



# COBOL-IT® Compiler Suite Enterprise Edition

## Compiler & Runtime Reference Manual

### Version 4.2



# Acknowledgment

This documentation is derived from COBOL-IT Source code, parts of which are derived from OpenCOBOL.

Copyright (C) 2002-2007 Keisuke Nishida

Copyright (C) 2007 Roger While

Copyright (C) 2008-2020 COBOL-IT

In 2008, COBOL-IT forked its own compiler branch, with the intention of developing a fully featured product and offering professional support to the COBOL user industry.

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this manual under the conditions for verbatim copying, provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this manual into another language, under the above conditions for modified versions, except that this permission notice may be stated in a translation approved by the Free Software Foundation.

## Conventions used in the General Format diagrams:

**Brackets [ ]** identify syntax elements that are supported but not required.

**Curly Braces { }** identify alternative syntax elements. Among syntax elements described within stacked curly braces, only one of the entries may be selected.

**Ellipses (...)** indicate (optional) repetition. If the syntax element is a required element, then it will be surrounded by curly braces.

**Copyright 2008-2020 COBOL-IT S.A.R.L. All rights reserved. Reproduction of this document in whole or in part, for any purpose, without COBOL-IT's express written consent is forbidden.**

COBOL-IT® Developer Studio, COBOL-IT® Sort (CitSORT®), COBOL-IT® MF Command Line Emulator (CitEMUL®), COBOL-IT® Lib Optimizer are registered trademarks of COBOL-IT, S.A.R.L All rights reserved.

The CitSQL® family: COBOL-IT® Precompiler for MySQL, COBOL-IT® Precompiler for PostgreSQL. COBOL-IT® Precompiler for Microsoft SQL Server, are registered trademarks of COBOL-IT. All rights reserved.

COBOL-IT® Precompiler for MySQL, COBOL-IT® Precompiler for PostgreSQL. COBOL-IT® Precompiler for Microsoft SQL Server are licensed by COBOL-IT under exclusive license with the Raincode Company.

---

**Third-Party software components embedded in the SOFTWARE and Services and submitted to specific licenses:**

**VBISAM**

- \* Copyright (C) 2003 Trevor van Bremen
- \* Copyright (C) 2008-2020 COBOL-IT
- \* License: LGPL

**GMP (GNU Multiprecision Library)**

- \* Copyright 1991, 1996, 1999, 2000, 2007 Free Software Foundation, Inc.
- \* License: LGPL

**GNU LIBICONV**

The libiconv libraries and their header files are under LGPL.

Microsoft and Windows are registered trademarks of the Microsoft Corporation. UNIX is a registered trademark of the Open Group in the United States and other countries. Other brand and product names are trademarks or registered trademarks of the holders of those trademarks.

---

**Contact Information:**

COBOL-IT  
231, rue Saint-Honoré - 75001 Paris - FRANCE  
Tel.: +33 1 75 43 05 50  
Fax: +33 1 75 43 05 16  
Email: [contact@COBOL-it.com](mailto:contact@COBOL-it.com)  
[www.COBOL-it.com](http://www.COBOL-it.com)

**ACKNOWLEDGMENT .....2**
**COMPILER & RUNTIME REFERENCE .....19**
**The COBOL Compiler: cobc .....19**

Informational Flags .....	19
-check-codepage <codepage-id> .....	19
--help .....	20
--list-codepage .....	20
--list-intrinsics .....	20
--list-mnemonics .....	20
--list-reserved .....	20
--version, -V .....	20
Standard Flags.....	20
-b .....	20
-c .....	21
-codepage <codepage-id> .....	21
-conf=[+]<file> .....	21
-constant "key=value" .....	21
-debug.....	21
--debugdb [=<DebugDB-name>].....	22
-dump-config.....	22
-err <file> .....	22
-ext<extension>.....	22
-fixed .....	23
-free .....	23
-g .....	23
-initcall=<program-name> .....	23
-l <lib> .....	23
-linkage-desc=[program.xsd] .....	23
-m .....	23
-makesyn "oldvalue=newvalue" .....	23
-o <file>   <dir>.....	24
-preprocess=<CMD> [input file] .....	24
-save-temps (= <dir>).....	24
-source-codepage <codepage-id>.....	24
-std=<dialect> .....	25
-sysin=<input file>.....	25
--sysout=<output file> [,S/L [,Min [,Max]]].....	25
-t <file>   <dir>.....	26
-use-extfh <NAME> .....	27
-use-extsm <NAME> .....	27
-v .....	27
-x .....	27
--xdd-prefix=<dir> .....	27
-C.....	28
-D <define> .....	28
-E.....	28
-G .....	28
-I <path>[,ext1,ext2,..,extn][@<LibName>]   <command-file> .....	29
-L <directory> .....	29
-MF <file>.....	29
-MT <target>.....	29
-O, -Os, -O2 .....	29
-R <directory>.....	30
-S.....	30
-Wc CC_opt .....	30

-W1 LD_opt.....	30
Guidelines for Searching and Locating COPY files .....	31
Guidelines for enforcing bounds-checking .....	34
Guidelines for optimizing performance .....	35
-O compiler flags.....	35
Optimizations enabled with the -O compiler flag.....	35
COB_OPTSIZE_FLAG .....	36
The CALL statement .....	37
The PERFORM statement.....	37
Resolving File Names .....	37
Removing debug-oriented compiler flags .....	38
Optimizing compiler flags set by default .....	39
Guidelines for use of -preprocess=cmd.....	39
Compiler -f Flags .....	41
-f77-opt.....	41
-faccept-with-auto .....	41
-faccept-with-update .....	42
-falign-8.....	42
-fall-external-call.....	42
-fall-external-link .....	42
-faloc-unused-linkage.....	42
-fas400-like .....	42
-fauto-load-symb .....	43
-fautolock .....	43
-fauto-sprwr.....	43
-fbdb .....	43
-fbinary-byteorder-big-endian.....	43
-fbinary-byte-order-native.....	43
-fbin-opt .....	43
-fbin-opt-strict .....	43
-fcall-comp5-as-comp .....	44
-fcall-lowercase .....	44
-fcall-opt.....	44
-fcall-uppercase .....	44
-fcarealia-sign.....	44
-fC-cmd-line.....	45
-fC-data-init.....	45
-fcheckpoint .....	45
-fcics.....	45
-fcmp-inline.....	45
-fcmp-opt.....	45
-fcobol-lines .....	45
-fcompat-display-to-int .....	46
-fcompute-ibm.....	46
-fcompute-ibm-trunc .....	46
-fcontinuation-line.....	46
-fcopy-default-leading .....	46
-fcopy-exec-replace.....	46
-fcopy-mark.....	47
-fcopy-partial-replace .....	47
-fctree .....	47
-fctree-field-numbering .....	47
-fctree-no-full-qualification.....	47
-fcudir-include .....	47
-fdebugdb .....	48
-fdebug-exec.....	48
-fdebugging-line.....	48
-fdebug-parser .....	48

-fdecimal-optimize .....	48
-fdisam .....	48
-fdisplay-dos.....	48
-fdisplay-ibm.....	49
-fdiv-check .....	49
-febcdic-charset.....	49
-femulate-vms .....	49
-fexclusivelock.....	49
-fexec-check .....	49
-fexpand-exec-copy.....	49
-fexpand-sql-include .....	50
-ffast-figurative-move .....	50
-ffast-op.....	50
-ffcdreg.....	50
-ffdclear.....	51
-ffile-auto-external .....	51
-ffold-copy-lower .....	52
-ffold-copy-upper .....	52
-ffp-opt .....	52
-ffree-thread-safe-data.....	52
-ffunctions-all.....	52
-ffunctions-all-intrinsic .....	52
-fgcc .....	52
-fgcc-bug .....	52
-fgcc-goto.....	52
-fgcc-O-bug.....	53
-fgcos-mode .....	53
-fgen-xdd.....	53
-fglobal-typedef.....	53
-fibm-listing-macro .....	53
-fibm-mainframe .....	53
-fibm-sync .....	53
-fimplicit-init .....	53
-finclude-main.....	53
-fincomplete-subscript.....	54
-findex-optimize.....	54
-finitialize-fd .....	55
-finitialize-opt.....	55
-fkeep-copy-statement.....	55
-fkeep-org-src-line.....	55
-fkeep-unused.....	55
-fline-seq-dos .....	55
-flink-only .....	55
-flisting-sources.....	55
-floosy-comment .....	55
-fls-expand-tab .....	56
-fmain .....	56
-fmain-as-object .....	56
-fmainframe-vb .....	56
-fmakesyn-patch-preprocess.....	56
-fmanuallock .....	56
-fmem-info .....	56
-fmfcomment.....	56
-mf-compat-parser .....	56
-mf-ctrl-escaped-parser.....	56
-mf-file-optional.....	57
-mf-gnt .....	57
-mf-hostnumcompare.....	57

-fmf-int.....	57
-fmf-relativefile.....	57
-fmodule-name-entry.....	58
-fmodule-uppercase.....	58
-fmove-all-edited.....	58
-fno-cbl-error-proc.....	58
-fno-realpath.....	58
-fnostrip.....	58
-fnotrunc.....	58
-fnull-param.....	58
-fnumeric-compare.....	58
-fnumval-validate.....	58
-fobj-cit.....	59
-fodo-slide.....	59
-foptimize-move.....	59
-foptimize-move-call.....	59
-foptional-file.....	59
-fperform-osvs.....	59
-fprepro_cut_line.....	60
-fprinter-crlf.....	61
-fprofiling.....	61
-fprotect-linkage.....	61
-fraw-by-value.....	61
-fraw-pic9-display (Internal use only).....	61
-fread-into-copy.....	61
-fready-trace.....	61
-frecmode-f.....	61
-frecmode-v.....	61
-frecord-depending-iso.....	62
-fregion0.....	62
-frelativefile-bigendian.....	62
-freplace-additive.....	62
-freturn-opt.....	62
-fround-fp.....	62
-frw-after-preprocess.....	62
-frw-mode-nopf.....	62
-frw-mode-nopf-dos.....	63
-fsafe-linkage.....	63
-fsequential-line.....	63
-fshare-all-autolock.....	63
-fshare-all-default.....	63
-fshare-all-manulock.....	64
-fsign-ascii.....	64
-fsign-ebcdic.....	64
-fsign-leading.....	65
-fsign-separate.....	65
-fsimple-trace.....	65
-fsource-location.....	65
-fsplit-debug-mark.....	65
-fstack-check.....	65
-fstatic-call.....	65
-fstrict-compare-low.....	65
-fstrict-record-contains.....	66
-fsyntax-only.....	66
-fthread-safe.....	66
-ftrace.....	66
-ftrace-ts.....	66
-ftrace-upon-sysout.....	66

-ftraceall .....	67
-ftrap-unhandled-exception .....	67
-ftruncate-listing .....	67
-funstring-use-move .....	68
-futf-8 .....	68
-futf16-le .....	68
-fvalidate-dep-on .....	68
-fvalidate-only .....	69
-fvalue-of-id-priority .....	69
-fvalue-size-is-auto .....	69
-fvbisam .....	69
-fvms-error-handler .....	69
-fxparse-event .....	69
Guidelines for handling Linkage Section parameters .....	69
Guidelines for modifying default handling of the LOCK MODE .....	70
Guidelines for use of Checkpoints .....	70
Guidelines for use of Profiler .....	71
Dumping Profiling Data at the Module Level .....	71
Using the PRAGMA statement to produce Profiling Reports .....	72
Info profiling debugging command .....	72
Attaching a program compiled with -fprofiling to a running process .....	73
Guidelines for thread-safe programs .....	73
Compiler -w Flags .....	74
-w .....	75
-Wall .....	75
-Warchaic .....	75
-Wbdb .....	75
-Wcall-params .....	75
-Wconstant .....	75
-Wimplicit-define .....	75
-Winformation .....	75
-Wlinkage .....	75
-Wobsolete .....	75
-Wparentheses .....	75
-Wredefinition .....	75
-Wstrict-typing .....	75
-Wsuggestion .....	75
-Wterminator .....	75
-Wtruncate .....	76
-Wunreachable .....	76
<b>Compiler Configuration File .....</b>	<b>77</b>
77-opt:[yes/no] .....	78
accept-but-ignore-comp6-signed:[yes/no] .....	78
accept-with-auto:[yes/no] .....	78
accept-with-update:[yes/no] .....	78
align-8:[yes/no] .....	78
all-external-call:[yes/no] .....	79
all-external-link:[yes/no] .....	79
alloc-unused-linkage:[yes/no] .....	79
as400-like :[yes/no] .....	79
assign-clause: [ COBOL2002 / mf / ibm / external ] .....	79
auto-initialize:[yes/no] .....	80
auto-load-symb:[yes/no] .....	80
autolock:[yes/no] .....	80
bdb:[yes/no] .....	80
bin-opt:[yes/no] .....	80
bin-opt-strict:[yes/no] .....	80



binary-byteorder: native / big-endian .....	81
binary-size: 2-4-8 / 1-2-4-8 / 1--8.....	81
binary-truncate:[yes/no] .....	81
bitfield-first-is-lsb: [yes/no] .....	82
call-comp5-as-comp:[yes/no].....	82
call-lowercase: [yes/no].....	82
call-opt: [yes/no] .....	82
call-uppercase: [yes/no].....	82
carealia-sign: [yes/no ] .....	83
C-cmd-line:[yes/no] .....	83
C-data-init:[yes/no] [ Internal use only ].....	83
check-linkage-bound: [yes/no] .....	83
checkpoint: [yes/no] .....	83
cics: [yes/no] .....	83
cmp-inline:[yes/no] .....	83
cmp-opt:[yes/no] .....	84
cobol-lines: [yes/no].....	84
codepage: <codepage-id>.....	84
comp5-byteorder: [native/big-endian].....	84
compat-display-to-int:[yes/no] .....	84
complex-odo:[yes/no] .....	84
compute-ibm:[yes/no] .....	84
compute-ibm-trunc: [yes/no].....	85
console-is-sysfile: [yes/no].....	85
constant: "key=value" .....	85
continuation-line .....	85
copy-default-leading:[yes/no] .....	85
copy-exec-replace[yes/no] .....	86
copy-partial-replace:[yes/no] .....	86
crtstatus-map:[cit-value] [user-value] .....	86
ctree: [yes/no].....	87
ctree-field-numbering: [yes/no].....	87
ctree-no-full-qualification: [yes/no] .....	87
curdir-include: [yes/no] .....	88
datacompress: <integer> .....	88
debug-exec: [yes/no] .....	88
debugging-line:[yes/no] .....	89
debug-parser: [yes/no].....	89
decimal-optimize:[yes/no].....	89
defaultbyte:[any integer] .....	89
defaultcall:[any integer] .....	89
disam:[yes/no].....	90
display-dos:[yes/no] .....	90
display-ibm:[yes/no] .....	90
displaynumeric-edited-mf50: [yes/no] .....	90
displaynumeric-mf50:[yes/no] .....	90
div-check: [yes/no].....	91
ebcdic-charset: [yes/no].....	91
emulate-vms [yes/no] .....	91
exception checking.....	92
exclusivelock: [yes/no].....	95
exec-check: [yes/no].....	95
exit-program-forced:[yes/no] .....	95
expand-exec-copy:[yes/no] .....	95
expand-sql-include:[yes/no] .....	96
external-link: <function name> .....	96
external-mapping:[yes/no] .....	96
fast-figurative-move: [yes/no].....	96

fcldreg: [yes/no] .....	96
fdclear:[yes/no] .....	97
file-auto-external:[yes/no] .....	97
filename-mapping:[yes/no] .....	97
first-tab-width:[any integer] .....	97
flat-extfh: <DRIVER NAME> .....	97
flat-extfh-lib: <library to use for this extfh driver> .....	97
fold-copy-lower: [yes/no] .....	97
fold-copy-upper:[yes/no] .....	98
fp-opt:[yes/no] .....	98
free-thread-safe-data: [yes/no] .....	98
fstatus-map:[cit-status] = [custom-status] .....	98
full-cancel:[yes/no] .....	98
functions-all:[yes/no] .....	99
functions-all-intrinsic:[yes/no] .....	99
gcc:[yes/no] .....	99
gcc-O-bug:[yes/no] .....	99
gcc-bug:[yes/no] .....	99
gcc-goto:[yes/no] .....	99
gcos-mode:[yes/no] .....	100
gen-xdd:[yes/no] .....	100
global-typedef:[yes/no] .....	100
ibm-listing-macro:[yes/no] .....	100
ibm-mainframe:[yes/no] .....	100
ibm-sync:[yes/no] .....	100
identifier-length:<max-length> .....	100
ignore-global-in-local-storage: [yes/no] .....	101
ignore-with-rollback: [yes/no] .....	101
implicit-init: [yes/no] .....	101
Include-main: [yes/no] .....	101
incomplete-subscript:[yes/no] .....	101
index-optimize:[yes/no] .....	101
indirect-redefines:[yes/no] .....	102
initcall:<program-name> .....	102
Initialize-fd:[yes/no] .....	102
initialize-filler:[yes/no] .....	102
initialize-opt:[yes/no] .....	103
initialize-pointer:[yes/no] .....	103
initialize-to-value:[yes/no] .....	103
isam-extfh: <DRIVER NAME> .....	103
isam-extfh-lib: <library to use for this extfh driver> .....	103
keep-copy-statement:[yes/no] .....	104
keep-org-src-line:[yes/no] .....	104
keep-unused:[yes/no] .....	104
key-dup-always-22:[yes/no] .....	104
keycompress: [ integer between 0 and 9 ] .....	104
larger-redefines-ok:[yes/no] .....	105
line-seq-dos:[yes/no] .....	105
line-seq-mf:[yes/no] .....	105
line-seq-notrunc:[yes/no] .....	105
line-seq-recording-mode:[yes/no] .....	106
line-seq-unix:[yes/no] .....	106
link-only: [yes/no] .....	106
listing-sources: [yes/no] .....	106
local-storage-guard: 8 (internal use only) .....	106
loosy-comment[yes/no] .....	106
ls-expand-tab:[yes/no] .....	106
ls-ignore-record-size:[yes/no] .....	107

ls-utf16:[yes/no] .....	107
main:[yes/no] .....	107
main-as-object:[yes/no] .....	107
mainframe-vb:[yes/no] .....	107
makesyn: oldvalue=newvalue .....	108
makesyn-patch-preprocess: [yes/no] .....	108
manuallock: [yes/no] .....	108
max-literal-expand: 32 (internal use only) .....	108
mem-info: [yes/no] .....	108
mfcomment: [yes/no] .....	109
mf-compat-parser: [yes/no] .....	109
mf-ctrl-escaped-parser: [yes/no] .....	109
mf-file-optional:[yes/no] .....	109
mf-gnt: [yes/no] .....	110
mf-hostnumcompare:[yes/no] .....	110
mf-int:[yes/no] .....	110
mf-relativefile :[yes/no] .....	110
module-load-priority:[yes/no] .....	111
module-name-entry: [yes/no] .....	111
module-uppercase: [yes/no] .....	111
move-all-edited: [yes/no] .....	111
move-high-low-to-displaynumeric [error/zero/value] .....	111
move-picx-to-pic9:[cit / mf50 / mf40 / mvs / raw / iso / none ] .....	112
move-spaces-to-comp3:[error/space/zero] .....	112
move-spaces-to-displaynumeric:[yes/no/error] .....	113
move-to-group-separated:[yes/no] .....	113
name:[any string] .....	113
no-realpath: [yes/no] .....	113
non-ibm-5.2-syntax: [(ok or yes)/(error or no)/warning] .....	114
nostrip: [yes/no] .....	114
notrunc: [yes/no] .....	114
not-reserved:[any reserved word] .....	114
null-param: [yes/no] .....	114
numeric-compare: [yes/no] .....	114
numeric-group: [yes/no/warning] .....	115
numval-validate: [yes/no] .....	115
obj-cit: [yes/no] .....	115
odo-slide: [yes/no] .....	115
optimize-move: [yes/no] .....	116
optimize-move-call:[yes/no] .....	116
optional-file: [yes/no] .....	116
pack-comp-4:[yes/no] .....	116
perform-osvs:[yes/no] .....	116
prepro_cut_line: [yes/no] .....	118
pretty-display:[yes/no] .....	118
printer-crlf:[yes/no] .....	118
profiling:[yes/no] .....	118
protect-linkage:[yes/no] .....	118
quote:[any single character] .....	118
raw-by-value: [yes/no] .....	118
raw-compare: [yes/no] .....	119
raw-pic9-display: [yes/no] .....	119
read-at-end-mf:[yes/no] .....	119
read-into-copy: [yes/no] .....	119
ready-trace:[y/n] .....	119
Enables paragraph tracing between READY TRACE and RESET TRACE procedural COBOL statements. ....	119
When set to yes, .....	119
In the interval between the READY TRACE and RESET TRACE statements, paragraph tracing output is written	

to the console in the format:.....	119
PROGRAM-ID: [program-id]: [paragraph name].....	119
recmode-f:[yes/no] .....	119
recmode-osvs:[yes/no] .....	120
recmode-v:[yes/no] .....	120
record-depending-iso:[yes/no] .....	120
redefine-identifier: [ error / warning / ok ] .....	120
region0: [yes/no] .....	121
relativefile-bigendian:[yes/no] .....	121
relax-bounds-check:[yes/no] .....	121
relax-level-hierarchy:[yes/no] .....	121
relaxed-syntax-check:[yes/no] .....	121
replace-additive:[yes/no].....	121
return-opt:[yes/no].....	122
round-fp:[yes/no].....	122
rtncode-size: <integer>.....	122
rw-after-preprocess:[yes/no] .....	122
rw-mode-nopf:[yes/no] .....	122
rw-mode-nopf-dos:[yes/no].....	123
safe-linkage:[yes/no] .....	123
screen-exceptions:[yes/no] .....	123
screen-raw-keys:[yes/no] .....	124
sequential-line:[yes/no] .....	124
share-all-autolock:[yes/no].....	124
share-all-default:[yes/no] .....	125
share-all-manulock:[yes/no].....	125
sign-ascii:[yes/no] .....	125
sign-ebcdic:[yes/no] .....	125
sign-leading:[yes/no].....	125
sign-separate:[yes/no] .....	126
signed-comp6-as-comp3:[yes/no] .....	126
simple-trace:[yes/no].....	126
source-location:[yes/no].....	126
split-debug-mark:[yes/no] .....	126
spzero:[yes/no] .....	126
stack-check:[yes/no].....	127
static-call:[yes/no] .....	127
static-link:[function-name].....	127
sticky-linkage:[yes / no / fixed / variable].....	127
strict-compare-low: [yes/no] .....	128
strict-record-contains:[yes/no] .....	128
synchronized-double-word-bound:[yes/no] .....	128
synchronized-propagate-to-occurs: [yes/no] (Internal Use Only) .....	129
synchronized-propagate-to-occurs-with-group-size: [yes/no] (Internal Use Only).....	129
syntax-only:[yes/no].....	129
syntax-support:[ok / archaic / obsolete / skip / ignore / unconformable / error].....	129
tab-width:[integer].....	130
tally-register:[yes/no] .....	130
text-column:[integer].....	130
thread-safe:[yes/no].....	130
trace:[yes/no].....	130
trace-ts: [yes/no].....	130
trace-upon-systout: [yes/no].....	131
traceall:[yes/no].....	131
trap-unhandled-exception:[yes/no].....	131
truncate-listing:[yes/no] .....	131
unstring-use-move:[yes/no].....	131
use-defaultbyte:[yes/no].....	131

utf16-le:[yes/no].....	132
utf-8:[yes/no].....	132
validate-dep-on: [yes/no].....	132
validate-odo:[yes/no].....	132
validate-only:[yes/no] .....	132
value-of-id-priority: [yes/no].....	132
value-size-is-auto: [yes/no] .....	133
variable-rec-pad-mf:[yes/no].....	133
vbisam: [yes/no].....	133
vms-error-handler: [yes/no].....	133
when-compiled-function-all :[yes/no].....	133
wnone:[yes/no].....	133
xparse-event:[yes/no] .....	133
zero-length-trim:[yes/no] .....	134

**Compiler Environment Variables ..... 135**

COB_AR <program> .....	135
COB_ARFLAGS <ar flags> .....	135
COB_CC <program> .....	135
COB_CFLAGS <cc flags> .....	135
COB_CONFIG_DIR<directory>.....	135
COB_CONSOLE_CP=<codepage-id> .....	135
COB_COPY_DIR<directory> .....	136
COB_EXTRA_FLAGS.....	136
COB_LDADD <ld flags> .....	136
COB_LDFLAGS <ld flags> .....	136
COB_LIBS <libs>.....	136
COB_OPTIMIZE_FLAG<cc flags>.....	136
COB_OPTSIZE_FLAG=[optimization flag] .....	136
COB_STDUNIX <1/0> .....	137
COB_SUNSTUDIO12=[Y/N] .....	137
COBCPY <directory list> .....	137
COBCTMP=<directory>.....	137
COBITOPT=[string of command-line compiler flags] .....	137
COBOPT=[string of command-line compiler flags] .....	138
COBOLITDIR=<directory> .....	138
TMPDIR or TMP=<directory> .....	138

**COBOL-IT Runtime Options ..... 138**

COBOL-IT runtime parameters .....	138
--checkpoint <file>.....	138
--console, -c.....	138
--debug, -d.....	138
--debug, -d --remote -r.....	138
--help, -h.....	139
--reload.....	139
--version, -V .....	139
COBOL-IT runtime environment variables .....	139
COB_CALL_CASE=xul [where x=exact match, u=uppercase, l=lowercase ].....	139
COB_LOAD_CASE=xul [where x= exact match, u=uppercase, l=lowercase ] .....	139
COB_CONSOLE_CP=<code page>.....	140
COB_CURRENT_DATE .....	140
COB_DEBUG_ALLUSER=1 .....	140
COB_DEBUG_ID=<debug-id>.....	141
COB_DEBUG_MODULES=<program-id1>:<program-id2>.....	141
COB_DEBUG_STARTUP_FILE=<filename> .....	142
COB_DEBUG_TMP=<directory>.....	143
COB_DISPLAY_PRINTER=<filename> .....	143

COB_DUMP=<filename> .....	143
COB_ERROR_FILE=<filename> .....	144
COB_EXTFH=<EXTFH Entry> .....	144
COB_EXTFH_FLAT=<EXTFH Entry> .....	144
COB_EXTFH_INDEXED=<EXTFH Entry>.....	145
COB_EXTFH_LIB=[ list of shared libraries ] .....	145
COB_FILE_CASE=[UPPER LOWER] .....	145
COB_FILEMAP_CASE=[UPPER/LOWER] .....	145
COB_FILE_PATH=[PATH] .....	145
COB_FILE_RELATIVE_MF=Y .....	146
COB_FILE_TRACE=[Y/N] .....	146
COB_FULL_CANCEL=[Y/N] .....	146
COB_KEY_DUP_ALWAYS_22=[Y/N] .....	146
COB_LIBRARY_PATH =[PATH] .....	147
COB_LOAD_CASE=[UPPER/LOWER] .....	147
COB_LOAD_PRIORITY=[Y/N] .....	147
COB_LS_DOS=[Y/N] .....	147
COB_LS_FIXED=[Y/N] .....	148
COB_LS_NULLS=[Y/N] .....	148
COB_NO_DOT_DAT .....	148
COB_NO_SIGNAL=[Y/N] .....	148
COB_PAD_BUG=[0/1] .....	148
COB_PRE_LOAD=[list of modules].....	149
COB_PROFILING_DIR .....	149
COB_PROFILING_EACH_MODULE .....	149
COB_RTL_CP=<codepage> .....	150
COB_RUNTIME_CHECK_TRACE=[Y/N/Module list separated by ; or : (Windows)].....	150
COB_SCREEN_DISABLE_REFORMAT=[Y/N] .....	151
COB_SCREEN_ESC=[Y/N] .....	151
COB_SCREEN_EXCEPTIONS=[Y/N] .....	151
COB_SCREEN_INPUT_BOLDDED=[Y/N].....	152
COB_SCREEN_INPUT_FILLER=[char] .....	152
COB_SCREEN_INPUT_INSERT_TOGGLE=[Y/N] .....	152
COB_SCREEN_INPUT_REVERSED=[Y/N] .....	152
COB_SCREEN_INPUT_UNDERLINED=[Y/N] .....	152
COB_SCREEN_RAW_KEYS=[Y/N] .....	152
COB_SCREEN_UPDATE_FIRST_KEY_ERASE=[Y/N] .....	153
COB_STDUNIX=[Y/N] .....	153
COB_SWITCH_0... COB_SWITCH_16.....	154
COB_SYNC=[Y/N] .....	154
COB_VAR_REC_PAD=[Y/N].....	154
COBLPFORM="n:n:n: : : : : :n" .....	154
COBOLITDIR=<directory> .....	155
TMPDIR or TMP=<directory> .....	155
<b>File Status Codes.....</b>	<b>156</b>
<b>Runtime Error Codes .....</b>	<b>159</b>
<b>Data Memory allocation.....</b>	<b>161</b>
BINARY, COMPUTATIONAL .....	161
bin-opt-strict:[yes/no].....	161
BINARY-CHAR, BINARY-CHAR SIGNED .....	161
BINARY-CHAR UNSIGNED.....	161
BINARY-C-LONG, BINARY-C-LONG SIGNED .....	161
BINARY-C-LONG UNSIGNED.....	162
BINARY-DOUBLE, BINARY-DOUBLE SIGNED.....	162

BINARY-DOUBLE UNSIGNED.....	162
BINARY-LONG, BINARY-LONG SIGNED, SIGNED-LONG, SIGNED-INT.....	162
BINARY-SHORT BINARY-SHORT SIGNED SIGNED-SHORT .....	162
BINARY-SHORT UNSIGNED UNSIGNED-SHORT .....	164
COMPUTATIONAL-1 .....	164
COMPUTATIONAL-2.....	164
COMPUTATIONAL-3 PACKED-DECIMAL.....	164
COMPUTATIONAL-4.....	164
bin-opt-strict:[yes/no].....	165
COMPUTATIONAL-5 .....	165
COMPUTATIONAL-6.....	165
COMPUTATIONAL-X.....	165
bin-opt-strict:[yes/no].....	166
DISPLAY .....	166
INDEX .....	166
POINTER PROGRAM-POINTER .....	166
<b>Using EXTFH-Compliant Indexed File Systems.....</b>	<b>167</b>
COBOL-IT includes EXTFH Libraries.....	167
VBISAM .....	167
-fvbisam .....	167
vbisam: [yes/no].....	168
BerkeleyDB .....	168
-fbdb .....	168
bdb: [yes/no] .....	168
D-ISAM .....	168
-fdisam .....	168
disam: [yes/no].....	168
dcheck .....	168
C-Tree ACE .....	169
Documentation .....	169
Installation.....	170
Compiling.....	171
Running.....	171
Start/Stop Engine .....	171
Data file location .....	171
<b>Reserved Word List.....</b>	<b>173</b>
<b>Intrinsic Function List.....</b>	<b>179</b>
<b>COBOL-IT® Library Routines.....</b>	<b>185</b>
C\$CALLERNAME.....	187
C\$CHDIR .....	188
C\$COPY .....	189
C\$DEBUG .....	190
C\$DELETE.....	201
C\$FILEINFO .....	202
C\$JUSTIFY .....	204
C\$MAKEDIR .....	204
C\$NARG .....	206
C\$PARAMSIZE .....	207
C\$PID .....	208
C\$\$SLEEP.....	209
C\$TOLOWER .....	210
C\$TOUPPER.....	211
CBL_ALLOC_DYN_MEM .....	211











































































































































































































































































































































































































































































































































































































































Example:

```
(cobcdb)
info profiling
```

## info sources

displays source files associated with objects loaded in memory

Example:

```
(cobcdb)
info sources
Source files
C:/COBOL/COBOLIT/samples/hello.cbl
(cobcdb)
```

## info target

displays the pid of the currently running process.

Example:

```
(cobcdb)
info target
Child PID 19012
(cobcdb)
```

## kill

kills the current process.

Example:

```
(cobcdb)
kill
-event-program-exited (cobcdb)#0 hello () at
C:/COBOL/COBOLIT/samples/hello.cbl!
10
```

## list

Requires that the source file be accessible. The list debugger command allows you to expand the source you can see inside the console debugger as you execute your debugger commands:

```
(cobcdb)
s
```

```
-event-step
(cobcdb)
-event-end-stepping-range #0 CUSTOMER0 () at /opt/cobol-it-
64/samples/customer0.cbl!99
.0000099>          CALL "C$PID" USING PID.
```

**list**

```
.0000094.
.0000095.
*****
.0000096.      PROCEDURE DIVISION.
.0000097.
.0000098.      Main Section.
.0000099>          CALL "C$PID" USING PID.
.0000100.      DISPLAY "PID = " PID.
.0000101.      *   CALL "C$DEBUG"
.0000102.      ACCEPT W-SYS-DATE FROM DATE.
.0000103.      MOVE W-SYS-YY          TO CURR-YY.
.0000104.      MOVE W-SYS-MM          TO CURR-MM.
(cobcdb)
```

## next

causes execution to pass to the next statement- jumping over a CALL or PERFORM statement before breaking, unless the CALL'ed paragraph or PERFORM statement contains a breakpoint. An event-next command is issued. This is interrupted when an -event-end-stepping-range event takes place. The next command can be abbreviated as "n".

Example :

```
(cobcdb)
```

**next**

```
-event-next
```

```
-event-end-stepping-range (cobcdb)#0 hello () at
C:/COBOL/COBOLIT/samples/hello.cbl!17
```

## print <variable-name>

displays the value of the variable in human readable format.

Example:

**print message-line**

```
$1 = @hello.WORKING-STORAGE.message-line [11]"XXXXXXXXXX"
(cobcdb)
```

The information returned is in the format:

```
$1=@module-name.section-name.variable-name[size]"[string]"
```

Where:

module-name is the program-id of the module being executed.  
section-name is the section containing the variable being displayed.  
size is the size, in bytes of the variable.  
string is the contents of the variable in human-readable format.

## printh <variable-name>

displays the value of the variable in hexadecimal format.

Example:

```
printh message-line
```

```
$1 = @hello.WORKING-STORAGE.message-line [22]"58585858585858585858"  
(cobcdb)
```

The information returned is in the format:

```
$1=@module-name.section-name.variable-name[size]"[string]"
```

Where:

module-name is the program-id of the module being executed.  
section-name is the section containing the variable being displayed.  
size is the size, in bytes of the variable.  
string is the contents of the variable in hexadecimal format.

## quit

causes an exit from the debugger.

Example:

```
(cobcdb)
```

```
quit
```

```
C:\COBOL\COBOLIT\samples>
```

## replace

allows the user to replace the prefix of the pathname to the source file associated with the compiled object.

The replace command may be used with the following parameters:

- |                                    |   |
|------------------------------------|---|
| >replace <oldprefix> : <newprefix> | Affects the output of the list command.<br>Example: replace /dirA : /dirB |
| >replace <no arguments>            | Resets the list command.  |
| >replace ?                         | Produces a list of active replacements                                    |

**>replace <oldprefix> : <newprefix>**

The replace <oldprefix>:<newprefix> command affects the output of the >list command. Other commands such as >info sources or >break still use the original pathname as it is stored in the binary code of the program.

Examples: >replace /dirA:/dirB

In this example, /dirA and /dirB are prefixes of a pathname to the source file that is associated with the compiled object. In this example, the >list command would replace the location of source code associated with the compiled object from /dirA/dev/sources to /dirB/dev/sources.

Where two or more replace commands are executed, the commands are stacked internally. Consider this case, where three replace commands are executed:

```
replace /dirA: /dirB
replace /dirC : /dirD
replace /dirE : /dirF
```

This provides a list of three possible replacements. Only the first matching replacement will be executed.

**>replace ?**

The replace ? command produces a list of active replacements.

Following our previous example, we would see:

```
/dirA : /dirB
/dirC : /dirD
/dirE : /dirF
```

**>replace <no arguments>**

The replace <no arguments> command re-sets the list command, removing all replacements established in the debugging session.

**set**

allows the user to set a <set parameter> to a different value.

The set command requires a <parameter>.

Parameters for the set command are:

*prompt*<prompt-string>      Sets the debugger prompt to <prompt-string>

`var <variable-name> <variable-value>` Sets the value of <variable-name>  
`varh <variable-name> <variable-value>` Sets the value of <variable-name> in hex notation

### set prompt <prompt string>

sets the COBOL-IT Debugger prompt. The default setting for the COBOL-IT Debugger

prompt is (cobcdb).

Example :

(cobcdb)

event:13556

-event-end-stepping-range #0 hello () at C:/COBOL/COBOLIT/samples/hello.cb!9

**set prompt >>>**

>>>

### set var <variable-name> <variable-value>

sets variable content for variable-name to variable-value. Values are converted to the appropriate type. A number stored in a PIC 999 field will be converted before storing.

Example :

(cobcdb)

**set var message-line "hello hello"**

\$1 = @hello.WORKING-STORAGE.message-line [11]"hello hello"

(cobcdb)

### set varh <variable-name> <variable-value-hex>

sets variable content for variable-name to variable-value-hex.

<variable-value-hex> must be a valid hexadecimal string. Note that in a valid hexadecimal string, a single character space is recorded with two characters, so the total string length of <variable-value-hex> must be exactly two times the length of <variable-name>.

(cobcdb)

**set varh ws-dummy 41**

\$1 = @hello.WORKING-STORAGE.ws-dummy [1]"A"

(cobcdb)

### step

causes execution of the program to execute a single step, and then break. An event-step command is issued. This is interrupted when an –event-end-stepping-range event takes place. The step command can be abbreviated as “s”.

Example:

```
(cobcdb)
```

```
step
```

```
-event-step
```

```
(cobcdb)-event-end-stepping-range #0 hello () at
```

```
C:/COBOL/COBOLIT/samples/hello.cbl!14
```

## stop

causes execution to stop (break) at the next statement

## up -[n]

changes the current frame. When you have several levels of CALLs, the **info** functions relate to the current module. In a CALL'ed subprogram, **up -[n]** can be used to change the frame back to a previous CALL'ing module. **Info locals** can then be viewed for that calling module.

In the example below, the **bt** command shows 3 frames, with frame 0 being the current frame in a called sub-program, and the **info locals** command showing the state of the variables in the subprogram. **up -1** sets the frame to the calling program, so that **info locals** can be viewed for the calling program.

```
bt
```

```
#0 subpgm () at C:/COBOL/COBOLIT/samples/subpgm.cbl!7
```

```
#1 hello () at C:/COBOL/COBOLIT/samples/hello.cbl!25
```

```
#2 hello () at C:/COBOL/COBOLIT/samples/hello.cbl!17
```

```
(cobcdb)
```

```
info locals
```

```
@subpgm.WORKING-STORAGE
```

```
  @subpgm.WORKING-STORAGE.RETURN-CODE = [10]" +000000000"
```

```
  @subpgm.WORKING-STORAGE.TALLY = [10]" +000000000"
```

```
  @subpgm.WORKING-STORAGE.SORT-RETURN = [10]" +000000000"
```

```
  @subpgm.WORKING-STORAGE.NUMBER-OF-CALL-PARAMETERS =
```

```
[10]" +000000000"
```

```
  @subpgm.WORKING-STORAGE.COB-CRT-STATUS = [4]" "
```

```
(cobcdb)
```

```
up -1
```

```
#1 hello () at C:/COBOL/COBOLIT/samples/hello.cbl!25
```

```
(cobcdb)
```

```
info locals
```

```
@hello.WORKING-STORAGE
```

```
  @hello.WORKING-STORAGE.RETURN-CODE = [10]" +000000000"
```

```
  @hello.WORKING-STORAGE.TALLY = [10]" +000000000"
```

```
  @hello.WORKING-STORAGE.SORT-RETURN = [10]" +000000000"
```

```
@hello.WORKING-STORAGE.NUMBER-OF-CALL-PARAMETERS =  
[10]"000000000"  
@hello.WORKING-STORAGE.message-line = [11]"XXXXXXXXXX"  
@hello.WORKING-STORAGE.ws-dummy = [1]" "  
@hello.WORKING-STORAGE.ctr = [6]"000000"  
@hello.WORKING-STORAGE.COB-CRT-STATUS = [4]"0000"  
(cobcdb)
```

## version

returns the version of the cobcdb/COBOL-IT runtime.

Example:

```
(cobcdb)
```

```
version
```

```
~"COBOL-IT cobcdb 3.6.4\n"
```

```
^done
```

```
(cobcdb)
```

## Debugger Events

### -event-breakpoint-hit

Returned when a breakpoint is hit.

### -event-continue

Returned by the continue command. Terminated by `-event-breakpoint-hit`.

### -event-contreturn

Returned by the contreturn command. Terminated by `-event-end-stepping-range`.

### -event-end-stepping-range

Returned when one of the debugger step commands (`step`, `next`, `contreturn`) reaches the end of its stepping range.

### -event-next

Returned by the next command. Terminated by `-event-end-stepping-range`.

### -event-program-exited

Returned by the kill command.



## -event-step

Returned by the step command. Terminated by `-event-end-stepping-range`.

## Our Sample Programs

For the purposes of this documentation, we are using a very short `hello.cbl` program as a reference.

(The program contains an `ACCEPT FROM COMMAND-LINE` statement, to illustrate this functionality in `cobcdb`.)

To compile: `>cobc -g hello.cbl`

`>cobc -g subpgm.cbl`

To run: `>cobcdb hello (or)`

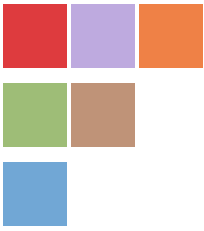
To run with parameters: `>cobcd hello hello-world`

### hello.cbl

```
000001 identification division.
000002 program-id. hello.
000003 environment division.
000004 data division.
000005 working-storage section.
000006 77 message-line pic x(11) value spaces.
000007 77 ws-dummy pic x value spaces.
000008 77 ctr pic 9(6) value 0.
000009 procedure division.
000010 main.
000011 accept message-line from command-line.
000012 if message-line not = spaces
000013 display message-line line 10 col 10
000014 else
000015 display "hello world" line 10 col 10
000016 end-if.
000017 perform para-1.
000018 display "returned from para-1" line 14 col 10.
000019 display "next line" line 16 col 10.
000020 accept ws-dummy line 16 col 30.
000021 stop run.
000022 para-1.
000023 move all "X" to message-line.
000024 display "in para-1" line 12 col 10.
000025 call "subpgm".
```

**subpgm.cbl**

```
000001 identification division.  
000002 program-id. subpgm.  
000003 environment division.  
000004 data division.  
000005 working-storage section.  
000006 procedure division.  
000007 main.  
000008     display "In Subpgm" line 20 col 10.  
000009     goback.
```



[www.cobol-it.com](http://www.cobol-it.com)

June, 2020

