CHARACTER on the user's guide for details.

# 2.2.3 Change in severity code of message when application is executed

## Content

The severity code of the message changed when the following applications were executed.

- Before V11.0

  JMP0086I-E CHARACTER CODE CONVERSION FAILED. $1 $2

- After V11.0

  JMP0086I-W CHARACTER CODE CONVERSION FAILED. $1 $2

**Influence**

The Return code (PROGRAM-STATUS) of the COBOL program changes with a change in the severity code.

**Action**

Refer to the Messages manual for additional details.

# 2.2.4 Interchangeable information regarding bug fixes

Here, it explains bug fixes corrected by NetCOBOL V10 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 |
|-----|--------|----------|---------|
| 1 | V12.0.0 | PH15066 | Under the following conditions, during the execution of a COBOL application, even though the internal floating-point data item of mainframe format is zero, the result converted to the binary item does not become zero. (*1)<br><br>*1: Even if it satisfies all of the conditions, this event might not occur according to the state of the memory.<br><br>  1. The COBOL program is compiled with FLOAT(M) compiler option specified.<br><br>  2. And, the internal floating-point data item is converted to the binary item,(*2)<br><br>  3. And, the length of internal floating-point data item is 4 bytes or less.<br><br>  4. And, the value of internal floating-point data item is zero.<br><br>*2: Conversion is done when showing in the following.<br><br>  - MOVE is done from the internal floating-point data item to the binary data item.<br><br>  - The receiving data item of an operational result that contains the floating-point data item is the binary data item. |
| 2 | V12.0.0 | PH15042 | Under the following conditions, during the execution of a COBOL application, the sorted records may not be correct.<br><br>  1. PowerBSORT is installed.<br><br>  2. And, the COBOL program is compiled with FLOAT(M) compiler option specified.<br><br>  3. And, the internal floating-point data item is specified for sort key.<br><br>  4. And, the SORT statement is executed with sort key of 3. |
| 3 | V10.1.0 to V11.0.0 | PH06622 | Under the following conditions, the record is not printed according to the specified printing attributes or printing position.<br><br>For a Unicode application, additionally, the following phenomena occurs. (*A)<br><br>  - The runtime system message "JMP320I 'CNVER=xx'" is output.<br><br>  - The printing results of the national item are garbled.<br><br>*A: It depends on the data stored in the national item.<br><br>  1. The print file without a FORMAT clause is used. |

| No. | V/L(*) | P number | Content |
|---|---|---|---|
| | | | 2. And, this program is output directly to the printer without using the related product.<br><br>3. And, the record item specified for the WRITE statement or the data item specified for the FROM phrase of WRITE statement is a group item that is subordinate to the data item with the REDEFINES clause.<br><br>4. And, the following either is described in the group item of (3) that specifies a REDEFINES clause.<br><br>   - There is a group item subordinate.<br><br>   - There is a national item subordinate.<br><br>   - An effective CHARACTER TYPE clause is specified for the data item subordinate. (*B)<br><br>   - An effective PRINTING POSITION clause is specified for the data item subordinate. (*B)<br><br>5. And, after the group item of (3), either of the following is described in the same level-number as (3).<br><br>   - There is an elementary item or a group item that specifies a CHARACTER TYPE clause.<br><br>   - There is an elementary item or a group item that specifies a PRINTING POSITION clause.<br><br>   - There is a group item that has the data item subordinate with an effective CHARACTER TYPE clause.<br><br>   - There is a group item that has the data item subordinate with a PRINTING POSITION clause.<br><br>*B: When the REDEFINES clause is specified for either of the following items, during the compilation of a COBOL program, the JMN2224I-W message is outputted. It is a warning that the CHARACTER TYPE clause or the PRINTING POSITION clause specified for the item is invalid.<br><br> - The data item with CHARACTER TYPE clause or the PRINTING POSITION clause specified<br><br> - The group item that has the subordinate data item with effective CHARACTER TYPE clause or PRINTING POSITION clause<br><br> - **Example program showing what occurs:**<br><br>DATA DIVISION.<br>WORKING-STORAGE SECTION.<br>01 DATA1.<br>03 DATA2.<br>05 DATA31.<br>07 DATA31A PIC X(10).<br>05 DATA32 REDEFINES DATA31.<br>07 DATA32A. *> Conditions of (4)<br>09 DATA32A1 PIC X(5).<br>09 DATA32A2 PIC X(5).<br>05 DATA33. *> Conditions of (5)<br>07 DATA33A PIC N(5) MODE-1.<br><br><br>PROCEDURE DIVISION.<br><br>WRITE PRINT-REC FROM <u>DATA1</u> AFTER PAGE. *> Conditions of (3) |

| No. | V/L(*) | P number | Content |
|---|---|---|---|
| 4 | V11.0.0 | PH03200 | Under the following conditions, when the COBOL application is executed, the record is not written as the collating sequence of the index key item.<br><br>Moreover, the intended record might not be read.<br><br>1. When the program is compiled, the compile option ENCODE(SJIS,SJIS) is specified.<br><br>2. And, the item attribute of the prime record key or alternate record key is a national item.<br><br>And, the WRITE statement is executed. |
| 5 | V10.1.0 to V11.0.0 | PH01026 | Under the following conditions, the character comparison cannot be compared correctly.<br><br>1. The COBOL program compiled with the compile option NSPCOMP(ASP) is executed.<br><br>2. And, the encoding of the data item is Shift-JIS for the following.<br><br>- The compile option RCS is omitted.<br><br>- Or, the compile option RCS(SJIS) is specified.<br><br>- Or, the compile option ENCODE(SJIS,SJIS) is specified.<br><br>3. And, either of the following character comparisons is done.<br><br>- National character comparison with a national data item as an operand<br><br>- Character comparison with a group item as an operand<br><br>However, the following conditions are excluded.<br><br>- Comparison between group items that do not include any national data items<br><br>- Comparison between group items including an item whose attribute does not specify explicit or implicit display<br><br>4. And, in character comparison of (3), the character code range is the following.<br><br>- X"8181" to X"819F"<br><br>- X"81E0" to X"81FC"<br><br>5. And, in either of the object of comparisons, the character immediately after the character of (4) is the national spaces(X"8140").<br><br>6. And, in either of (5),<br><br>- there is an ANK space (X"2020") of 2 bytes in the same character position as a national space of (5).<br>or<br><br>the character position of (4) is an end of data item. |
| 6 | V10.1.0 to V10.4.0 | PG97090 | The problem of not correctly replacing the alternative character specified for argument-2 was corrected when there was no national character corresponding to the character converted by the NATIONAL-OF function when the COBOL program was executed under the following conditions.<br><br>1. It is a program compiled using compiler option RCS(UTF16,BE).<br><br>2. And, the NATIONAL-OF function is used.<br><br>3. And, Argument-2 is specified for the function of No.2.<br><br>And, data that is not alphanumeric characters is specified for argument-1 in No.2 above, and the code conversion error (there is no corresponding National character) occurs internally. |
| 7 | V10.1.0 | PG79859 | Under the following conditions, when COBOL application is debugged by remote debugging facility of NetCOBOL Studio, the message that the following do not anticipate while debugging it is output. |

| No. | V/L(*) | P number | Content |
|---|---|---|---|
| | | | There is no disk in the drive. Please insert the disk in the drive.<br><br>1. COBOL application is debugged by remote debugging facility of NetCOBOL Studio.<br><br>2. The load module that operates in the process to be diagnosed makes DEBUG information file (*) (PDB file).<br><br>3. In the folder where the load module is stored in 2, there is no DEBUG information in 2.<br><br>4. The system that uses NetCOBOL drives the state that cannot be referred by the un-mount of virtual CD drive.<br><br>5. The drive character of the drive that makes DEBUG information file in 2 and the drive that cannot be referred in 4 is corresponding.<br><br>* : DEBUG information file (PDB file) is generated when the linkage option "/DEBUG" is specified. |
| 8 | V10.1.0 | PG79852 | Under the following conditions, when the message of the application error or execution time is generated in the running of COBOL application, the message that the following do not anticipate before outputting diagnostic report is output in COBOL Error Report.<br><br>There is no disk in the drive. Please insert the disk in the drive.<br><br>1. Environment variable @CBR_JUSTINTIME_DEBUG=NO is not specified.<br><br>2. The load module that operates in the process to be diagnosed makes DEBUG information file (*) (PDB file).<br><br>3. In the folder where the load module is stored in 2, there is no DEBUG information in 2.<br><br>4. The system that uses NetCOBOL drives the state that cannot be referred by the un-mount of virtual CD drive.<br><br>5. The drive character of the drive that makes DEBUG information file in 2 and the drive that cannot be referred in 4 is corresponding.<br><br>* : DEBUG information file (PDB file) is generated when the linkage option "/DEBUG" is specified. |
| 9 | V10.1.0 | PG77099 | Under the following conditions, the execution environment variable @CBR_SSIN_FILE=THREAD doesn't become effective, and one input file is shared by file input of ACCEPT statement in the process. (*)<br><br>1. It is an application that operates by multi-thread.<br><br>2. The execution environment variable @CBR_SSIN_FILE=THREAD is specified.<br><br>3. ACCEPT statement is executed, and data is input from the file.<br><br>* : When the execution environment variable @CBR_SSIN_FILE=THREAD is specified, input file of each thread can be opened by file input of ACCEPT statement. |
| 10 | 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.<br><br>1. Compiler option RCS(UCS2,LE) or RCS(UTF16,LE) is specified.<br><br>2. And, there is a size comparison of National item and National item or National language character constant.<br><br>3. And, at least one side is an item with the ANY LENGTH clause is specified or reference modification.<br><br>And, the length of the compared targets differs. |

| No. | V/L(*) | P number | Content |
|-----|--------|----------|---------|
| 11 | V10.1.0 | PG73346 | Under the following conditions, at the 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/U.<br><br>1. File operation using COBOL file system (*) is done.<br><br>2. Either of the following I/O statements is executed.<br><br>  - READ statement<br><br>  - WRITE statement<br><br>  - REWRITE statement<br><br>  - DELETE statement<br><br>  - START statement<br><br>3. The execution of I/O statement failed.<br><br>* : It doesn't correspond when BSAM is specified. |

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

# 2.3 J Adapter Class Generator

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

- NetCOBOL Enterprise Edition Server Runtime (64bit) for Windows

## 2.3.1 Code conversion when application is executed

### Content

The replacement character used has changed when Shift JIS is specified as code set during execution and data that is out of Shift JIS is included in character strings that are got by using GET-STRING-X method or GET-STRING-N method.

- Before V10.1

Normal-width underscore "_"

- After V11.0

If the data before conversion is alphanumeric character, Normal-width underscore.

If the data before conversion is Japanese character, Double-byte underscore.

### Action

Specify the execute environment variable information @CBR_CONVERT_CHARACTER=SYSTEM expressly.

However, it is not possible to specify it by the program using the specification after V11.0.

Refer to environment variable information @CBR_CONVERT_CHARACTER on the User's Guide for details.

## 2.3.2 Error message output when code conversion error occurs

### Content

In the conversion of data using the java-lang-String class method (such as GET-STRING-X, GET-STRING-N, NEW-STRING-X and NEW-STRING-N), if the data is incorrect, a code conversion error will occur.

Then, in NetCOBOL V11.0 or later, the following error message is to be output.

```
CHARACTER CODE CONVERSION FAILED. ERRNO:DETAIL
```

**Action**

Modify incorrect data items to store character codes correctly.

### 2.3.3 Characters that the results of code conversion are different

**Content**

In operation of an application with Shift_JIS, some Unicode characters that are passed to Java have changed as follows.

| Shift_JIS | Unicode before V10.1 | Unicode after V11 |
|-----------|----------------------|-------------------|
| 8160 | U+301C | U+FF5E |
| 8161 | U+2016 | U+2225 |
| 817C | U+2212 | U+FF0D |
| 8191 | U+00A2 | U+FFE0 |
| 8192 | U+00A3 | U+FFE1 |
| 81CA | U+00AC | U+FFE2 |

**Action**

Change the setting that is before V10.1 by specifying the following environment variable.

```
COBJNI_CONVERT=SJIS
```

## 2.4 PowerFORM

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

- NetCOBOL Enterprise Edition Server Runtime (64bit) for Windows


There is no information on interchangeability.

## 2.5 Fujitsu mainframe floating-point arithmetic emulator

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows


There is no information on interchangeability.

## 2.6 PowerBSORT

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

- NetCOBOL Enterprise Edition Server Runtime (64bit) for Windows

## 2.6.1 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 | V7.0.0 | PH05442 | Under the following conditions, PowerBSORT cannot correctly check whether the reconstruction field and the selection field exist in the record. As a result of an error in usage or implementation, the output may produce unpredictable or abnormal results. 1. The bsortex command is used. 2. And, the record format is one of the following. - Binary file variable-length record form (-record recform=var), or - Text file fixed field specification (-record recform=txtfix) 3. And, two output file information options (-output) or more are specified. 4. And, the following options are specified by two output file information options (-output) or more. - Record reconstruction option (-output reconst=...) - Record selection option (-output include=.../omit=.../case=...) 5. And, the field in the record is specified for a reconstruction field for the record reconstruction option (form of "pos.len" or "pos.END"). 6. And, the maximum position of the reconstruction field specified by each output file information option (-output) is different. or, The maximum position of the selection field specified by each output file information option (-output) is different. |

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

# 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 applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

Table 3.1 Program fix list for the NetCOBOL Development Environment

| No. | V/L(*) | P number | Content |
|---|---|---|---|
| 1 | V10.1.0 to V12.2.0 | PH20793 | Under the following conditions, during installation, Microsoft Windows Error Reporting (WER) cannot be set.<br><br>1. When either OS is used.<br>  - Windows 10<br>  - Windows Server 2016<br>  - Windows Server 2019<br><br>2. And, NetCOBOL V12.2.0 is installed.<br><br>3. And, registry key to enable WER does not exist.<br>The registry key to enable WER:<br><br><code>"HKLM\Software\Microsoft\Windows\Windows Error Reporting\LocalDumps"</code> |

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

## 3.2 NetCOBOL Runtime Environment

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

- NetCOBOL Enterprise Edition Server Runtime (64bit) for Windows

Table 3.2 Program fix list for the NetCOBOL Runtime Environment

| No. | V/L(*) | P number | Content |
|---|---|---|---|
| 1 | V10.1.0 to V12.2.0 | PH20793 | Under the following conditions, during installation, Microsoft Windows Error Reporting (WER) cannot be set.<br><br>1. When either OS is used.<br>  - Windows 10<br>  - Windows Server 2016<br>  - Windows Server 2019<br><br>2. And, NetCOBOL V12.2.0 is installed.<br><br>3. And, registry key to enable WER does not exist.<br>The registry key to enable WER:<br><br><code>"HKLM\Software\Microsoft\Windows\Windows Error Reporting\LocalDumps"</code> |

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

## 3.3  J Adapter Class Generator

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

- NetCOBOL Enterprise Edition Server Runtime (64bit) for Windows


There is no information on the program correction.

## 3.4  PowerFORM

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

- NetCOBOL Enterprise Edition Server Runtime (64bit) for Windows


There is no information on the program correction.

## 3.5  Fujitsu mainframe floating-point arithmetic emulator

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows


There is no information on the program correction.

## 3.6  PowerBSORT

Information described here applies to the following products.

- NetCOBOL Enterprise Edition Developer (64bit) for Windows

- NetCOBOL Enterprise Edition Server Runtime (64bit) for Windows


There is no information on the program correction.

# 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 Assumption

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 Environments

There are following notes in the environments.

- This product must only be installed in a folder which has the only ASCII character set.

- When cobmkmf is used, the character outside ASCII cannot be used for the file name and the folder name of program resources.

## 4.3 Languages

The language used by this product (GUI, messages, etc.) is determined according to the setting of the display language of the execution environment of the Windows system.

By default, if the set display language cannot be used, English is selected as the default

The default of the display language of the user account can be set depending on the following criteria.

- For Windows 8.1 or later and beyond, Windows Server 2012 or later

  "Language" of Control Panel

- Windows 7 Enterprise/Ultimate, Windows Server 2008 R2

  "Keyboards and Languages" tab of "Region and Language" of Control Panel

- The other edition of Windows 7

  Note that for the other editions of Window 7, the display language default cannot be set by the Control Panel. Usually, the display language of the Windows system is the default of the display language of the user account.

### Compilation messages

In this compiler, in accordance with the UI language during compilation, the COBOL command messages are set to Japanese or English, accordingly.

### Runtime messages

The language of the Runtime Error Messages is decided when the runtime environment is established.

If there is no message of an applicable language, a Runtime Error Message is displayed in English.

### NetCOBOL Studio

- When you use NetCOBOL Studio, define the applicable language of OS when the NetCOBOL product is installed, the system locale (This is set on the "Administrative" of "Region and Language" dialog box of the Control Panel), the display language (This is set on the "display language" of the Control Panel), and the language of the Format (This is set on the "Formats" of "Region and Language" dialog box of the Control Panel).

- Match the language of the client to the language of the server when you use the remote development function of NetCOBOL Studio. When the language of the client and the server is different, the message of the server might not be able to be displayed with NetCOBOL Studio of the client.

**PowerFORM**

When you use PowerFORM, define the same OS language for the system locale (This is set on the "Administrative" of "Region and Language" dialog box of the Control Panel) and the display language (This is set on the "display language" of the Control Panel) when the NetCOBOL product is installed.

# 4.4 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.

## 4.4.1 Related products

The following NetCOBOL family products are not supported.

- FORM

- MeFt/Web

The following Fujitsu products are not supported.

- Interstage Charset Manager

- Interstage List Works

- Interstage Business Application Server

- PowerRDBconnector

- INTARFRM

🛑 Note

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

| Global version | Japanese version |
| --- | --- |
| PowerFORM RTS | MeFt |
| PowerBSORT | PowerSORT |

## 4.4.2 Specifications

The following specifications are different with the Japanese version.

**Currency symbol**

In this compiler, currency sign of OS is used as Currency Symbol character.

If character other than the currency sign of OS is used, the method of specifying may differ based on the character intended to use.

- Characters with same code as $(X'24') or \(X'5C')

Specified with/based on compile option CURRENCY.

- 1 byte character other than this

   CURRENCY SIGN clause is used, specified inside the source program.

**Note**

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

- Characters consisting of multiple bytes cannot be used as currency symbol character. For possible characters that can be used as currency symbol, please refer "CURRENCY SIGN clause" of COBOL Language Reference.

- If currency sign of OS cannot be used as currency symbol, in that case \ (X'5C') will be taken as currency symbol.

- If data is passed between compilation units with different currency symbols, results might not be as expected. Specify the compilation option CURRENCY or CURRENCY SIGN clause to use the same currency symbol.

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

## 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.

**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.

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

## Printing font

If you use a print file without a FORMAT clause, a default font name according to the setting of the display language of the execution environment of the Windows system will be used.

When the display language is Japanese, the default font name is "MS Mincho, MS Gothic" and with any other language, the default font name is "Courier New".

In this scenario, the printing font value can be changed by the environment variables.

## 4.4.3 PowerFORM

- The following function is not supported in Global version.

    - Making of Form Descriptors (PXD) for Class Interface.

    - Selection of Target System of the overlay pattern table. Target System becomes Windows fixed.

    - Overlay conversion of Template Image.

- When PowerFORM is used, the only ASCII characters can be used.

- When Form Descriptors is edited, the initial values of the size of the form and the font-name, etc. are different.

## 4.4.4 PowerFORM RTS

When the Form is output in a Global version, it is necessary to specify the following.

- Specify Unicode for an application code.

### 📚 See
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
Refer to "How to Use PowerFORM RTS" of "PowerFORM Runtime Reference" for details.
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

- Specify the font name by an English name.

### 📚 See
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
Refer to "Appendix C. Printer Information File" of "PowerFORM Runtime Reference" for details.

    - MAPFONT (Map font names)
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

- Specify the printer information file to which BOM(UTF-8) is added.

### 📚 See
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
Refer to "Chapter 5. Printer Information File" of "PowerFORM Runtime Reference" for details.
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

- Specify "USECHARTYPE UNI" with the printer information file. Adjust each character-code by "UNICODEN" and "UNICODEW" when the output result is not correct.

### 📚 See
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
Refer to "Appendix C. Printer Information File" of "PowerFORM Runtime Reference" for details.

    - USECHARTYPE (Unicode full-width/half-width character table)

    - UNICODEW (Unicode full-width character range)

    - UNICODEN (Unicode half-width character range)
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

- The default value of the font name is different according to the locale.

## See

Refer to "Appendix C. Printer Information File" of "PowerFORM Runtime Reference" for details.

- MAPFONT (Map font names)