Enterprise COBOL for z/OS
6.3

*What's New*

IBM

**Note**

Before using this information and the product it supports, be sure to read the general information under "Notices" on page 13.

**First edition (31 May 2024 update)**

This edition applies to Version 6 Release 3 of IBM® Enterprise COBOL for z/OS® (program number 5655-EC6) and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure that you are using the correct edition for the level of the product.

You can view or download softcopy publications free of charge in the Enterprise COBOL for z/OS library. Because Enterprise COBOL for z/OS supports the continuous delivery (CD) model and publications are updated to document the features delivered under the CD model, it is a good idea to check for updates once every two months.

It is our intention to update the product documentation for this release periodically, without updating the order number. If you need to uniquely refer to the version of your product documentation, refer to the order number with the date of update.

# Contents

# About this information

This information provides an executive overview of new and improved features in Enterprise COBOL for z/OS 6.3 and Enterprise COBOL for z/OS 6.3 with PTFs installed.

Throughout this information, "COBOL" or "Enterprise COBOL" means "IBM Enterprise COBOL for z/OS".

# How to send your comments

Your feedback is important in helping us to provide accurate, high-quality information. If you have comments about this information or any other Enterprise COBOL documentation, send your comments to: compinfo@cn.ibm.com.

Be sure to include the name of the document, the publication number, the version of Enterprise COBOL, and, if applicable, the specific location (for example, the page number or section heading) of the text that you are commenting on.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way that IBM believes appropriate without incurring any obligation to you.

# Chapter 1. Video: What is new in Enterprise COBOL for z/OS 6.3

Watch the video to get a quick overview of the major new features and enhancements in Enterprise COBOL for z/OS 6.3.

To discover all the new features and enhancements, refer to the complete list in Chapter 2, "What is new in Enterprise COBOL for z/OS 6.3 and COBOL 6.3 with PTFs installed," on page 3.

To watch more Enterprise COBOL for z/OS videos, visit the COBOL video gallery.

# Chapter 2. What is new in Enterprise COBOL for z/OS 6.3 and COBOL 6.3 with PTFs installed

This section lists new and improved features in Enterprise COBOL for z/OS 6.3 and Enterprise COBOL for z/OS 6.3 with PTFs installed.

Enterprise COBOL for z/OS supports the continuous delivery (CD) model so that you can receive new features and enhanced capabilities as soon as the code is ready without waiting for the next release. The below feature descriptions that begin with an APAR number are introduced in the PTFs that are shipped with Enterprise COBOL for z/OS V6.3. Install the latest PTFs using the FIXCAT feature of SMP/E to take advantage of these features. For more information on SMP/E FIXCATs, see Determining service required. For more information on PTFs and APARs for Enterprise COBOL for z/OS, see the COBOL for z/OS support page.

The enhancements that are described in this section also have an associated cross-reference to the COBOL publications for your convenience.

The changes mainly fall into the following categories:

- AMODE 64 (64-bit) support

  Enhanced language elements:

    POINTER and POINTER-32 phrases
    SYNCHRONIZED clause
    Intrinsic functions LENGTH, ULENGTH, UPOS, USUPPLEMENTARY, and UVALID
    Support for compiling programs that contain UTF-8 data items with LP(64) (service)
    Support for compiling programs that contain dynamic-length elementary items with LP(64) (service)
    DATA directive
    Dynamic call between AMODE 31 and AMODE 64 programs

- UTF-8 support

    New USAGE clause of UTF-8

- Support of 2002 and 2014 COBOL standards

  New DYNAMIC  LENGTH clause

  The REPOSITORY paragraph FUNCTION specifier INTRINSIC

  New date and time intrinsic functions (service)

    COMBINED-DATETIME
    FORMATTED-CURRENT-DATE
    FORMATTED-DATE
    FORMATTED-DATETIME
    FORMATTED-TIME
    INTEGER-OF-FORMATTED-DATE
    SECONDS-FROM-FORMATTED-TIME
    SECONDS-PAST-MIDNIGHT
    TEST-DATE-YYYYMMDD
    TEST-DAY-YYYYDDD
    TEST-FORMATTED-DATETIME

- JSON enhancements

| Statement | Description |
|---|---|
| `JSON GENERATE` | **New phrases (service):**<br><br>`NAME is OMITTED`<br>`CONVERTING`<br>`when-phrase`<br>`generic-suppression-phrase` |
| `JSON PARSE` | **New phrases (service):**<br><br>`CONVERTING` |

- Other language element changes

    `POINTER-32` phrase added to USAGE clause
    Restored capability of `CALL ... USING` (service)
    New UUID4 intrinsic function (service)

- Compiler option changes

| New | Changed | Deprecated |
|---|---|---|
| LP | ARCH | ZONEDATA (service) |
| TUNE (service) | INITCHECK (service) | |
| INVDATA (service) | NUMCHECK (service) | |
| CONDCOMP(service) | TEST (service) | |
| | RULES (service) | |
| | SOURCE (service) | |
| | OFFSET (service) | |

- Runtime changes

    – PERFORM ... UNTIL EXIT support

    – VSAM dynamic access optional logic path

    – New runtime options

- Migration assistance

    New runtime option `IGZCOMPAT` for MERGE statement
    New option `QSAMBUFFINITCHAR` added to `IGZUOPT` module

- Listing changes
- Installation customization changes
- Compiler message changes

| New | Changed | Removed |
|---|---|---|
| IGYCB7315-W | IGYSI5305-U (service) | IGYPS0212-S |
| IGYCB7316-W | IGYSI5306-U (service) | |
| IGYCB7317-W | | |
| IGYCB7318-W | | |

- COBOL/JNI interface enhancement

## AMODE 64 (64-bit) support

You can now develop AMODE 64 (64-bit) applications with Enterprise COBOL.

Consider the changes in development and compilation. (Compiling, binding, and running COBOL AMODE 64 applications)

The following language elements are enhanced to provide the AMODE 64 support:

**Using POINTER and POINTER-32 phrases under LP(64)**
When LP(32) is in effect, the USAGE POINTER and USAGE POINTER-32 data types are synonyms. However, they behave differently when LP(64) is in effect.

- A POINTER data item can be SET to a value from a POINTER-32 data item. The high-order word of the POINTER data item is cleared to zero. (POINTER phrase)
- A POINTER-32 data item can be SET to a value from a POINTER data item. Only the low-order word of the POINTER data item is used in this case. It is a programming error if the high-order word of the AMODE 64 POINTER data item is not zero. (POINTER-32 phrase)

**Using SYNCHRONIZED clause under LP(64)**
If the SYNCHRONIZED clause is specified for data items that have the following usages, the items will be aligned on a fullword boundary if LP(32) is in effect, or on a doubleword boundary if LP(64) is in effect. (SYNCHRONIZED clause)

Note the following usages when you compile code under LP(64):

- USAGE FUNCTION-POINTER (FUNCTION-POINTER phrase)
- USAGE INDEX (INDEX phrase)
- USAGE OBJECT REFERENCE (OBJECT REFERENCE phrase)
- USAGE POINTER (POINTER phrase)
- USAGE POINTER-32 (POINTER-32 phrase)
- USAGE PROCEDURE-POINTER (PROCEDURE-POINTER phrase)

**Using intrinsic functions under LP(64)**
When you compile with LP(64), you must be aware that the returned value is an 18-digit integer. On the other hand, when you compile with LP(32), the returned value of the following intrinsic functions is a 9-digit integer.

- LENGTH (LENGTH)
- ULENGTH (ULENGTH)
- UPOS (UPOS)
- USUPPLEMENTARY (USUPPLEMENTARY)
- UVALID (UVALID)

**Compiling programs that contain UTF-8 data items with LP(64)**
PH18638: You can compile programs with the LP(64) option when the program contains UTF-8 data items.

- Symbols used in the PICTURE clause
- UTF-8 literals
- Defining UTF-8 data items

**Compiling programs that contain dynamic-length elementary items with LP(64)**
PH18640: You can compile programs with the LP(64) option when the program contains dynamic-length elementary items. (Dynamic-length items)

**DATA directive**
PH40311: A new DATA directive is added and is supported when LP(64) is in effect. (DATA)

**Dynamic call between AMODE 31 and AMODE 64 programs**

- Runtime APAR PH40348 (V2R3/V2R4): Dynamic call is supported in a mixed AMODE 31/AMODE 64 environment.
- Runtime APAR PH59864 (V2R5 or later): When PTFs for LE APAR PH56800 and COBOL Runtime APAR PH59864 are installed on z/OS 2.5 or later systems where COBOL programs run, the AMODE 31 subprograms called by the caller program can be either a DLL or a non-DLL.

(Dynamic call between AMODE 31 and AMODE 64 programs)

## UTF-8 support

With the native language support for UTF-8, you can now define, move, and compare UTF-8 data items.

- The native support for UTF-8 data items is provided by a new USAGE clause of UTF-8, plus a picture symbol 'U', which together define a new class, category, and USAGE of data in Enterprise COBOL. (Symbols used in the PICTURE clause)
- The UTF-8 literal formats that Enterprise COBOL provides are basic UTF-8 literals and hexadecimal notation for UTF-8 literals. (UTF-8 literals)
- You can work directly with UTF-8 data. (Using UTF-8 data (Unicode) in COBOL)

PH47577: When you use the figurative constant HIGH-VALUE in a context that requires UTF-8 characters, its value is UTF-8 character UX'F48FBFBF' corresponding to Unicode code point U+10FFFF, except when HIGH-VALUE is used in a move or compare operation with a fixed byte-length UTF-8 data item that has a length that is not a multiple of 4 bytes. (Using UTF-8-character figurative constants)

## Support of 2002 and 2014 COBOL standards

**New DYNAMIC LENGTH clause**
Use this clause to specify a dynamic-length elementary item. A dynamic-length elementary item is a data item whose length might change at run time. This is part of the 2014 COBOL Standard. (Dynamic-length items)

**The REPOSITORY paragraph FUNCTION specifier INTRINSIC**
INTRINSIC allows declaration of intrinsic function names that may be used without specifying the word FUNCTION. This is part of the 2002 COBOL Standard. (REPOSITORY paragraph)

**PH31047: New date and time intrinsic functions**
New date and time intrinsic functions are introduced that support encoding and decoding of date and time information to and from formats specified in ISO 8601, and that support encoding and decoding date and time information to and from integers that are suitable for arithmetic.

**Note:** COBOL Runtime LE APAR PH31133 must also be applied on all systems where programs that make use of these new date and time intrinsic functions are linked or run.

**Support for 2002 COBOL Standard:**

- **TEST-DATE-YYYYMMDD**: Tests whether a date in standard date form (YYYYMMDD) is a valid date in the Gregorian calendar. (TEST-DATE-YYYYMMDD)
- **TEST-DAY-YYYYDDD**: Tests whether a date in Julian date form (YYYYDDD) is a valid date in the Gregorian calendar. (TEST-DAY-YYYYDDD)

**Support for 2014 COBOL Standard:**

- **COMBINED-DATETIME**: Combines a date in integer date form and a time in standard numeric time form into a single numeric item from which both date and time components can be derived. (COMBINED-DATETIME)
- **FORMATTED-CURRENT-DATE**: Returns a character string that represents the current date and time provided by the system on which the function is evaluated. (FORMATTED-CURRENT-DATE)

- **FORMATTED-DATE**: Converts a date from its integer date form to the requested format. (FORMATTED-DATE)
- **FORMATTED-DATETIME**: Uses a combined time and date format to convert and combine a date in the integer date form and a numeric time expressed as seconds past midnight to a formatted date and time representation according to that combined date and time format. (FORMATTED-DATETIME)
- **FORMATTED-TIME**: Uses a format to convert a value that represents seconds past midnight to a formatted time of day in the requested format. (FORMATTED-TIME)
- **INTEGER-OF-FORMATTED-DATE**: Converts a date that is in a specified format to an integer date form. (INTEGER-OF-FORMATTED-DATE)
- **SECONDS-FROM-FORMATTED-TIME**: Converts a time that is in a specified format to a numeric value that represents the number of seconds after midnight. (SECONDS-FROM-FORMATTED-TIME)
- **SECONDS-PAST-MIDNIGHT**: Returns a value in standard numeric time form that represents the current local time of day provided by the system on which the function is evaluated. (SECONDS-PAST-MIDNIGHT)
- **TEST-FORMATTED-DATETIME**: Tests whether a data item that represents a date, a time, or a combined date and time is valid according to the specified format. (TEST-FORMATTED-DATETIME)

## JSON enhancements

You can watch this video to get an overview of the JSON support in Enterprise COBOL for z/OS V6.

**New NAME is OMITTED phrase**
PH18641: A new NAME is OMITTED phrase is added to the JSON GENERATE statement to allow generation of an anonymous JSON object, whose top-level parent name is not generated. (JSON GENERATE statement)

**New CONVERTING phrase**
PH26789: A new CONVERTING phrase is added to the JSON GENERATE and JSON PARSE statements so that you can generate and parse JSON boolean values:

- JSON GENERATE statement
- JSON PARSE statement
- Producing JSON output
- Processing JSON input

**Note:** COBOL Runtime LE APAR PH26698 must also be applied on all systems where programs that make use of this new feature are linked or run.

**New when-phrase and generic-suppression-phrase**
PH30975: New when-phrase and generic-suppression-phrase are added to the JSON GENERATE statement so that you can conditionally suppress data items during JSON GENERATE. (JSON GENERATE statement)

**Note:** COBOL Runtime LE APAR PH31172 must also be applied on all systems where programs that make use of this new feature are linked or run.

## Other language element changes

**POINTER-32 phrase added to USAGE clause**
A POINTER-32 phrase is added to the USAGE clause, which can be used to define pointer data items or data-pointers. (POINTER-32 phrase)

**Restored capability of CALL ... USING**
PH20724: The use of passing a file-name to a subprogram with the USING phrase of the CALL statement was removed in Enterprise COBOL V6.3, but is restored in Enterprise COBOL V6.3 with PTF for APAR PH20724 installed.

- CALL statement
- Passing data

### New UUID4 intrinsic function

- PH20997: The UUID4 intrinsic function is introduced.

  **Note:** COBOL Runtime LE PTF UI66560(V2R2)/UI66555(V2R3)/UI66557(V2R4) must also be applied on all systems where programs that make use of this new feature are linked or run.

- Runtime APAR PH34885 (V2R3/V2R4): The UUID4 randomness and UUID4 intrinsic function requires significant CPU usage.

  (UUID4)

## Compiler option changes

### New compiler options

**LP**
Use LP to indicate whether an AMODE 31 (31-bit) or AMODE 64 (64-bit) program should be generated with the related language features enabled. LP(32) is the default. (LP)

**TUNE**
PH34804: Use TUNE to specify the architecture for which the executable program will be optimized. (TUNE)

**INVDATA**
PH37328: INVDATA replaces the deprecated ZONEDATA compiler option and provides users fine-grained control over how the compiler generates code to handle USAGE DISPLAY and USAGE PACKED-DECIMAL data items that contain invalid data. (INVDATA)

**CONDCOMP**
PH49504: CONDCOMP compiler option is introduced to control how conditional code will be displayed in the listing. (CONDCOMP)

### Modified compiler options

**ARCH**
ARCH(7) is no longer accepted.

A new higher level of ARCH(13) is accepted.

ARCH(8) is the default.

(ARCH)

**INITCHECK**
PH22581: New suboptions LAX | STRICT are added to control whether the compiler will issue warning messages for data items unless they are initialized on at least one, or on all, logical paths to a statement. (INITCHECK)

**NUMCHECK**

Regardless of whether NUMCHECK(MSG) or NUMCHECK(ABD) is specified, invalid data found at compile time will produce a compile-time error message and the check will be removed.

PH27536: NUMCHECK(ZON) provides new suboptions LAXREDEF | STRICTREDEF to control whether the compiler will check and issue warning messages for redefined items.

PH29542: NUMCHECK(BIN) provides new suboptions TRUNCBIN| NOTRUNCBIN to control whether the compiler will generate the checking code for binary data items.

PTF UI71591 (no APAR number): NUMCHECK is enhanced with the capability to check alphanumeric senders whose contents are being moved to a numeric receiver. For alphanumeric senders whose contents are being moved to a numeric receiver, the compiler treats the sender as a numeric integer so NUMCHECK generates an implicit numeric class test for each alphanumeric sender.

PH40356: NUMCHECK(ZON(LAX)): When a zoned decimal data item is moved to another zoned decimal data item, NUMCHECK will not check the sender of the move. However, if the sender is subsequently used in a numeric context, it will be checked.

PH49715: NUMCHECK: The NUMCHECK compiler option is updated to avoid generating runtime checking code of zoned-decimal senders in MOVE statements when the receiver is an alphanumeric data item and NUMCHECK(ZON(LAX)) is in effect.

(NUMCHECK)

**TEST**

Runtime APARs PH29755(V2R3/V2R4) and PH30338(V2R3/V2R4 AMODE 64): New support is added for LLA/VLF managed programs where DWARF diagnostic information is included. (TEST)

**RULES**

PH33122: New suboptions LAXREDEF | NOLAXREDEF are added to inform users of redefined items with mismatched lengths. (RULES)

**SOURCE**

PH35643: New suboptions DEC | HEX are added to control whether the generated sequence numbers for the listing of the source are in decimal or hexadecimal format. (SOURCE)

**OFFSET**

PH35652: The OFFSET option behavior is changed. If there are multiple blocks of instructions for a single line of COBOL code, multiple entries will be generated for those instructions in the OFFSET table. (OFFSET)

**Deprecated compiler options**

**ZONEDATA**

PH37328: This compiler option is deprecated but tolerated and is automatically mapped to an equivalent form of the new INVDATA compiler option. (ZONEDATA)

**NUMCHECK(ZON(LAXREDEF|STRICTREDEF))**

PH40356: LAXREDEF|STRICTREDEF option is deprecated but is tolerated for compatibility, and it is replaced by the LAX|STRICT option. (NUMCHECK)

## Runtime changes

**PERFORM ... UNTIL EXIT support**

PH47372: You can specify EXIT in place of a condition in a PERFORM statement. If the UNTIL phrase with the EXIT reserved word is specified, execution proceeds exactly as if the same PERFORM statement were coded with *condition-1* specified, except that *condition-1* never evaluates as true. (PERFORM with UNTIL phrase)

**VSAM dynamic access optional logic path**

PH56036 and PH56037: An optional alternate logic path is introduced for VSAM files that use the ACCESS IS DYNAMIC mode. The alternate logic path uses a direct read-by-key request instead of a point to a record by key. (VSAM dynamic access optional logic path)

**New runtime options**

PH56036 and PH56037: The following COBOL runtime options are added:

- VSAMDYNAMICDIR: Changes the dynamic access method of VSAM to a direct read-by-key request. (VSAMDYNAMICDIR)

- DISABLEUOPTREPORT: Suppresses the COBOL runtime options report. (DISABLEUOPTREPORT)

## Migration assistance

**New runtime option IGZCOMPAT for MERGE statement**

Runtime APARs PH20569(V2R2) and PH21261(V2R3/V2R4): A new runtime option IGZCOMPAT for MERGE statement is introduced to obtain support for DFSORT option NOBLKSET and the conventional merge method for Enterprise COBOL V5 or later versions.

- Sorting and merging files
- Using DFSORT option NOBLKSET

**New option `QSAMBUFFINITCHAR` added to `IGZUOPT` module**
Runtime APAR PH25917: A new option QSAMBUFFINITCHAR is added to the IGZUOPT module that allows you to control the initial character used for QSAM buffer initialization. (Controlling initialization of QSAM buffer)

## Listing changes

- Listing terminologies change as follows:
  - `Static Map` is changed to `INITIAL HEAP STORAGE MAP`. (Example: initial heap storage map)
  - `Writeable static area (WSA)` is changed to `storage`. (Reading LIST output)
  - `WSA24` is changed to `BELOW THE LINE STORAGE`. (Example: initial heap storage map)
  - `Automatic Map` is changed to `STACK STORAGE MAP`. (Example: DSA memory map (stack storage map))
- PH40562: A new signature information byte is added for the `DECIMAL-POINT IS COMMA` clause so that the compiler can indicate to the debugger when COBOL programs have the `DECIMAL-POINT IS COMMA` clause. (Signature information bytes)

## Installation customization changes

- The installation customization for placing compiler phases into shared storage is removed.
- PH37331: Adds support for diagnosing miscoded options or options coded as `OPTION()` instead of `OPTION=` in the COBOL customization macro. (Changing the defaults for compiler options)

## Compiler message changes

**New messages**
- IGYCB7315-W: A reference modifier with a non-positive value was found. (IGYCB7315-W)
- IGYCB7316-W: A reference-modification length with a non-positive value was found. (IGYCB7316-W)
- IGYCB7317-W: A reference-modification length value caused reference to be made beyond the rightmost character of &1. (IGYCB7317-W)
- IGYCB7318-W: A reference-modifier start value exceeded the number of characters in &1. (IGYCB7318-W)

**Changed messages**
PH42280: The following messages are updated to report additional information and return codes:
- IGYSI5305-U (IGYSI5305-U)
- IGYSI5306-U (IGYSI5306-U)

**Removed messages**
- IGYPS0212-S

## COBOL/JNI interface enhancement

Runtime APAR PH37101(V2R3/V2R4): An enhancement is made to assist COBOL programs running in AMODE 31 to interact with Java™ programs in AMODE 64. (COBOL programs in AMODE 31 interacting with Java programs in AMODE 64)

**Note:** IBM SDK, Java Technology Edition V8.0.6.35 (JVM) is needed for this enhancement.

## Debugging enhancement

Runtime APAR PH61133 (V2R4 or later): You can use ddname `IGZPROUT` at the run step of your JCL to generate a report of all COBOL 5 or later programs that are called dynamically, statically, and via DLL linkage. (Generating a report of dynamically called COBOL programs)

## Prefer video?

To get a quick overview of the major new features and enhancements in Enterprise COBOL for z/OS 6.3, watch the Chapter 1, "Video: What is new in Enterprise COBOL for z/OS 6.3," on page 1.

# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law**: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who want to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

**COPYRIGHT LICENSE:**

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 1991, 2020.

**PRIVACY POLICY CONSIDERATIONS:**

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, or to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at http://www.ibm.com/privacy and IBM's Online Privacy Statement at http://www.ibm.com/privacy/details in the section entitled "Cookies, Web Beacons and Other Technologies,"

and the "IBM Software Products and Software-as-a-Service Privacy Statement" at http://www.ibm.com/software/info/product-privacy.

## Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Other company, product, or service names may be trademarks or service marks of others.

# Enterprise COBOL for z/OS publications

## COBOL for z/OS publications

You can find the following publications in the Enterprise COBOL for z/OS library:

- *What's new*
- *Customization Guide,* SC27-8712-02
- *Language Reference,* SC27-8713-02
- *Programming Guide,* SC27-8714-02
- *Migration Guide,* GC27-8715-02
- *Performance Tuning Guide,* SC27-9202-01
- *Messages and Codes,* SC27-4648-01
- *Program Directory,* GI13-4526-02
- *Licensed Program Specifications,* GI13-4532-02

## Softcopy publications

The following collection kits contain Enterprise COBOL and other product publications. You can find them at http://www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss.

- *z/OS Software Products Collection*
- *z/OS and Software Products DVD Collection*

## Support

If you have a problem using Enterprise COBOL for z/OS, see the following site that provides up-to-date support information: https://www.ibm.com/support/home/product/B984385H82239E03/Enterprise_COBOL_for_z/OS.

# Index

**IBM** ®

Product Number:   5655-EC6