Elastic COBOL Language Reference Manual Appendix

COBOL-85 Standard ANSI X3.23B

REVISION: AUGUST 2012

Preface

Trademarks

- IBM is a registered trademark of International Business Machines Inc.
- Oracle and Java are registered trademarks of Oracle and/or its affiliates.
- UNIX is a registered trademark licensed exclusively to X/Open Company Limited.
- Linux is a registered trademark of Linus Torvalds.
- Windows is a registered trademark of Microsoft Corporation.
- Eclipse is a trademark of the Eclipse Foundation Inc.
- Other names may be trademarks of their respective owners.

COBOL Language Reference

This describes the base COBOL language supported by this system: this COBOL language is based on the ANSI COBOL standards X3.23-1985, X3.23a-1989 and X3.23b-1993, and is supported by a number of COBOL systems. In addition, support has been added for some of the features from ISO/IEC 1989:2002, Programming language COBOL.

COBOL is an industry language and is not the property of any company or group of companies, or of any organization or group of organizations.

No warranty, expressed or implied, is made by any contributor or by the CODASYL Programming Language Committee as to the accuracy and functioning of the programming system and language. Moreover, no responsibility is assumed by any contributor, or by the committee, in connection herewith.

The authors and copyright holders of the copyrighted material used herein:

> FLOW-MATIC Programming for the Univac I and II, Data Automation Systems copyrighted 1958,1959, by Sperry Rand Corporation; IBM Commercial Translator Form No. F28-8013, copyrighted 1959 by IBM; FACT, DSI27A5260-2760, copyrighted 1960 by Minneapolis-Honeywell

have specifically authorized the use of this material in whole or in part, in the COBOL specifications. Such authorization extends to the reproduction and use of COBOL specifications in programming manuals or similar publications.

Contents

Appendix A - COBOL Reserved Words

ACCEPT
ACCESS
ADD
ADVANCING
AFTER
ALL
ALPHABET
ALPHABETIC
ALPHABETIC-LOWER
ALPHABETIC-UPPER
ALPHANUMERIC
ALPHANUMERIC-EDITED
ALSO
ALTER
ALTERNATE
AND
ANY
ARE
AREA
AREAS
ASCENDING
ASSIGN
AT
AUTHOR

BEFORE
BINARY
BLANK
BLOCK
BOTTOM
BY

CALL
CANCEL
CD
CF
CH
CHARACTER
CHARACTERS
CLASS
CLOCK-UNITS
CLOSE
COBOL
CODE
CODE-SET
COLLATING
COLUMN
COMMA
COMMON
COMMUNICATION
CONFIGURATION
CONTAINS
CONTENT
CONTINUE
CONTROL
CONTROLS
CONVERTING
COPY
CORR
CORRESPONDING
COUNT
CURRENCY

DATA
DATE
DATE-COMPILED
DATE-WRITTEN
DAY
DAY-OF-WEEK
DE
DEBUG-CONTENTS
DEBUG-ITEM
DEBUG-LINE
DEBUG-NAME
DEBUG-SUB-1
DEBUG-SUB-2
DEBUG-SUB-3
DEBUGGING
DECIMAL-POINT
DECLARATIVES
DELETE
DELIMITED
DELIMITER
DEPENDING
DESCENDING
DESTINATION
DETAIL
DISABLE
DISPLAY
DIVIDE
DIVISION
DOWN
DUPLICATES
DYNAMIC

EGI
ELSE
EMI
ENABLE
END
END-DELETE
END-DIVIDE
END-EVALUATE
END-IF
END-MULTIPLY
END-OF-PAGE
END-PERFORM
END-READ
END-RECEIVE
END-RETURN
END-REWRITE
END-SEARCH
END-START
END-STRING
END-SUBTRACT
END-UNSTRING
END-WRITE
ENTER
ENVIRONMENT
EOP
EQUAL
ERROR
ESI
EVALUATE
EVERY
EXCEPTION
EXIT
EXTEND
EXTERNAL

FALSE
FD
FILE
FILE-CONTROL
FILLER
FINAL
FIRST
FOOTING
FOR
FROM

GENERATE
GIVING
GLOBAL
GO
GREATER
GROUP

HEADING
HIGH-VALUE
I-O-CONTROL
IDENTIFICATION
IF
IN
INDEX
INDEXED
INDICATE
INITIAL
INITIALIZE
INITIATE
INPUT
INPUT-OUTPUT
INSPECT
INSTALLATION
INTO
INVALID
IS

JUST
JUSTIFIED

KEY

LABEL
LAST
LEADING
LEFT
LENGTH
LESS
LIMIT
LIMITS
LINAGE
LINAGE-COUNTER
LINE
LINE-COUNTER
LINES
LINKAGE
LOCK
LOW-VALUE
LOW-VALUES

MEMORY
MERGE
MESSAGE
MODE
MODULES
MOVE
MULTIPLE
MULTIPLY

COMP
COMPUTATIONAL
COMPUTE
NO
NOT
NUMBER
NUMERIC
NUMERIC-EDITED

OBJECT-COMPUTER
OCCURS
OF
OFF
OMITTED
ON
OPEN
OPTIONAL
OR
ORDER
ORGANIZATION
OTHER
OUTPUT
OVERFLOW

PACKED-DECIMAL
PADDING
PAGE
PAGE-COUNTER
PERFORM
PF
PH
PIC
PICTURE
PLUS
POINTER
POSITION
POSITIVE
PRINTING
PROCEDURE
PROCEDURES
PROCEED
PROGRAM
PROGRAM-ID
PURGE

QUEUE

END-ADD
END-CALL
END-COMPUTE
QUOTE
QUOTES

RANDOM
RD
READ
RECEIVE
RECORD
RECORDS
REDEFINES
REEL
REFERENCE
REFERENCES
RELATIVE
RELEASE
REMAINDER
REMOVAL
RENAMES
REPLACE
REPLACING
REPORT
REPORTING
REPORTS
RERUN
RESERVE
RESET
RETURN
REVERSED
REWIND
REWRITE
RF
RH
RIGHT
ROUNDED
RUN

SAME
SD
SEARCH
SECTION
SECURITY
SEGMENT
SEGMENT-LIMIT

HIGH-VALUES

I-O
SELECT
SEND
SENTENCE
SEPARATE
SEQUENCE
SEQUENTIAL
SET
SIGN
SIZE
SORT
SORT-MERGE
SOURCE
SOURCE-COMPUTER
SPACE
SPACES
SPECIAL-NAMES
STANDARD
STANDARD-1
STANDARD-2
START
STATUS
STOP
STRING
SUB-QUEUE-1
SUB-QUEUE-2
SUB-QUEUE-3
SUBTRACT
SUM
SUPRESS
SYMBOLIC
SYNC
SYNCHRONIZED

TABLE
TALLYING
TAPE
TERMINAL
TERMINATE
TEST
TEXT
THAN
THEN
THROUGH

NATIVE
NEGATIVE
NEXT
THRU
TIME
TIMES
TO
TOP
TRAILING
TRUE
TYPE

UNIT
UNSTRING
UNTIL
UP
UPON
USAGE
USE
USING

VALUE
VALUES
VARYING

WHEN
WITH
WORDS
WORKING-STORAGE
WRITE

ZERO
ZEROES
ZEROS

+
-
*
/
**
>
<
=
>=
<=

Appendix B – Elastic COBOL Reserved Words

$INCLUDE	**	*>	>>
>>D	ACCEPT	ACCESS	ADD
ADDRESS	ADVANCE	ADVANCING	AFTER
ALIGNED	ALL	ALLOCATE	ALLOWING
ALPHABET	ALPHABETIC	ALPHABETIC-HIGHER	ALPHABETIC-LOWER
ALPHABETIC-UPPER	ALPHANUMERIC	ALPHANUMERIC-EDITED	ALSO
ALTER	ALTERNATE	ALTERNATIVE	AND
ANY	APPLY	ARE	AREA
AREAS	ARGUMENT-NUMBER	ARGUMENT-VALUE	AS
ASCENDING	ASCII	ASSIGN	AT
ATTRIBUTE	AUTHOR	AUTO	AUTO-SKIP
AUTOMATIC	AUTOTERMINATE	B-AND	B-NOT
B-OR	B-XOR	BACKGROUND-COLOR	BACKGROUND-COLOUR
BACKWARD	BASED-STORAGE	BEEP	BEFORE
BELL	BINARY	BINARY-BYTE	BINARY-CHAR
BINARY-DOUBLE	BINARY-LONG	BINARY-SHORT	BIT
BLANK	BLINK	BLINKING	BLOCK
BOTTOM	BRIGHT	BUILD	BY
C01	C02	C03	C04
C05	C06	C07	C08
C09	C10	C11	C12
CALL	CANCEL	CAST	CASTING
CD	CENTURY-DATE	CENTURY-DAY	CF
CH	CHAIN	CHAINING	CHANGED
CHARACTER	CHARACTERS	CLASS	CLASS-ID
CLOSE	COBOL	CODE	CODE-SET
COL	COLLATING	COLUMN	COMMA
COMMAND-LINE	COMMIT	COMMON	COMMUNICATION
COMP	COMP-1	COMP-1-MVS	COMP-1-REV
COMP-2	COMP-2-MVS	COMP-2-REV	COMP-3
COMP-4	COMP-5	COMP-6	COMP-D
COMP-N	COMP-S	COMP-X	COMPONENT
COMPUTATIONAL	COMPUTATIONAL-1	COMPUTATIONAL-1-MVS	COMPUTATIONAL-1-REV
COMPUTATIONAL-2	COMPUTATIONAL-2-MVS	COMPUTATIONAL-2-REV	COMPUTATIONAL-3
COMPUTATIONAL-4	COMPUTATIONAL-5	COMPUTATIONAL-6	COMPUTATIONAL-D
COMPUTATIONAL-N	COMPUTATIONAL-S	COMPUTATIONAL-X	COMPUTE
CONDITION-CODE	CONDITIONALLY	CONFIGURATION	CONSOLE
CONSTANT	CONTAINS	CONTENT	CONTINUE
CONTROL	CONTROLS	CONVERSION	CONVERT
CONVERTING	COPY	CORR	CORRESPONDING
COUNT	CRT	CSP	CURRENCY
CURRENT-THREAD	CURSOR	CYCLE	DATA
DATE	DATE-COMPILED	DATE-WRITTEN	DAY

DAY-OF-WEEK	DE	DEBUGGING	DECIMAL-POINT
DECLARATIVES	DELETE	DELIMITED	DELIMITER
DEPENDING	DESCENDING	DESTINATION	DETAIL
DIM	DISABLE	DISC	DISK
DISPLAY	DISPLAY-1	DISPLAY-WS	DIVIDE
DIVISION	DOUBLE	DOWN	DUPLICATES
DYNAMIC	EBCDIC	EGI	ELSE
EMI	EMPTY-CHECK	ENABLE	END
END-ACCEPT	END-ACCEPT	END-ADD	END-BUILD
END-CALL	END-CLASS	END-COMPUTE	END-DECLARATIVES
END-DELETE	END-DISPLAY	END-DISPLAY	END-DIVIDE
END-EVALUATE	END-EVENT	END-EXEC	END-HIDE
END-IF	END-INVOKE	END-METHOD	END-MULTIPLY
END-OF-PAGE	END-PERFORM	END-PROGRAM	END-READ
END-RECEIVE	END-RETURN	END-REWRITE	END-SEARCH
END-SHOW	END-START	END-STRING	END-SUBTRACT
END-THREAD	END-UNSTRING	END-WRITE	ENTER
ENTRY	ENVIRONMENT	ENVIRONMENT-NAME	ENVIRONMENT-VALUE
EOP	EQUAL	EQUALS	ERASE
ERROR	ESCAPE	ESI	EVALUATE
EVENT	EXAMINE	EXCEPTION	EXCLUSIVE
EXEC	EXECUTE	EXIT	EXTEND
EXTERNAL	EXTERNAL-FORM	FAC	FALSE
FD	FIGURATIVE-CONSTANTS	FILE	FILE-CONTROL
FILE-ID	FILE-PREFIX	FILE-SECTION	FILLER
FINAL	FIRST	FLOAT	FLOAT-LONG
FLOAT-SHORT	FOOTING	FOR	FOREGROUND-COLOR
FOREGROUND-COLOUR	FREE	FROM	FULL
FUNCTION	GENERATE	GIVING	GLOBAL
GO	GOBACK	GREATER	GREATER
GRIDLINE	GROUP	HANDLE	HEADING
HIDE	HIGH	HIGH-VALUE	HIGH-VALUES
HIGHLIGHT	HOLD	I-O	I-O-CONTROL
ID	IDENTIFICATION	IDENTIFIED	IF
IGNORE	IGNORING	IN	INDEX
INDEXED	INDICATE	INHERITS	INITIAL
INITIALIZE	INITIALIZED	INITIATE	INPUT
INPUT-OUTPUT	INSPECT	INSTALLATION	INTO
INTRINSIC	INVALID	INVOKE	IO
IS	JAVA-PARAMETER	JAVA-RETURN	JUST
JUSTIFIED	KEPT	KEY	KEYBOARD
LABEL	LAST	LEADING	LEFT
LEFT-JUSTIFY	LEFTLINE	LENGTH	LENGTH-CHECK
LESS	LIMIT	LIMITS	LINAGE
LINAGE-COUNTER	LINE	LINE-COUNTER	LINES
LINKAGE	LOCAL-STORAGE	LOCK	LOCK-HOLDING
LOW	LOW-VALUE	LOW-VALUES	LOWLIGHT
MANUAL	MASS-UPDATE	MEMORY	MERGE
MESSAGE	METHOD	METHOD-ID	MODE
MODULES	MOVE	MULTIPLE	MULTIPLY
NAME	NAMED	NATIONAL	NATIVE
NEGATIVE	NEXT	NO	NO-ECHO

NODISPLAY	NORMAL	NOT	NOTE
NULL	NULLS	NUMBER	NUMERIC
NUMERIC-EDITED	OBJECT	OBJECT-COMPUTER	OCCURS
OF	OFF	OMITTED	ON
ONLY	OPEN	OPTIONAL	OR
ORDER	ORGANIZATION	OTHER	OTHERS
OTHERWISE	OUTPUT	OVERFLOW	OVERLINE
PACKED-DECIMAL	PADDING	PAGE	PAGE-COUNTER
PARAGRAPH	PASSWORD	PERFORM	PERFORMS
PF	PFKEY	PFKEYS	PH
PIC	PICTURE	PLUS	POINTER
POSITION	POSITIVE	PREVIOUS	PRINT
PRINT-CONTROL	PRINTER	PRINTER-1	PRINTING
PRIORITY	PROCEDURE	PROCEDURE-POINTER	PROCEDURES
PROCEED	PROGRAM	PROGRAM-ID	PROGRAM-POINTER
PROMPT	PROPERTY	PURGE	QUEUE
QUOTE	QUOTES	RAISE	RAISING
RANDOM	RD	READ	READERS
RECEIVE	RECORD	RECORD-KEY	RECORDING
RECORDS	RECURSIVE	REDEFINES	REEL
REFERENCE	REFERENCES	RELATIVE	RELEASE
REMAINDER	REMARKS	REMOVAL	RENAMES
REPLACE	REPLACING	REPORT	REPORTING
REPORTS	REPOSITORY	REQUIRED	RERUN
RESERVE	RESET	RESIDENT	RETRY
RETURN	RETURN-CODE	RETURNING	REVERSE
REVERSE-VIDEO	REVERSED	REWIND	REWRITE
RF	RH	RIGHT	RIGHT-JUSTIFY
ROLLBACK	ROUNDED	ROW	RUN
S01	S02	S03	S04
S05	SAME	SCREEN	SD
SEARCH	SECONDS	SECTION	SECURE
SECURITY	SEGMENT	SEGMENT-LIMIT	SELECT
SELF	SEND	SENTENCE	SEPARATE
SEQUENCE	SEQUENTIAL	SERVLET-IN	SERVLET-OUT
SERVLETIN	SERVLETOUT	SET	SHARED
SHARING	SHIFT-IN	SHIFT-OUT	SHOW
SIGN	SIGNED-INT	SIGNED-LONG	SIGNED-SHORT
SIZE	SORT	SORT-MERGE	SOURCE
SOURCE-COMPUTER	SPACE	SPACE-FILL	SPACES
SPECIAL-NAMES	STANDARD	STANDARD-1	STANDARD-2
START	STATUS	STOP	STRING
SUB-QUEUE-1	SUB-QUEUE-2	SUB-QUEUE-3	SUBTRACT
SUM	SUPER	SUPPRESS	SWITCH
SWITCH-1	SWITCH-10	SWITCH-11	SWITCH-12
SWITCH-13	SWITCH-14	SWITCH-15	SWITCH-16
SWITCH-17	SWITCH-18	SWITCH-19	SWITCH-2
SWITCH-20	SWITCH-21	SWITCH-22	SWITCH-23
SWITCH-24	SWITCH-25	SWITCH-26	SWITCH-3
SWITCH-4	SWITCH-5	SWITCH-6	SWITCH-7
SWITCH-8	SWITCH-9	SYMBOLIC	SYNC
SYNCHRONIZED	SYSERR	SYSIN	SYSIPT
SYSLIST	SYSLST	SYSOUT	SYSOUT-FLUSH

SYSPCH	SYSPUNCH	SYSTEM-INFO	SYSTEM-SWITCH-1
SYSTEM-SWITCH-10	SYSTEM-SWITCH-2	SYSTEM-SWITCH-3	SYSTEM-SWITCH-4
SYSTEM-SWITCH-5	SYSTEM-SWITCH-6	SYSTEM-SWITCH-7	SYSTEM-SWITCH-8
SYSTEM-SWITCH-9	TAB	TABLE	TALLY
TALLYING	TAPE	TERMINAL	TERMINAL-INFO
TERMINATE	TEST	TEXT	THAN
THEN	THREAD	THROUGH	THRU
TIME	TIME-OF-DAY	TIMEOUT	TIMES
TO	TOP	TRAILING	TRANSACTION
TRANSFORM	TRUE	TYPE	TYPEDEF
UN-EXCLUSIVE	UNDERLINE	UNDERLINED	UNIT
UNLOCK	UNSIGNED-INT	UNSIGNED-LONG	UNSIGNED-SHORT
UNSTRING	UNTIL	UP	UPDATE
UPDATERS	UPON	USAGE	USE
USER	USING	VALUE	VALUES
VARYING	WAIT	WHEN	WITH
WORDS	WORKING-STORAGE	WRITE	WRITERS
YEAR	YYYYDDD	YYYYMMDD	ZERO
ZERO-FILL	ZEROES	ZEROS	

Appendix C - Definitions

The definitions presented are generally applicable to COBOL and include certain definitions not directly related to Elastic COBOL.

A

Term	Definition
abbreviated combined relation condition	The combined condition that results from the explicit omission of a common subject or a common subject and common relational operator in a consecutive sequence of relation conditions. Elastic COBOL limits abbreviated combined relation conditions to eight (8) elements.
access mode	The manner in which records are to be operated upon within a file.
activated run time entity	A function, method, or program placed into the active state by a statement.
activating statement	A statement that causes the execution of a function, method, or program.
activating run time entity	A function, method, or program that contains the activating statement.
active state	The state of a program that has been called but has not yet returned to the calling program.
actual decimal point	The physical representation, using the decimal point characters period (.) or comma (,), of the decimal point position in a data item.
address-identifier	An identifier that references the address of a data item or pointer.
alphabet-name	A user-defined word in the SPECIAL-NAMES paragraph of the environment division that assigns a name to a specific character set and collating sequence.
alphabetic character	A basic letter or a space character.
alphanumeric character	Any character in an alphanumeric character set.
alphanumeric character position	The amount of physical storage required to store, or presentation space required to print or display, a single character of an alphanumeric character set.
alphanumeric character set	Any character set that is used to represent data associated with COBOL's usage display.
alphanumeric collating sequence	A collating sequence associated with an alphanumeric character set.
alphanumeric function	A function whose value is composed of a string of one or more alphanumeric characters from an alphanumeric character set.
alphanumeric literal	A literal delimited by either (1) paired quotation symbols or (2) the opening delimiter T" or X" on the left and a matching quotation symbol on the right. The literal has the class and category alphanumeric and its content may be one or more characters from the computer's character set.
alternate record key	A key, other than the prime record key, whose contents identify a record within an indexed file.
Argument	An operand specified in the activating statement of a function,

Term	Definition
	method, or program that specifies the data to be passed upon activation of the run time entity.
arithmetic operation	Any of the basic mathematical processes of addition, subtraction, multiplication, division, exponentiation, unary plus, and unary minus.
ascending key	A key upon the values of which data is ordered starting with the lowest value of key up to the highest value of key in accordance with the rules for comparing data items.
assumed decimal point	A decimal point position that does not involve the existence of an actual character in a data item. The assumed decimal point has logical meaning with no physical representation.
at end condition	A condition caused: • During the execution of a READ statement for a sequentially accessed file, when no next logical record exists in the file, or when the number of significant digits in the relative record number is larger than the size of the relative key data item, or when an optional input file is not present. • During the execution of a RETURN statement, when no next logical record exists for the associated sort or merge file. • During the execution of a SEARCH statement, when the search operation terminates without satisfying the condition specified in any of the associated WHEN phrases.
automatic data	The data described in the local-storage section.
automatic data item	A data item that is described as part of an automatic data record.
automatic data record	A logical record that is described in the local-storage section.

B

Term	Definition
basic letters	The uppercase letters 'A' through 'Z' or the lowercase letters 'a' through 'z' in the COBOL character set.
binding	The process of linking a method invocation to a method implementation. Binding can be done at compile time if the compiler is able to determine the class of the object; in this case it is called static or early binding. Binding that is not done at compile time is called dynamic or late binding and is a form of method resolution that associates a method with an operation at run time, depending on the class of the receiving object.
bit	The smallest unit in a computer's storage structure capable of representing two distinct alternatives (zero and one).
bit position	The amount of physical storage required to store, or presentation space required to print or display, a single bit.
block; physical block	A physical unit of data that is normally composed of one or more logical records.
bottom margin	An empty area that follows the page body.

C

Term	Definition
call convention	The information required to interact successfully with a function,

Term	Definition
	method, or program. This includes items such as name case sensitivity, how arguments are passed, and stack management.
called program; subprogram	A program that receives control as the result of the execution of a CALL statement.
calling program	A program that transfers control to another program by execution of a CALL statement.
cd-name	A user-defined word that names an MCS interface area described in a communication description entry within the communication section of the data division. Not supported in Elastic COBOL.
Character	The basic unit of the language.
character addressable terminal	A terminal that consists of a terminal display screen and a terminal keyboard for entering data where the display screen represents a rectangular grid of character positions, for example 25 display lines of 80 character positions
character boundary	The leftmost bit of an addressing boundary in the storage of the computer.
character position	The amount of physical storage required to store, or presentation space required to print or display, one character -- either an alphanumeric character or a national character.
character-string	A sequence of contiguous characters that form a COBOL word, a literal, or a PICTURE character-string.
Class	The entity that defines common behavior and implementation for zero, one, or more objects. The objects that share the same implementation are considered to be objects of the same class.
class condition	The proposition, for which a truth value can be determined, that the content of an item is wholly alphabetic, wholly uppercase alphabetic, wholly lowercase alphabetic, is wholly numeric, is wholly boolean, or consists exclusively of those characters listed in the definition of a class-name.
class definition	The source unit that defines a class.
class-name (for truth value proposition)	A user-defined word defined in the SPECIAL-NAMES paragraph of the environment division that assigns a name to the proposition for which a truth value can be determined, that the content of a data item consists exclusively of those characters listed in the definition of the class-name.
class-name (for object orientation)	A user-defined word that identifies a class.
CLASSPATH	The path of directories and archive files through which the Java Virtual Machine attempts to resolve references to classes to their coded implementations. The classpath may be explicitly specified to the JVM or implicitly specified by using the CLASSPATH environment variable on many systems. See the Java documentation for further information.
clause	A clause is an ordered set of consecutive COBOL character-strings whose purpose is to specify an attribute of an entry.
COBOL character set	A character set used to express the syntax of a COBOL source program, except for the content of alphanumeric literals, national literals, and comments.
COBOL word	A character string that forms a compiler-directive word, a context-sensitive word, an exception-name, an intrinsic-function-

Term	Definition
	name, a reserved word, a system-name, or a user-defined word.
code-name	A system-name that identifies a character code set and/or a collating sequence.
coded character set	A set of unambiguous rules that establishes a character set and the relationship between the characters of the set and their coded representation. [ISO/IEC 10646-1]
collating sequence	The sequence in which the characters that are acceptable to a computer are ordered for purposes of sorting, merging, comparing, and for processing indexed files sequentially.
column	A character position within a print line or on the character addressable terminal.
combined condition	A condition that is the result of connecting two or more conditions with the 'AND' or the 'OR' logical operator.
combining character	A member of an identified subset of the coded character set of ISO/IEC 10646 intended for combination with the preceding non-combining graphic character, or with a sequence of combining characters preceded by a non-combining character. [ISO/IEC 10646-1] Not supported in Elastic COBOL.
comment indicator	The two contiguous COBOL characters '*>' or the character '|', which indicate a comment line or an in-line comment.
comment line	A line that serves only for documentation.
common program	A program that, despite being directly contained within another program, may be called from any program directly or indirectly contained in that other program.
communication description entry	An entry in the communication section of the data division that describes the interface between the message control system (MCS) and the COBOL program. Not supported in Elastic COBOL.
communication device	A mechanism (hardware or hardware/software) capable of sending data to a queue and/or receiving data from a queue. This mechanism may be a computer or a peripheral device. Not supported in Elastic COBOL.
communication section	The section of the data division that describes the interface areas between the message control system (MCS) and the program. Not supported in Elastic COBOL.
compilation group	A sequence of source units submitted for compilation together.
compiler directive	An instruction to the COBOL compiler to take specific action when compiling a source program.
compiler-directive word	A COBOL word that is used in the syntax of a compiler directive.
compile time	The time at which a COBOL source program is translated to an executable program.
compiler directing statement	A statement that instructs the compiler to take a specific action in the process of compiling a program.
complex condition	A condition in which one or more logical operators act upon one or more conditions.
composite sequence	A sequence of graphic characters consisting of a non-combining character followed by one or more combining characters. [ISO/IEC 10646-1].
computer-name	A system-name that identifies the computer upon which the program is to be compiled or run.
concatenation expression	An expression consisting of two operands separated by a concatenation

Term	Definition
	operator.
concatenation operator	The symbol '&'.
condition	A status of a program at execution time for which a truth value can be determined.
condition-name	A user-defined word that assigns a name to a subset of values that a conditional variable may assume; or a user-defined word assigned to a status of SWITCH-1 through SWITCH-26.
condition-name condition	The proposition, for which a truth value can be determined, that the value of a conditional variable is a member of the set of values attributed to a condition-name associated with the conditional variable.
conditional expression	A simple condition or a complex condition specified in an EVALUATE, IF, PERFORM, or SEARCH statement.
conditional phrase	A phrase that specifies the action to be taken upon determination of the truth value of a condition resulting from the execution of a conditional statement.
conditional statement	A statement for which the truth value of a specified condition is determined and used to control the subsequent action of the program.
conditional variable	A data item one or more values of which has a condition-name assigned to it.
configuration section	A section of the environment division that describes overall specifications of source and object programs.
conformance (for object orientation)	The property that allows an object with a given interface to be used where an object with a different interface is expected. Conformance ensures that any operation specified for the conformed interface is supported by the conforming interface. Not supported in Elastic COBOL.
context-dependent function key	A key on the keyboard that is defined as carrying out a particular function in a specific context.
context-sensitive words	COBOL words that are specified in formats but are reserved only in the context in which they are specified.
contiguous items	Items that are described by consecutive entries in the data division and that bear a definite hierarchical relationship to each other.
counter	A data item used for storing numbers or number representations in a manner that permits these numbers to be increased or decreased by the value of another number, or to be changed or reset to zero or to an arbitrary positive or negative value.
CRT status	A four-character conceptual entity whose value is set to indicate the status of a terminal input-output operation during the execution of an ACCEPT screen statement.
cultural element	An element of data for computer use that may vary dependent on language, geographical territory, or other cultural circumstances.
currency sign	The character '$' of the COBOL character set.
currency string	The set of characters placed into numeric edited data items as a result of editing operations when the item includes a currency symbol in its PICTURE character-string.
currency symbol	The character used in the PICTURE character-string to represent the presence of a currency string.
current record	In file processing, the record that is available in the record area associated with a file.
current volume pointer	A conceptual entity that points to the current volume of a

Term	Definition
	sequential file.
cursor	A visible indicator on a character addressable terminal screen that shows the position on the screen at which the next data character input at the keyboard will be displayed.
cursor control keys	Keys on the keyboard of a character addressable terminal that control the positioning of the cursor on the screen. Typically these include keys that move the cursor up, down, left or right

D

Term	Definition
data clause	A clause, appearing in a data description entry in the data division of a COBOL program, that provides information describing a particular attribute of a data item.
data description entry	A data division entry that specifies the characteristics of a data item.
data item	A unit of data (excluding literals) defined by the COBOL program or by the rules for function evaluation.
data keys	Keys on the keyboard of a character addressable terminal that represent individual printable data characters.
data-name	A user-defined word that names a data item described in a data description entry or a record described in a record-description entry.
data-pointer data item	A data item in which the address of a data item may be stored.
debugging indicator	The three contiguous COBOL characters '>>D' followed by a space, when specified as the first non-space characters of a free-form source line, that indicate a debugging line. The debugging indicator is treated as a comment indicator if debugging mode is not on.
debugging line	A source line that optionally may be compiled, depending on the setting of a debugging mode switch specified in the OBJECT-COMPUTER paragraph or compiler option –debug.
declarative statement	A statement beginning with the word USE that directs the compiler to generate code to take a specific action during the processing of other statements.
declaratives	A set of one or more special purpose sections, the first of which is preceded by the key word DECLARATIVES and the last of which is followed by the key words END DECLARATIVES.
de-edit	The logical removal of all editing characters from a numeric edited data item in order to determine that item's unedited numeric value.
delimited scope statement	Any statement that includes its explicit scope (END-x) terminator.
descending key	A key upon the values of which data is ordered starting with the highest value of key down to the lowest value of key, in accordance with the rules for comparing data items.
device-name	A system-name that identifies an input-output device.
digit position	The amount of physical storage required to store, or presentation space required to print or display, a single digit.
display attributes	Attributes associated with screen items that specify their rendition on the terminal when they are referenced in an ACCEPT or DISPLAY statement.
division	A collection of zero, one, or more sections or paragraphs, called

Term	Definition
	the division body, that are formed and combined in accordance with a specific set of rules. Each division consists of the division header and the related division body. There are four divisions in a COBOL program: identification, environment, data, and procedure.
division header	A combination of words, followed by a separator period, that indicates the beginning of a division.
dynamic access	An access mode in which specific logical records can be obtained from or placed into a mass storage file in a non-sequential manner and obtained from a file in a sequential manner during the scope of the same OPEN statement.
dynamic storage	Storage that is allocated during run time on request with an ALLOCATE statement and marked for release with a FREE statement.
dynamic table	A table for which the number of occurrences for which storage is allocated may be incremented or decremented at execution time. Not supported in Elastic COBOL.

E

Term	Definition
elementary item	A data item that is described as not being further logically subdivided.
end marker	A marker for the end of a source unit.
end of procedure division	The physical position of a COBOL source program after which no further procedures appear.
end program marker	A combination of words, followed by a separator period, that indicates the end of a COBOL source program. The end program marker is: END PROGRAM program-name. Or END PROGRAM “program-name”. Or END PROGRAM.
enter key	A key on the keyboard of a character addressable terminal that signals that the input of the screen item or the screen record is complete.
entry	Any descriptive set of consecutive clauses terminated by a separator period and written in the identification division, environment division, or data division of a COBOL program.
exception condition	A condition detected during the execution of a program that indicates that an error or exception to normal processing has occurred.
execution time	The time at which a program is executed.
explicit scope terminator	A reserved word that terminates the scope of a particular procedure division statement.
exponent	In floating-point representation, that power to which the radix of the representation is to be raised.
expression	An arithmetic or conditional expression.
extend mode	The state of a file connector after execution of an OPEN statement, with the EXTEND phrase specified, for that file connector and before the execution of a CLOSE statement without the REEL or UNIT phrase for that file connector.

Term	Definition
extended letter	A letter, other than the basic letters, in the set of characters defined for the COBOL character set.
external data	The data described in a program as external data items and external file connectors.
external data item	A data item that is described as part of an external record in one or more programs of a run unit and that may be referenced from any program in which it is described.
external data record	A logical record that is described in one or more programs of a run unit and whose constituent data items may be referenced from any program in which they are described.
external file connector	A file connector that is accessible to one or more object programs in the run unit.
external media format	A form of data suitable for presentation or printing, including any control functions necessary for representation as readable text.
external repository	Storage for information relating to class names, method names, method parameters, and any other related information is provided by the Java environment and active CLASSPATH.
external switch	A software device, SWITCH-1 through SWITCH-26, that is used to indicate that one of two alternate states (true and false) exists. The switch is initially false, but may be made true by passing a parameter at runtime as /1 through /9, /A through /Z, or /a through /z.

F

Term	Definition
feature-name	A system-name that identifies a feature of an input-output device.
field	A contiguous area of a terminal screen that represents an elementary screen item.
figurative constant	A compiler-generated value referenced through the use of certain reserved words or reserved words and alphanumeric, boolean, or national literals. May specifically be a Wang compatible figurative constant.
file	A physical collection of logical records.
file attribute conflict condition	A condition occurring when an unsuccessful attempt has been made to execute an input-output operation on a file and the file attributes, as specified for that file in the program, do not match the fixed attributes for that file.
file connector	A storage area that contains information about a file and is used as the linkage between a file-name and a physical file and between a file-name and its associated record area.
file control entry	A SELECT clause and subordinate clauses that declare the relevant physical attributes of a file.
file description entry	A data division entry that specifies the characteristics of a file.
file-name	A user-defined word that names a file connector described in a file description entry or a sort-merge file description entry within the file section of the data division.
file organization	The permanent logical file structure established at the time that a file is created.
file position indicator	A conceptual entity that contains the value of the current key within the key of reference for an indexed file, or the record number of the current record for a sequential file,

Term	Definition
	or the relative record number of the current record for a relative file, or indicates that no next logical record exists, or that the number of significant digits in the relative record number is larger than the size of the relative key data item, or that an optional input file is not present, or that the at end condition already exists, or that no valid next record has been established.
file section	The section of the data division that contains file description entries and sort-merge file description entries together with their associated record descriptions.
file sharing	A cooperative environment that controls concurrent access to the same physical file.fixed file attributes: Information about a file that is established when a file is created and shall not subsequently be changed during the existence of the file.
fixed-length record	A record associated with a file whose file description or sort-merge description entry requires that all records contain the same number of character positions.
fixed-point numeric item	A numeric data item using fixed-point representation.
fixed-point numeric literal	A quantity, in fixed-point representation, that has a radix of ten and is expressed as a literal composed of one or more numeric characters and optionally either a decimal point or an algebraic sign, or both.
fixed-point representation	A positional representation in which each number is represented by a single sequence of digits, the position of the radix point being fixed with respect to the rightmost end of the set according to the position of the implicit or explicit radix point.
floating-point numeric item	A numeric data item using floating-point representation.
floating-point numeric literal	A quantity, in floating-point representation, that has a radix of ten.
floating-point representation	A number representation in which a number is represented by two sequences of digits, the significant and the exponent.
footing area	The position of the page body adjacent to the bottom margin.
formal parameter	A data-name specified in the USING phrase of the procedure division header that gives the name used in the function, method, or program for a parameter.
format	A specific arrangement of a set of data.
function	A temporary data item whose value is determined at the time an intrinsic or a user-defined function is referenced during the execution of a statement.
function-identifier	An identifier that references an intrinsic-function-name, a user-defined-name, or a function-prototype-name.
function key	A key on the keyboard of a character addressable terminal that, when enabled and pressed, signals that input of the screen record is complete and a function is requested.

G

Term	Definition
global name	A name declared in only one program that may be referenced from that program and from any program contained within that program.
graphic character	A character, other than a control function, that has a visual

Term	Definition
	representation normally handwritten, printed, or displayed. [ISO/IEC 10646-1].
graphic symbol	The visual representation of a graphic character or of a composite sequence. [ISO/IEC 10646-1] .
group item	A data item that is composed of subordinate data items.

H

Term	Definition
high-order end	The leftmost position of a string of characters or a string of bits.

I

Term	Definition
i-o mode	The state of a file connector after execution of an OPEN statement, with the I-O phrase specified, for that file connector and before the execution of a CLOSE statement without the REEL or UNIT phrase for that file connector.
i-o status	A conceptual entity that contains a two-character value indicating the status of an input-output operation.
identifier	A language element that uniquely references a data item or a function.
imperative statement	A statement that specifies an unconditional action to be taken or a conditional statement that is delimited by its explicit scope terminator (delimited scope statement). An imperative statement may consist of a sequence of imperative statements.
implicit scope terminator	A separator period that terminates the scope of any preceding unterminated statement, or a phrase of a statement that by its occurrence indicates the end of the scope of any statement contained within the preceding phrase.
index	A computer storage area or register, the content of which represents the identification of a particular element in a table.
index data item	A data item in which a value associated with an index-name may be stored in a form specified by the implementor.
index-name	A user-defined word that names an index associated with a table.
indexed file	A file with indexed organization.
indexed organization	The permanent logical file structure in which each record is identified by the value of one or more keys within that record.
inheritance (for classes)	A mechanism for using the interface and implementation of one class as the basis for another class. A subclass inherits from one superclass.
initial data	The data described in the working-storage or file section of an initial program.
initial data item	A data item that is described as part of an initial data record.
initial data record	A logical record that is described in the working-storage or file section of an initial program and is initialized on every call to the program.
initial file connector	A file connector that is described in an initial program and is not in an open mode on any call to the program.
initial program	A program that is placed into an initial state every time the program is called in a run unit.

Term	Definition
initial state	The state of a program when it is first called in a run unit.
in-line comment	a comment preceded on a source line by one or more COBOL words or character-strings.
input file	A file that is opened in the input mode.
input mode	The state of a file connector after execution of an OPEN statement, with the INPUT phrase specified, for that file connector and before the execution of a CLOSE statement without the REEL or UNIT phrase for that file connector.
input-output control system; IOCS	A system provided by the Elastic COBOL runtime that directs, or controls, the processing of files.
input-output file	A file that is opened in the I-O mode.
input-output section	The section of the environment division that names the files and the external media required by an object program and provides information required for transmission and handling of data during execution of the object program.
input-output statement	A statement that causes files to be processed by performing operations upon individual records or upon the file as a unit. The input-output statements are: CLOSE, DELETE, OPEN, READ, REWRITE, START, and WRITE.
input procedure	A set of statements to which control is given during the execution of a SORT statement, for the purpose of controlling the release of specified records to be sorted.
integer:	1. A fixed-point numeric literal that does not include any digit positions to the right of the decimal point. 2. A fixed-point numeric data item defined in the data division that does not include any digit positions to the right of the decimal point. 3. An integer function 4. A standard intermediate data item whose decimal fixed-point representation contains only zeros to the right of the decimal point. NOTE 1 The unique value zero is an integer.
integer function	A function whose category is numeric and whose definition provides that all digits to the right of the decimal point are zero in the returned value for any possible evaluation of the function.
internal data	The data described in a program excluding all external data items and external file connectors.
internal data item	A data item described in one program.
internal file connector	A file connector that is accessible to only one object program in the run unit.
intrinsic-function-name	A word that names a mechanism to determine the value of a function.
invalid key condition	A condition occurring at execution time when a specific value of the key associated with an indexed or relative file is determined to be invalid invocation; method invocation: See method invocation.
IOCS; input-output control system	See input-output control system.
item identification	The process of identifying the data item referenced by an identifier.

J

Term	Definition
Java	A programming language centered around object-orientation and platform independence. Its executable files are stored in files ending in .class, which may in turn be archived into code archives ending in .jar. Its executable format is platform independent bytecode and is either interpreted or compiled by a Just In Time compiler to native code at runtime. It is typically invoked from the command line using a 'java' or 'jre' command.
Java Development Kit	The JRE plus Java development tools. Originally, Java was available only as the JDK. Often, Java versions refer to JDK numbers, such as JDK 1.02, JDK 1.1, JDK 1.2, JDK 1.3. As of JDK 1.2, a separate naming scheme is used; Java 2 is the same as JDK 1.2.
Java Virtual Machine	The interpreter or Just In Time compiler which is capable of executing Java executable files (.class files consisting of bytecode).
JDK	See Java Development Kit.
JVM	See Java Virtual Machine.
Java Runtime Environment	The JVM and associated runtimes for executing Java, not including Java development tools.
JRE	See Java Runtime Environment.

K

Term	Definition
key	A data item that identifies the location of a record, or a set of data items that serve to identify the ordering of data.
key of reference	The key, either prime or alternate, currently being used by a file connector to access records within an indexed file.
key word	A reserved word or intrinsic-function-name whose presence is required when the general format in which the word appears is used in a source program.

L

Term	Definition
letter	A basic letter or an extended letter.
level indicator	Two alphabetic characters that identify a specific type of file or a position in a hierarchy. The level indicators in the data division are: CD, FD, RD, and SD.
level-number	A user-defined word, expressed as a one or two digit number, that indicates the hierarchical position of a data item or the special properties of a data description entry.
library-name	A system-name that names a COBOL library that is to be used by the compiler for a given source program compilation.
library text	Text that resides in a COBOL library for the purpose of being introduced into a source program at compile time by a COPY statement.
linkage section	A section in the data division of a called program that describes data items available from a calling program.

Term	Definition
literal	A character-string whose value is derived from the ordered set of characters in the string.
locale	The definition of the subset of a user's information technology environment that depends on language and cultural conventions.
lock mode	The state of an open file connector when record locking is in effect that indicates whether record locking is manual or automatic.
logical operator	One of the reserved words AND, OR, or NOT.
logical page	A conceptual entity consisting of the top margin, the page body, and the bottom margin.
logical record; record	A data item defined with level number 01.
low-order end	The rightmost position of a string of characters or a string of bits.

M

Term	Definition
mass storage	A storage medium in which data may be organized and maintained in both a sequential and non-sequential manner.
mass storage control system; MSCS	An input-output control system that directs, or controls, the processing of mass storage files.
mass storage file	A collection of records that is assigned to a mass storage medium.
MCS; message control system	See message control system. Not supported in Elastic COBOL.
merge file	A collection of records to be merged by a MERGE statement.
message	Data associated with an end of message indicator or an end of group indicator in the communications facility.
message control system; MCS	A communication control system that supports the processing of messages. Not supported in Elastic COBOL.
method	Procedural code that is associated with a particular object and its data or class and its data. A method is to object-oriented programming as a function or procedure is to functional or procedural programming.
method data	The data declared in the data division of a method definition.
method definition	The source unit that defines a method.
method invocation; invocation	The request to execute a named method on a given object. A method invocation identifies an object, a method name, and the parameters required by the method definition.
method-name	A user-defined word that identifies a method.
mnemonic-name	A user-defined word that is associated in the environment division with a specific system-name.
MSCS; mass storage control system	See mass storage control system.
multiple (of a COLUMN, LINE, SOURCE or VALUE clause in the report section)	Having more than one integer operand.

N

Term	Definition

Term	Definition
national character	Any character in a national character set.
national character position	The amount of physical storage required to store, or presentation space required to print or display, a single character whose usage is national.
national character set	Any character set that is used to represent data associated with COBOL's usage national.
national collating sequence	A collating sequence associated with a national character set.
national function	A function whose value is composed of a string of one or more national characters from a national character set.
national literal	A literal delimited by the opening delimiter N" or G" on the left and a matching quotation symbol on the right. The literal has the class and category national and its content may be one or more characters from the computer's national character set.
native alphanumeric character set	A character set, used to represent data associated with COBOL's usage display, that consists of all characters in the computer's alphanumeric character set.
native arithmetic	A mode of arithmetic in which the techniques used in handling arithmetic are specified by the implementor.
native character set	Elastic COBOL has Unicode as its native character set.
native collating sequence	The Unicode numeric collating sequence.
native national character set	A character set, used to represent data associated with COBOL's usage national, that consists of all characters in the computer's national character set.
negated combined condition	The 'NOT' logical operator immediately followed by a parenthesized combined condition.
negated simple condition	The 'NOT' logical operator immediately followed by a simple condition.
next executable sentence	The next sentence to which control will be transferred after execution of the current statement is complete.
next executable statement	The next statement to which control will be transferred after execution of the current statement is complete.
next record	The record that logically follows the current record of a file.
noncontiguous item	Elementary data items in the working-storage, local-storage, and linkage sections that bear no hierarchic relationship to other data items.
numeric character	A character that belongs to the following set of digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
numeric function	A function whose class and category are numeric but that for some possible evaluation does not satisfy the requirements of an integer function.
numeric item	A data item whose description restricts its content to a value represented by characters chosen from the digits '0' through '9' and optionally the signs '+', '-', or other representation of an operational sign.
numeric literal	Either a fixed-point numeric literal or a floating-point numeric literal.

O

Term	Definition
object (instance)	A unit consisting of data and the methods that act upon that data.

Term	Definition
object computer entry	An entry in the OBJECT-COMPUTER paragraph of the environment division that describes the computer environment in which the object program is to be executed.
object data	Data described in the data division of an object definition, excluding data described in its methods.
object definition	The source unit that defines an object.
object identifier	An identifier that identifies an object. An object identifier may be a predefined object identifier or an object reference identifier.
object method	A method of an object (as opposed to a factory method), although a factory method is a method of a factory object.
object of entry	A set of operands and reserved words, within a data division entry of a COBOL program, that immediately follows the subject of the entry.
object program	A set or group of executable machine language instructions and other material designed to interact with data to provide problem solutions. In this context, an object program is generally the machine language result of the operation of a COBOL compiler on a source program. Where there is no danger of ambiguity, the word 'program' alone may be used in place of the phrase 'object program'.
object property	A name that may be used to qualify an object reference to get a value from or pass a value to an object.
object reference	An data item that contains a reference to an object.
object reference identifier	An identifier that identifies an object by an object reference.
obsolete element	A language element in Standard COBOL that is to be deleted from the next revision of Standard COBOL.
open mode	The state of a file after execution of an OPEN statement for that file and before the execution of a CLOSE statement without the REEL or UNIT phrase for that file. The particular open mode is specified in the OPEN statement as either INPUT, OUTPUT, I-O, or EXTEND.
operational sign	An algebraic sign associated with a numeric data item or a numeric literal that indicates whether its value is positive or negative.
optional file	A file declared as being not necessarily present each time the program is executed.
optional word	A reserved word that is included in a specific format only to improve the readability of the language and whose presence is optional to the user when the format in which the word appears is used in a source program.
output file	A file that is opened in either the output mode or extend mode.
output mode	The state of a file connector after execution of an OPEN statement, with the OUTPUT or EXTEND phrase specified, for that file connector and before the execution of a CLOSE statement without the REEL or UNIT phrase for that file connector.
output procedure	A set of statements to which control is given during execution of a SORT statement after the sort function is completed, or during execution of a MERGE statement after the merge function reaches a point at which it can select the next record in merged order when requested.

P

Term	Definition
padding character	An alphanumeric character used to fill the unused character positions in a physical record.
paragraph	1. In the procedure division, a paragraph-name followed by a separator period and by zero, one, or more sentences. 2. In the identification division, a paragraph header followed by one entry. 3. In the environment division, a paragraph header followed by zero, one, or more entries.
paragraph header	A reserved word, followed by the separator period, that indicates the beginning of a paragraph in the identification and environment divisions.
paragraph-name	A user-defined word that identifies and begins a paragraph in the procedure division.
phrase	An ordered set of COBOL character-strings that specifies an attribute of an entry or a statement and is a subset of a clause or paragraph in all divisions except the procedure division, and a subset of a statement in the procedure division.
physical page	A device-dependent concept.
physical record	The term is synonymous with block.
predefined object identifiers	The reserved words NULL, SELF, and SUPER used to identify particular objects.
prime record key	A key whose contents uniquely identify a record within an indexed file.
procedure	A paragraph or group of logically successive paragraphs, or a section or group of logically successive sections, within the procedure division.
procedure branching statement	A statement that causes the explicit transfer of control to a statement other than the next executable statement in the sequence in which the statements are written in the source program. The procedure branching statements are: CALL, EXIT, EXIT PROGRAM, GO TO, GOBACK, MERGE (with the OUTPUT PROCEDURE phrase), PERFORM, RECOVER, and SORT (with the INPUT PROCEDURE or OUTPUT PROCEDURE phrase).
procedure-name	A user-defined word that names a paragraph or section in the procedure division.
processor	The computing system that enables a user to compile and execute source units, consisting of both hardware and relevant associated software.
program identification entry	An entry in the PROGRAM-ID paragraph of the identification division that specifies the program-name and assigns selected program attributes to the program.
program-name	A user-defined word that identifies a COBOL source program.
program-pointer data item	A data item in which the reference to a program may be stored.
program prototype definition	A definition that specifies the rules governing the class of the parameters expected to be received by a particular subprogram, and any other requirements needed to transfer control to, and get control and return information back from that subprogram.
property	See object property.
pseudo-text	A sequence of text-words, comment lines, or the separator space in a source program or COBOL library bounded by, but

Term	Definition
	not including, pseudo-text delimiters.
pseudo-text delimiter	Two contiguous characters "==" used to delimit pseudo-text.
punctuation character	A character used in formation of separators.

Q

Term	Definition
qualified data-name	An identifier composed of a data-name followed by one or more sets of either of the connectives OF or IN followed by a data-name qualifier.
qualifier	1. A data-name or a name associated with a level indicator that is used in a reference either together with a data-name of an item that is subordinate to the qualifier or together with a condition-name. 2. A section-name that is used in a reference together with a paragraph-name specified in that section. 3. A library-name that is used in a reference together with a text-name associated with that library.
quotation symbol	Either the COBOL character quotation mark (") or the COBOL character apostrophe (') used in the opening and closing delimiters of literals.

R

Term	Definition
radix	In positional representation of numeric values, that positive integer by which the significance of a digit place must be multiplied to give the significance of the next higher digit position.
radix point	A generalization of the decimal point for numeric representations in any radix.
random access	An access mode in which the program-specified value of a key data item identifies the logical record that is obtained from, deleted from, or placed into a relative or indexed file.
record; logical record	See logical record.
record area	A storage area allocated for the purpose of processing the record described in a record description entry in the file section of the data division.
record description; record description entry	The total set of data description entries associated with a particular record.
record description entry; record description	See record description.
record key	A key, either a prime record key or an alternate record key, whose contents identify a record within an indexed file.
record-key-name	A user-defined word that names a key associated with an indexed file. This key may be made up from one or more portions of the record area associated with the file.
record lock	An indicator that is associated with a specific record in a file and is set and released by the locking facility. It is used to determine whether multiple file connectors may access the record concurrently.
record locking	The controlling of record access for shared files in which a

Term	Definition
	record lock prevents access to the associated record from other file connectors.
record-name	A data-name that names a record described in a record description entry.
record number	The ordinal number of a record in a file whose organization is sequential.
reel; unit; volume	A discrete portion of a storage medium that contains part of a file, all of a file, or any number of files.
reference format	A format that provides a standard method for describing COBOL source programs.
reference-modifier	An identifier that references a unique data item created by specifying an identifier, a starting position, and a length.
relation character	A character that belongs to the following set: Character Meaning > greater than < less than = equal to
relation condition	A proposition, for which a truth value may be determined, that the value of an expression, data item, literal, or index has a specific relationship to another expression, data item, literal, or index.
relational operator	A reserved word, a relation character, a group of consecutive reserved words, or a group of consecutive reserved words and relation characters used in the construction of a relation condition.
relative file	A file with relative organization.
relative key	A key whose contents identify a logical record in a relative file.
relative organization	The permanent logical file structure in which each record is uniquely identified by an integer value greater than zero, that specifies the record's logical ordinal position in the file.
relative record number	The ordinal number of a record in a file whose organization is relative.
report writer	A comprehensive set of data clauses and statements that enable a print layout to be described according to its general appearance rather than through of a series of procedural steps. Not supported in Elastic COBOL.
reserved word	A word used in the COBOL language that is not available for use as a user-defined word or a system-name.
resource	A facility or service, controlled by the operating system, that can be used by an executing program.
restricted pointer	A pointer data item that is restricted to data items of a specified type or to programs with the same signature as a specified program.
resultant identifier	A user-defined data item that is to contain the result of an arithmetic operation.
run time entity	The executable unit resulting from compiling a source element, which may be a function, method, or program.
run unit	One or more object programs that interact with one another and that function, at execution time, as an entity to provide problem solutions.

S

Term	Definition
screen description entry	A data description entry in the screen section that describes a screen item and its attributes.
screen item	A unit of display upon a character addressable terminal.
screen-name	A user-defined word that names a screen item described in a screen description entry.
screen record	A screen description entry with a level-number of 01.
section	A set of zero, one, or more paragraphs or entries, called a section body, the first of which is preceded by a section header.
section header	A combination of words followed by a separator period that indicates the beginning of a section in the environment, data, and procedure division.
section-name	A user-defined word that names a section in the procedure division.
sentence	A sequence of one or more statements, the last of which is terminated by a separator period.
separately-compiled program	A program that, together with its contained programs, is compiled separately from all other programs.
separately-compiled source unit	A class definition, function definition, function prototype definition, interface definition, or program prototype definition that can be compiled separately from any other source unit.
separator	A character or two contiguous characters used to delimit character-strings.
sequential access	An access mode in which logical records are obtained from or placed into a file in a consecutive predecessor-to-successor logical record sequence determined by the order of records in the file.
sequential file	A file with sequential organization.
sequential organization	The permanent logical file structure in which a record is identified by a predecessor-successor relationship established when the record is placed into the file.
shared file	A file for which file sharing has been specified.
sharing mode	The state of an open file connector that indicates the mode of file sharing. The sharing modes are: sharing with all other, sharing with no other, and sharing with read only.
sign condition	The proposition, for which a truth value can be determined, that the algebraic value of a data item or an arithmetic expression is either less than, greater than, or equal to zero.
signature	The information stored about a compilation unit in the external repository.
significant	In floating-point representation, the fixed-point numeral that represents the significant digits of the number.
simple condition	Any single condition chosen from the set: • relation condition • class condition • condition-name condition • switch-status condition • sign condition • (simple-condition)
size error condition	A condition that may be raised during the execution of arithmetic statements to indicate that a size error occurred and is tested with the SIZE ERROR phrase of arithmetic statements. It is not

Term	Definition
	associated with an exception status indicator
sort file	A collection of records to be sorted by a SORT statement.
sort-merge file description entry	A data division entry that specifies the characteristics of a sort or merge file.
source computer entry	An entry in the SOURCE-COMPUTER paragraph of the environment division that describes the computer environment in which the source program is to be compiled.
source element	A source unit excluding any contained source units.
source item	An identifier designated by a SOURCE clause that provides the value of a printable item.
source text manipulation statement	A statement beginning with the word COPY or the word REPLACE that provides the capability to insert and replace source program text as part of the compilation of the source program.
source unit	A sequence of statements beginning with an identification division and finishing with an end marker or the end of the compilation group, including any contained source units.
special character word	A reserved word that is composed entirely of special characters.
special registers	Certain compiler generated storage areas whose primary use is to store information produced in conjunction with the use of specific COBOL features.
standard data format	A concept used to describe data in a COBOL data division whereby the characteristics or properties of data are expressed in terms of the appearance of graphic characters on a printed page, rather than the manner in which data is stored internally in the computer or on a particular external medium.
static data	The data described in the file or working-storage section of a source unit that is not an initial program.
static data item	A data item that is described as part of a static data record.
static data record	A logical record described in the file or working-storage section of a source unit that is not an initial program.
subclass	A class that inherits from another class. When two classes in an inheritance relationship are considered together, the subclass is the inheritor or inheriting class; the superclass is the inheritee or inherited class. NOTE - In the industry literature, the term derived class is also often used as an alternative to the term subclass. These terms are equivalent.
subject of entry	The data item that is being defined by a data description entry.
subprogram; called program	See called program.
subscript	An occurrence number used to index a specific element of a table, or in the case of the subscript 'ALL', all elements of a table.
subscripted data-name	An identifier that is composed of a data-name followed by one or more subscripts enclosed in parentheses.
superclass	A class that is inherited by another class. See also subclass.
switch-name	A system-name that identifies a defined external switch SWITCH-1 to SWITCH-26..
switch-status condition	The proposition, for which a truth value can be determined, that a switch, capable of being set to an 'on' or 'off' status, has been set to a specific status.
symbolic-character	A user-defined word that specifies a user-defined figurative constant

Term	Definition
system-name	A COBOL word that is used to communicate with the operating environment.

T

Term	Definition
table	A set of logically consecutive items of data that are defined in the data division of a COBOL program by means of the OCCURS clause.
table element	A data item that belongs to the set of items in a table.
terminal (in the screen section)	A character addressable device that includes a display and a keyboard.
termination key	Any function key or the enter key of a terminal.
text-name	A system-name that identifies library text.
top margin	An empty area that precedes the page body.
truth value	The representation of the result of the evaluation of a condition in terms of one of two values: true, false.

U

Term	Definition
Unary operator	A '+' or a '-' sign preceding a variable or a left parenthesis in an arithmetic expression.
Unit; reel; volume	See reel.
Universal object reference	An object reference that is not restricted to a specific class or interface.
Unsuccessful execution	The attempted execution of a statement that does not result in the execution of all the operations specified by that statement.
User-defined word	A COBOL word supplied by the user to satisfy the format of a clause or statement.

V

Term	Definition
Variable	A data item whose value may be changed by execution of the program.
Variable-length record	A record associated with a file whose file description or sort-merge description entry permits records to contain a varying number of character positions.
Variable-occurrence data item	A table element that is repeated a variable number of times. Such an item contains an OCCURS clause with the DEPENDING phrase in its data description entry or is subordinate to such an item.
Volume; unit; reel	See reel.

Appendix D - Runtime Configuration

Configuration Parameters

Configuration Parameters control the runtime behavior of the Elastic COBOL runtime system. Configuration parameters may be set in a number of ways.

1. They may be set on the command line as System Properties. This is the only way to set items initialized only once and then never checked again:
   java -Dconfig=value myprogram
2. They may be set as program parameters on the command line:
   java myprogram config=value
3. They can be set from a configuration file, the first found in the search order of:
   cblconfig
   cblconfi
   /etc/cblconfig
   /etc/cblconfi

   A configuration file has comment lines beginning with '#'. Set lines are a name, followed by a space, tab, colon ':' or equals sign '=', followed by a value. A configuration file may be included in a deployment .jar file.
4. They can be set from a properties file, the name of the main executable class (without .class) followed by .ini. This is in Java properties format, which is comment lines beginning with '#', and set lines as name=value.
5. In program code, they may be set by using the SET verb:
   SET CONFIGURATION "config" TO "value"

In the configuration variables below, the type Boolean implies that 0 or 1, Y or N, T or F may be used. Integer is a decimal number without decimal places.

File Serving

Elastic COBOL permits four methods accessing remote file systems. These are related to the type of file the format of the data and the remote host that may be serving the data. These file serving mechanisms are:

1. Elastic COBOL File System -- access to Indexed, Sequential, and Relative files written with Elastic COBOL.
2. AS/400 Distributed Data Management (DDM) – access through the AS/400 toolbox to AS/400 Indexed files.
3. NFS – Network File System access to remote sequential files.
4. AcuConnect – access to AcuCOBOL Indexed files.

All files accessed through Elastic COBOL runtimes can READ and WRITE remote files.

Remote Files with AcuConnect

Elastic COBOL permits access to AcuCOBOL files and AS/400 files with the use of the AcuConnect configuration file format. This format is only applicable with Elastic COBOL runtime services.

Remote File General Information

FILE_PREFIX	Specify a series of space separated prefixes to be applied to a file when searching for the file. Setting: {file_prefix {space}}…
USERNAME or USER	Specify the username of the AcuConnect or AS/400. Setting: User name text
LOCALHOST	Replace default localhost name with this text. Setting: localhost alternate name.
SERVER_PORT	N/A for AS/400 Specify the server port used to connect to using AcuConnect. Setting: Integer
ACUCONNECT_RUNTIME_FLAGS	N/A for AS/400 Specify parameters to AcuConnect that would normally be put on the runtime command line when starting a remote application. Setting: AcuConnect runtime parameters
ACUCONNECT_CONFIGURATION_FILE	Specify configuration file to remote program. Setting: AcuConnect configuration filename
COMPRESS_FACTOR	Compression factor for AcuConnect indexed files. Setting: Integer
CODE_PREFIX	Specify the space-delimited search path for AcuConnect to find a CALLed program. Setting: {code-prefix {space}}…
CODE_MAPPING	Enable code-mapping, checking if the Called name is a config parameter, and replacing it with the config parameter's contents if it is. Setting: 0 to disable code aliases 1 to enable code aliases
CODE_SUFFIX	Specify text to be automatically appended to program names with an extension. This is primarily for when CALLing .acu files remotely. Avoid this parameter and CALL the actual desired name. Setting: Extension text
ACUCONNECT_CLOSE_AFTER_CANCEL	Specify whether the connection to a remote application should remain open or be closed after a CALL is complete. Setting: 0 to leave open 1 to close
AS400CONNECT_SERVER	Specify the hostname or IP address of the AS/400 server to which to connect.

AS400CONNECT_CLOSE_AFTER_CANCEL	Specify whether the connection to a remote application should remain open or be closed after a CALL is complete. Setting: 0 to leave open 1 to close

File

IDXCACHEMODE	Indexed File Cache Mode. Settings: READWRITE or RW, READONLY or RO
IDXCACHE	Percentage of default index cache to use, may be above or below 100. Settings: Integer
FILESYSTEM	Default file format protocol for all file systems. This may be overridden by explicit protocol in ASSIGN. Default is ELASTICCOBOL.
FILESYSTEMIDX	Default file format protocol for indexed files. This may be overridden by explicit protocol in ASSIGN. Default is ELASTICCOBOL. Setting: MF, MFU, MFW, MICROFOCUS, MERANT, ACON, ACONNECT, ACUCONNECT, ISAM, CISAM, C-ISAM, C_ISAM, DISAM, D-ISAM, D_ISAM, ACU, ACUCOBOL, VIS4, VISION4, VIS, VISION
FILESYSTEMSEQ	Default file format protocol for sequential files. This may be overridden by explicit protocol in ASSIGN. Default is ELASTICCOBOL. Setting: MF, MICROFOCUS, MERANT, ACU, ACUCOBOL, ACUCORP
FILESYSTEMREL	Default file format protocol for sequential files. This may be overridden by explicit protocol in ASSIGN. Default is ELASTICCOBOL. Setting: MF, MICROFOCUS, MERANT, ACU, ACUCOBOL, ACUCORP
REMOTEHOST	Default remote: host name. Setting: Hostname or IP Address

Console and Graphics

CONSOLEFG	Settings: BLACK, BLUE, GREEN, CYAN, RED, MAGENTA, YELLOW, BROWN, WHITE, BRIGHT-BLACK, BRIGHT-BLUE, BRIGHT-GREEN, BRIGHT-CYAN, BRIGHT-RED, BRIGHT-MAGENTA, BRIGHT-YELLOW, BRIGHT-BROWN, BRIGHT-WHITE
CONSOLEBG	Settings: BLACK, BLUE, GREEN, CYAN, RED, MAGENTA, YELLOW, BROWN, WHITE, BRIGHT-BLACK, BRIGHT-BLUE, BRIGHT-GREEN, BRIGHT-CYAN, BRIGHT-RED, BRIGHT-MAGENTA, BRIGHT-YELLOW, BRIGHT-BROWN, BRIGHT-WHITE
SECURE-CHAR	Setting: Any character to be used as the secure character, default is '*'.
CONSOLE_FONT_SIZE[_lang	The default console font size.

uage[_region]]	Setting: Integer
CURSOR-MODE	Setting: 1 Do not hide cursor 2 Do not show cursor 3 Normal cursor
CONSOLE_FONT	Set the default font name for the console. Setting: Font name
CONSOLE_CELL	Render console cell by cell rather than line by line. Setting: Boolean (true/false)
CONSOLE_WIDTH	Default font width. Setting: Integer.
CONSOLE_WIDTH_MAX	Maximum default font width. Setting: Integer.
CONSOLE_WIDTH_MIN	Minimum default font width. Setting: Integer
CONSOLE_WIDTH_MULT	Multiplier for default font width. Setting: Real number.
CONSOLE[_language[_region]]	Setting: console_cell,console_width, console_width_min, console_width_max, console_width_mult, console_font
FINISHED-MESSAGE	Message to display in main console title bar when application is finished. Default is 'Application Finished'. Setting: Message text
WINDOW-TITLE	Message to display in main console title bar while application is running. Default is derived from the name of the program. Setting: Message text
COLUMNS	Sets the default number of columns. (Default is 80.) Setting: Integer
LINES	Sets the default number of lines. (Default is 25.) Setting: Integer
SEARCH-PROMPT	In graphical screen section LIST-BOX, this may override the search prompt message. Setting: Message text
UI	Look and Feel User Interface classname. Default is native platform look and feel. Setting: METAL, MOTIF, WINDOWS, Classname
CRTKEY1	Allow CRTKEY1 of CRT STATUS to be stored as a binary value rather than ASCII value. Setting: BINARY, ASCII (default)
PFTERMS	By default, all function keys are allocated to the system and are designated as terminators. To define which function keys are not terminators or which are user-defined function key terminators, set the PFTERMS parameter. The function key numbers can be from 1 to 24. Those listed, when preceded by an exclamation point, are disabled as terminators. Those listed but not preceded by an exclamation point are allocated as user defined function key terminators. Those not listed are allocated as system defined function key terminators. Example: PFTERMS=!1-6,7,!8,9-20 The above example allocates function keys 1-6 and function key 8 as non-terminal function keys. It also allocates function keys 7 and 9 through 20 as user

	defined function key terminators. All others are allocated as system defined function key terminators. Setting: [!]<function key number>[-<function key number>],…
TERMINATE	By default, the <ENTER> key is the normal terminator for ACCEPT statements. To create custom normal termination keys for ACCEPT statements, set the program parameter TERMINATE. The possible modifiers are CTRL or CONTROL, ALT, or SHIFT. Setting: [<Modifier>-]<key>,… The possible keys are:

Possible Terminator Keys

0..9	A-Z	ACCEPT	BACK_QUOTE
BACK_SLASH	BACK_SPACE	CANCEL	CAPS_LOCK
CLEAR	CLOSE_BRACKET	COMMA	CONTROL
CONVERT	DECIMAL	DELETE	DIVIDE
DOWN	END	ENTER	EQUALS
ESCAPE	F1..F12	FINAL	HELP
HOME	INSERT	KANA	KANJI
LEFT	META	MODECHANGE	MULTIPLY
NONCONVERT	NUM_LOCK	NUMPAD0..NUMPAD9	OPEN_BRACKET
PAGE_DOWN	PAGE_UP	PAUSE	PERIOD
PRINTSCREEN	QUOTE	RIGHT	SCROLL_LOCK
SEMICOLON	SEPARATOR	SHIFT	SLASH
SPACE	SUBTRACT	TAB	UP

Graphical Screen Section

USER-GRAY or USER-GREY	Hex encoded color value overriding discovered value for USER-GRAY color in graphical screen section. Setting: RGB value as decimal, 0xRRGGBB, #RRGGBB, 0RRRGGGBBB.
USER-WHITE	Hex encoded color value overriding discovered value for USER-GRAY color in graphical screen section. Setting: RGB value as decimal, 0xRRGGBB, #RRGGBB, 0RRRGGGBBB.
FIXED-FONT TRADITIONAL-FONT LARGE-FONT MEDIUM-FONT SMALL-FONT DEFAULT-FONT	Set the appropriate default font for the graphical screen section. Setting: Fontname[-bold][-italic][-bolditalic][-size]
EF-UPPER-WIDE	Graphical screen section ENTRY-FIELD setting. If set to the UPPER style, use the wide font measure if this is set to true. Setting: 0 or 1

EF-WIDE-SIZE	Graphical screen section ENTRY-FIELD setting. This sets the boundary below which the wide font measure rather than standard font measure is used. Setting: Integer
FONT-WIDE-SIZE-ADJUST	Graphical screen section ENTRY-FIELD setting. This value is added directly to the computed wide measure font size. Setting: Integer
BROWSER.SEARCH	Graphical screen section browser component search page. Default is blank. Setting: HTTP URL
BROWSER.HOME	Graphical screen section browser component home page. Default is blank. Setting: HTTP URL
TEMPORARY-CONTROLS	Set if all controls created are temporary controls. Setting: Boolean
VERBOSE-CALL	Prompt with dialog if CALL target is not found. Setting: Boolean
MPE-TRACE	Visibly show MPE Intrinsic CALL being done. Setting: Boolean
DEFSYS	Default IO device is system rather than console. Setting: Boolean

CGI Support

CGI support requires that the program be passed several environment variables as configuration variables. (Java does not have access to the environment variables used by CGI unless explicitly passed.) This may be setup as a script or batch file for CGI.

HTML-TEMPLATE-PREFIX	A list of search locations for HTML files. Setting: file-prefix…
REQUEST_METHOD	This should be passed in at runtime for CGI support, as: REQUEST_METHOD=$REQUEST_METHOD (Posix) REQUEST_METHOD=%REQUEST_METHOD% (Windows) Setting: GET or POST
QUERY_STRING	This should be passed in at runtime for CGI support, as: QUERY_STRING=$QUERY_STRING (Posix) QUERY_STRING=%QUERY_STRING% (Windows)
CONTENT_LENGTH	This should be passed in at runtime for CGI support, as: -DCONTENT_LENGTH=$CONTENT_LENGTH (Posix) -DCONTENT_LENGTH=%CONTENT_LENGTH% (Windows) Setting: Integer value

MQ-Series

MQSERVER	Set MQ-Series support to be the server environment rather than client environment. Setting: Boolean (true if MQSERVER environment rather than MQCLIENT environment)
MQ_HOSTNAME	Set MQ-Series remote connection host name or IP address. Setting: Host name
MQ_PORT	Set MQ-Series remote connection port number. Setting: Port Number
MQ_CHANNEL	Set MQ-Series channel. Setting: Channel
MQ_USER_ID	Set MQ-Series User ID. Setting: User ID
MQ_PASSWORD	Set MQ-Series Password. Setting: Password

General Runtime

DEBUGMODE	If the program was compiled in debug mode, start the program in the debugger. Setting: Boolean
LOCK-THREAD-TIME-LIMIT	Minimum time limit to wait on locked thread. The setting 0 implies forever. Setting: Milliseconds
S1 … S26	Set switch default state for SWITCH-1 through SWITCH-26. Setting: Boolean
BE or IE or ENCODENATIONAL	Set international encoding to Java encoding format. Default is default for national operating system platform dependent on Java version Setting: Varies per Java platform.
DEBUGGER	Select which debugger to use, Command Line or GUI. Setting: COMMAND, GUI
INTRINSICSJAR	Filename of .jar containing MPE/iX intrinsic definitions. Setting: JAR filename
CICS_CLIENT	CICS Client setup for Java Gateway. Setting: Name of Java Gateway
WINVERSION	Override results found by WIN$VERSION. Default is os.name space os.version. Setting: OS Name, space, OS Version
SQLDRIVER	Set the default SQL driver name for when no driver name is specified. The standard default is 'sun.jdbc.odbc.JdbcOdbcDriver'. Setting: Default SQL driver name
SQLDRIVERSEARCH	Search for the appropriate driver using heuristics from the given information. Setting: Boolean

Native Code

DLL-LINK	Automatically load and unload specified DLL's, allowing access closer to directly linked behavior. Setting: Comma delimited list of shared libraries (.DLL, .so, .sa, .sl, etc.) to automatically load and cancel.
DLL-CONVENTION	Calling convention to use when calling native code in DLL's. Setting: 0 for C, 1 for STDCALL

Record Locking

LOCKPORT	Set the port number for the record lock server. Setting: Port Number for Lock Server
LOCKSERVER	Set the hostname or IP address of the record lock server. Default is localhost. Setting: Host or IP address

Data

DISPLAY.0	How to display USAGE DISPLAY items. Setting: I Internal display, just render the internal memory storage. E External display of data, reformat data nicely with sign and decimal. S Smart display of data (default); reformat if internal display would be misleading as to sign or decimal.
DISPLAY.1	How to display USAGE COMP-1 items. Setting: I Internal display, just render the internal memory storage. E External display of data, reformat data nicely with sign and decimal. S Smart display of data (default); reformat if internal display would be misleading as to sign or decimal.
DISPLAY.2	How to display USAGE COMP-2 items. Setting: I Internal display, just render the internal memory storage. E External display of data, reformat data nicely with sign and decimal. S Smart display of data (default); reformat if internal display would be misleading as to sign or decimal.
DISPLAY.3	How to display USAGE PACKED-DECIMAL items. Setting: I Internal display, just render the internal memory storage. E External display of data, reformat data nicely with sign and decimal. S Smart display of data (default); reformat if internal display would be misleading as to sign or decimal.
DISPLAY.5	How to display USAGE BINARY items. Setting:

	I Internal display, just render the internal memory storage. E External display of data, reformat data nicely with sign and decimal. S Smart display of data (default); reformat if internal display would be misleading as to sign or decimal.
DISPLAY.S	How to display USAGE COMP-S items. Setting: I Internal display, just render the internal memory storage. E External display of data, reformat data nicely with sign and decimal. S Smart display of data (default); reformat if internal display would be misleading as to sign or decimal.
DISPLAY.D	How to display USAGE COMP-D items. Setting: I Internal display, just render the internal memory storage. E External display of data, reformat data nicely with sign and decimal. S Smart display of data (default); reformat if internal display would be misleading as to sign or decimal.

Appendix E - Handle Components

Handle Component Description

A handle-component is created through a DISPLAY and accepted through an ACCEPT. Its properties are modified with a MODIFY, or set through the initial DISPLAY. Its properties may be inquired with an INQUIRE. Setting a property initially in the DISPLAY and setting it later using a MODIFY have an identical syntax. INQUIRE has some extensions which allow it to set certain 'cursor' style items implicitly before the INQUIRE, allowing a simpler INQUIRE syntax.

Styles

A style is a property which has a boolean (true or false) setting; it may be activated by property-name alone or deactivated with NOT property-name.

Properties

Certain properties set with MODIFY cause actions to be taken, rather than a property to be set. This is generally obvious from the name.

Many properties and styles are shared between all handle-components; these are listed first in the Common Properties and Styles.

Properties may take and return a variety of datatypes. These are documented as being boolean, integer, float (taking a decimal), text, or identifier. *(The setting of identifier is for properties that will set identifier to a value, a rare type)* Boolean types may be passed as a 0 or 1. A dash means that a property has no parameter for MODIFY, or that INQUIRE is supported with the same type as the MODIFY. A N/A means that the MODIFY or INQUIRE is not available for the property. Normally, INQUIRE will attempt to match the data to the user type given in the INQUIRE verb. A handle will be passed as an integer to a property.

Certain properties are marked as low or mid level properties. Properties of this classification are oriented more towards Java usage, and are available to provide the maximum power and flexibility. All low and mid level properties are common properties.

Handle-Component Table

Object		Synonyms	Description
Number	Name		
1	LABEL	TEXT-LABEL	Label for text, similar to a protected text field.
2	ENTRY-FIELD		Field where the user may enter data.
3	PUSH-BUTTON		Button which the user pushes.
4	CHECK-BOX		Button which the user checks on and off.

5	RADIO-BUTTON		Grouped button from which the user makes selection.
6	SCROLL-BAR		A scroll bar by which the user may select a value.
7	LIST-BOX		Box with a list of items.
8	COMBO-BOX		Pull down list of items combined with entry-field.
9	FRAME	GROUP-FRAME	Graphical frame to grouping items visually for user.
10	TAB-CONTROL		Tabbed pane, allowing the user to select tabs.
11	BAR		Graphical bar for drawing.
12	GRID		GRID-CONTROL Grid control, similar to a spreadsheet, for 2D data.
13	BITMAP		Image control, showing a graphical bitmap.
14	TREE-VIEW		A collapsible/expandable tree.
15	WEB-BROWSER		An HTML viewer.
1000	SLIDER		Similar to a scroll bar, but allows values to be made
1004	STATUS-BAR		Status bar at the bottom of a window.
1008	MENU		A graphical menu control.

Common Properties and Styles

Properties and styles are generally applicable to all component types.

All MODIFY clauses may also be done in the DISPLAY during the creation of the component.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
SHOWING	N/A	boolean	Is the component currently visible on screen? Is it visible and in a visible container?
ENABLED	boolean	boolean	Enable the component. An enabled component can respond to user input and generate events.
FONT	integer	integer	The integer is a font-handle. A font handle is declared as HANDLE OF FONT. This identifiers the font for the component as a preloaded font.
HELP-ID	integer	integer	The integer is the help identification number. This number is available to any help plug-in.
ENABLED	boolean	boolean	An enabled component may be activated by the user. When not enabled, the component is typically shown in an altered

Name	MODIFY	INQUIRE	Descriptions
			manner, such as being 'grayed-out'.
VISIBLE	boolean	boolean	A visible component is visible to the user if its containing window is visible and it is within the visible rectangle of the window.
COPYRIGHT	N/A	text	Obtain the copyright for the component.
REVERSE-ORDER	boolean	boolean	Is the component's drawing order reversed?
NO-TAB	boolean	boolean	NO-TAB components are skipped rather than entered by the 'Tab' or 'Backtab' keys.
PERMANENT	boolean	boolean	PERMANENT style components may be destroyed only by the DESTROY verb.
TEMPORARY	boolean	boolean	TEMPORARY style components may be destroyed by the DESTROY verb, or placing another component in an identical screen location.
HEIGHT-IN-CELLS	boolean	boolean	HEIGHT-IN-CELLS components have height expressed in cell units rather than control units; the default is control units. This style is applied implicitly when specifying CELLS for the height.
HEIGHT-IN-CONTROL-UNITS	N/A	boolean	HEIGHT-IN-CONTROL-UNITS components have height expressed in internal control units unique to the component. This style is the default.
WIDTH-IN-CELLS	boolean	boolean	WIDTH-IN-CELLS components have width expressed in cell units rather than control units; the default is control units. This style is applied implicitly when specifying CELLS for the width.
WIDTH-IN-CONTROL-UNITS	N/A	boolean	WIDTH-IN-CONTROL-UNITS components have width expressed in internal control units unique to the component. This style is the default.
OVERLAP-LEFT	boolean	boolean	OVERLAP-LEFT components are shifted slightly to the left.
OVERLAP-TOP	boolean	boolean	OVERLAP-TOP components are shifted slightly upwards.
3-D	boolean	boolean	
SELF-ACT	boolean	boolean	The component will act on itself with a default event action.
NOTIFY	boolean	boolean	The component will send notification events that

Name	MODIFY	INQUIRE	Descriptions
			otherwise would not be sent.
TERMINATION-VALUE	integer	integer	The value returned by the component upon termination in the termination-value slot.
EXCEPTION-VALUE	integer	integer	The value returned by the component upon exception in the exception-value slot.
TITLE	text	text	The main text of the component, such as a title or label. It may include an ampersand (&) character to precede an implicit hot-key if applicable for the component. (The ampersand hot-key works on all platforms, not just Windows.) The TITLE is the implicit property assigned if no other property name is given after a handle-component. This allows PUSH-BUTTON "My &Push Button", or RADIO-BUTTON "Check Me" to be declared.
KEY	text	text	The hot-key for the component if used. The hot-key is the keypress which brings focus directly to the component, or the next focusable component after the component if the component cannot have focus.
ID	integer	integer	The identification number for a component. The handle identification may change for a component from run to run, but the identification number will remain. This must be assigned by the program if it is to be read. The identification number is not used by the graphical runtime; it is reserved for COBOL program usage only.
HIGH	boolean	boolean	The component will be displayed in highlighted mode.
HIGHLIGHT	boolean	boolean	The component will be displayed in highlighted mode.
BOLD	boolean	boolean	The component will be displayed in highlighted mode.
LOW	boolean	boolean	The component will be displayed in non-highlighted mode.
LOWLIGHT	boolean	boolean	The component will be displayed in non-highlighted mode.
FOREGROUND-COLOR	integer	integer	Set the foreground-color of the

Name	MODIFY	INQUIRE	Descriptions
			component to the given color value. Some components have custom foreground-colors and will ignore this color request.
BACKGROUND-COLOR	integer	integer	Set the background-color of the component to the given color value. Some components have custom background-colors and will ignore this color request.
FOREGROUND-COLOUR	integer	integer	Set the foreground-color of the component to the given color value. Some components have custom foreground-colors and will ignore this color request.
BACKGROUND-COLOUR	integer	integer	Set the background-color of the component to the given color value. Some components have custom background-colors and will ignore this color request.
COLOR	integer	integer	Set the combined color value for the component. Some components have custom color values and will ignored portions of this color request.
COLOUR	integer	integer	Set the combined color value for the component. Some components have custom color values and will ignored portions of this color request.
STYLE	integer	integer	Set the style of the component. The style is a set of boolean properties for an individual component, composed from a bit vector of individual properties. Constants for these styles are available from the internal copyfiles (see Appendix).
BACKGROUND-HIGH	boolean	boolean	Set the background of the component to high intensity.
BACKGROUND-LOW	boolean	boolean	Set the background of the component to low intensity.
BACKGROUND-STANDARD	boolean	boolean	Set the background of the component to its standard intensity (may be low or high, dependent on system).
TRANSPARENT	boolean	boolean	Set the background of the component to invisible, allowing anything beneath it to show through. This is often used in conjunction with background images.
BELL	boolean	boolean	Request that the component sound a bell when

Name	MODIFY	INQUIRE	Descriptions
			displayed/accepted. Generally, this request is ignored by the components.
BEEP	boolean	boolean	Request that the component sound a bell when displayed/accepted. Generally, this request is ignored by the components.

Mid Level Common Properties

These properties should generally be avoided. They are provided to offer a lower-level of control for more power and flexibility.

Name	MODIFY	INQUIRE	Descriptions
CURRENT-WIDTH	N/A	integer	Obtain the current width in pixels of the component.
CURRENT-HEIGHT	N/A	integer	Obtain the current height in pixels of the component.
CURRENT-X	N/A	integer	Obtain the horizontal location of the component in pixels from the top-left of its containing window.
CURRENT-Y	N/A	integer	Obtain the vertical location of the component in pixels from the top-left of its containing window.
TOOLTIP-TEXT	text	text	Register text to display over a component when the mouse pointer lingers over a component. Most components automatically generate tooltips, but this may be used to generate more explicit contextual help.
VISIBLE-X	N/A	integer	Obtain the component's visible rectangle, the intersection of its position and size, with its ancestors; this property returns the horizontal X position in pixels.
VISIBLE-Y	N/A	integer	Obtain the component's visible rectangle, the intersection of its position and size, with its ancestors; this property returns the vertical Y position in pixels.
VISIBLE-WIDTH	N/A	integer	Obtain the component's visible rectangle, the intersection of its position and size, with its ancestors; this property returns the width in pixels.
VISIBLE-HEIGHT	N/A	integer	Obtain the component's visible rectangle, the intersection of its position and size, with its ancestors; this property returns the height in

			pixels.
GRAB-FOCUS	boolean	boolean	Immediately take the focus to the component. Use request-focus instead.
DOUBLE-BUFFERED	N/A	boolean	Is this component using a buffer for painting?
FOCUS-CYCLE-ROOT	N/A	boolean	Is this component providing its own focus cycle?
FOCUS-TRAVERSABLE	N/A	boolean	Is this component focus traversable using tabs?
MANAGING-FOCUS	N/A	boolean	Is this component managing its own focus?
OPAQUE	boolean	boolean	Set this component to paint all of its pixels, including its background pixels.
OPTIMIZED-DRAWING-ENABLED	N/A	boolean	Is this component tiling its children, guaranteeing that they do not overlap?
PAINTING-TILE	N/A	boolean	Is this component currently painting a tile?
REQUEST-FOCUS-ENABLED	N/A	boolean	Can this component obtain focus by setting request-focus?
VALIDATE-ROOT	N/A	boolean	Is this component the root of a validation tree?
REMOVE-NOTIFY	N/A	N/A	Notify a component that it no longer has a parent container.
REQUEST-FOCUS	boolean	boolean	Request that focus be transferred to this component.
REQUEST-DEFAULT-FOCUS	boolean	boolean	Request that the component should have focus by default.
REVALIDATE	N/A	N/A	Request that the component be revalidated.
AUTOSCROLLS	N/A	boolean	AUTOSCROLLS components automatically scroll contents when dragged.
DEBUG-GRAPHICS-OPTIONS	N/A	boolean	Enable diagnostic information about the component.
INSETS-BOTTOM	N/A	integer	A component has insets, a blank area at the edges within the component to space it away from other components. This obtains the bottom inset.
INSETS-LEFT	N/A	integer	A component has insets, a blank area at the edges within the component to space it away from other components. This obtains the bottom inset.
INSETS-RIGHT	N/A	integer	A component has insets, a blank area at the edges within the component to space it away from other components. This obtains the bottom inset.
INSETS-TOP	N/A	integer	A component has insets, a blank area at the edges within the

			component to space it away from other components. This obtains the bottom inset.
UPDATE-UI	N/A	N/A	Reset the appearance of the component according to the current look and feel.

Low Level Common Properties

These properties should generally be avoided. They are provided to offer a lower-level of control for more power and flexibility.

Name	MODIFY	INQUIRE	Descriptions
DO-LAYOUT		N/A	Prompt the layout manager to lay out this component. This method is applied automatically.
ALIGNMENT-X	float		Alignment along the X axis, specifying how the component would like to be aligned relative to other components. This value is anywhere between 0 and 1, where 0 represents alignment along the origin, 1 furthest from origin, .5 centered.
ALIGNMENT-Y	float		Alignment along the Y axis, specifying how the component would like to be aligned relative to other components. This value is anywhere between 0 and 1, where 0 represents alignment along the origin, 1 furthest from origin, .5 centered.
BACKGROUND-RGB	integer		Set the background color of the component to an integer RGB value. Blue is in bits 0-7, green is in bits 8-15, red is in bits 16-23.
FOREGROUND-RGB	integer		Set the foreground color of the component to an integer RGB value. Blue is in bits 0-7, green is in bits 8-15, red is in bits 16-23.
CURSOR-TYPE	integer		Set the cursor type to one of the following cursor numbers: Name Number DEFAULT_CURSOR 0 CROSSHAIR_CURSOR 1 TEXT_CURSOR 2 WAIT_CURSOR 3 SW_RESIZE_CURSOR 4 SE_RESIZE_CURSOR 5 NW_RESIZE_CURSOR 6 NE_RESIZE_CURSOR 7 N_RESIZE_CURSOR 8 S_RESIZE_CURSOR 9 W_RESIZE_CURSOR 10 E_RESIZE_CURSOR 11 HAND_CURSOR 12

Name	MODIFY	INQUIRE	Descriptions
			MOVE_CURSOR 13
DO-INVALIDATE		N/A	Invalidate the component, causing it and all of its parents to be marked layout.
DO-VALIDATE		N/A	Ensure that the component has a valid layout.
DO-REPAINT		N/A	Visually repaint the component as soon as possible.
VALID	N/A	boolean	Is the component correctly sized and positioned within its parent container with all of its children also valid?
DO-LIST		N/A	Print a listing of the component's state to standard out.
VISIBLE	boolean		Show or hide the component.
LOCATION-X	N/A	integer	Obtain the horizontal location of the component in pixels from the top-left of its containing window.
LOCATION-Y	N/A	integer	Obtain the vertical location of the component in pixels from the top-left of its containing window.
LOCATION-X-ON-SCREEN	N/A	integer	Obtain the horizontal location of the component in pixels from the top-left of its screen.
LOCATION-Y-ON-SCREEN	N/A	integer	Obtain the vertical location of the component in pixels from the top-left of its screen.
MAXIMUM-WIDTH	N/A	integer	Obtain the maximum width in pixels desired by the component.
MAXIMUM-HEIGHT	N/A	integer	Object the maximum height in pixels desired by the component.
MINIMUM-WIDTH	N/A	integer	Obtain the minimum width in pixels desired by the component.
MINIMUM-HEIGHT	N/A	integer	Obtain the minimum height in pixels desired by the component.
PREFERRED-WIDTH	N/A	integer	Obtain the preferred width in pixels desired by the component.
PREFERRED-HEIGHT	N/A	integer	Obtain the preferred height in pixels desired by the component.
SIZE-WIDTH	N/A	integer	Obtain the current width in pixels of the component.
SIZE-HEIGHT	N/A	integer	Obtain the current height in pixels of the component.

Events

Events are issued by components. They are grouped into three categories: command, notification and message. Command events are prefixed by CMD-, notification by NTF-, and message by MSG-.

The following events may be issued by components.

Events Window Table

Name	Value	Description
CMD-CLOSE	1	User selected close window.
CMD-ACTIVATE	6	User activated window.
NTF-RESIZED	4114	User resized window. VENT-DATA-1 is new height. VENT-DATA-2 is new width
MSG-CLOSE	16415	. User selected to close program.

Events Component Table

Name	Value	Description
CMD-GOTO	3	User selected component for focus.
CMD-CLICKED	4	User clicked component.
CMD-DBLCLICK	5	User double-clicked component (if NOTIFY-DBLCLICK).
CMD-TABCHANGED	7	User selects different tab. EVENT-DATA-1 is selected tab.
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
NTF-CHANGED	4100	User changed entry field (if NOTIFY-CHANGE). EVENT-DATA-1 is position of cursor (based at 1).
NTF-PL-NEXT	4101	PAGED LIST-BOX; user scrolls one record down.
NTF-PL-PREV	4102	PAGED LIST-BOX; user scrolls one record up.
NTF-PL-NEXTPAGE	4103	PAGED LIST-BOX; user scrolls one page down.
NTF-PL-PREVPAGE	4104	PAGED LIST-BOX; user scrolls one page up.
NTF-PL-FIRST	4105	PAGED LIST-BOX; user scrolls to top.
NTF-PL-LAST	4106	PAGED LIST-BOX; user scrolls to bottom.
NTF-PL-SEARCH	4107	PAGED LIST-BOX: user requests search. EVENT-DATA-1 contains length of search text. INQUIRE SEARCH-TEXT contains search text.
NTF-SELCHANGE	4099	User selected item (if NOTIFY-SELCHANGE). EVENT-DATA-1 is selected item (based at 1).
MSG-SB-NEXT	16385	User clicked down/right unit.
MSG-SB-PREV	16386	User clicked up/left unit.
MSG-SB-NEXTPAGE	16387	User clicked down/right page.
MSG-SB-PREVPAGE	16388	User clicked up/left page.
MSG-SB-THUMB	16389	User repositioned thumb control. EVENT-DATA-2 is new position.
MSG-SB-THUMBTRACK	16390	User respositioning thumb (if TRACK-THUMB). EVENT-DATA-2 is new position.
MSG-VALIDATE	16391	Allow program to validate data entered.
MSG-BEGIN-ENTRY	16392	N/A (GRID).
MSG-FINISH-ENTRY	16393	N/A (GRID).
MSG-CANCEL-ENTRY	16394	N/A (GRID).
MSG-GOTO-CELL	16395	N/A (GRID).
MSG-GOTO-CELL-MOUSE	16396	N/A (GRID).
MSG-BITMAP-CLICKED	16400	N/A (GRID).

Name	Value	Description
MSG-BITMAP-DBLCLICK	16401	N/A (GRID).
MSG-HEADING-CLICKED	16402	N/A (GRID).
MSG-HEADING-DBLCLICK	16403	N/A (GRID).
MSG-GOTO-CELL-DRAG	16404	N/A (GRID).
MSG-HEADING-DRAGGED	16405	N/A (GRID).
MSG-BEGIN-DRAG	16406	N/A (GRID).
MSG-END-DRAG	16407	N/A (GRID).
MSG-BEGIN-HEADING-DRAG	16408	N/A (GRID).
MSG-END-HEADING-DRAG	16409	N/A (GRID).
MSG-COL-WIDTH-CHANGED	16410	N/A (GRID).
MSG-TV-SELCHANGING	16411	TREE-VIEW selection about to change. EVENT-DATA-1 is reason for change. EVENT-DATA-2 is ID of item to be selected.
MSG-TV-SELCHANGE	16412	TREE-VIEW selection has changed. EVENT-DATA-1 is reason for change. EVENT-DATA-2 is ID of item selected.
MSG-TV-EXPANDING	16413	TREE-VIEW item about to expand/collapse. EVENT-DATA-1 is TVFLAG-EXPAND (2) or TVFLAG-COLLAPSE (1). EVENT-DATA-2 is ID of item.
MSG-TV-EXPANDED	16414	TREE-VIEW item has expanded/collapsed. EVENT-DATA-1 is TVFLAG-EXPAND (2) or TVFLAG-COLLAPSE (1). EVENT-DATA-2 is ID of item.
MSG-SPIN-UP	16416	User clicked up spinner.
MSG-SPIN-DOWN	16417	User clicked down spinner.
MSG-PAGED-NEXT	16419	N/A (GRID).
MSG-PAGED-PREV	16420	N/A (GRID).
MSG-PAGED-NEXTPAGE	16421	N/A (GRID).
MSG-PAGED-PREVPAGE	16422	N/A (GRID).
MSG-PAGED-FIRST	16423	N/A (GRID).
MSG-PAGED-LAST	16424	N/A (GRID).
MSG-GRID-RBUTTON-DOWN	16426	N/A (GRID).
MSG-GRID-RBUTTON-UP	16427	N/A (GRID).
MSG-TV-DBLCLICK	16428	User double-clicked item with no children. EVENT-DATA-2 is ID of item.
MSG-WB-BEFORE-NAVIGATE	16429	N/A. When about to navigate to new URL.
MSG-WB-DOWNLOAD-BEGIN	16431	N/A. When about to download.

Name	Value	Description
MSG-WB-DOWNLOAD-COMPLETE	16432	N/A. When download is complete
MSG-WB-NAVIGATE-COMPLETE	16430	N/A. When navigation is complete.
MSG-WB-PROGRESS-CHANGE	16433	N/A. When progress amount is changed
MSG-WB-STATUS-TEXT-CHANGE	16434	N/A. When the status text is changed.

Events Menu Table

Name	Value	Description
MSG-INIT-MENU	16398	N/A. Component pop-up to be displayed. EVENT-DATA-2 is menu handle.
MSG-MENU-INPUT	16397	N/A. User selected item from pop-up. EVENT-DATA-2 is menu ID.
MSG-END-MENU	16399	N/A. Pop-up is no longer visible. EVENT-DATA-2 is menu handle.

LABEL

Synonym: TEXT-LABEL

General information

Labels display text in a graphical manner to the user. The label is the graphical counterpart to non-acceptable text screen section data.

Labels use the TITLE property for the text; they do not use values. A label with a value defaults to the natural size of the label. A label may occupy multiple lines on the screen, but defaults to one line.

The LINES scale is the height of the font. The SIZE scale is the width of '0' in the given font.

A LABEL uses both foreground- and background-color if specified. If not specified, the background is the background color of the window.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
LEFT	boolean	boolean	Left-align the text within the label's specified area.
RIGHT	boolean	boolean	Right-align the text within the label's specified area.
CENTER	boolean	boolean	Center-align the text within the label's specified area.
CENTERED	boolean	boolean	Center-align the text within the label's specified area.
NO-KEY-LETTER	boolean	boolean	Ignore any implied or specified key letter. Otherwise, the ampersand (&) character may precede the requested hot-key.

			(The ampersand hot-key is supported on all graphics platforms, not just Windows from which it originates.)
LABEL-OFFSET	integer	integer	Allow the label to be shifted down the screen by the given amount, especially useful for aligning labels with textfields. The units are hundredths of a line, so the value 20 (default) is 0.20 line.

STATUS-BAR

The STATUS-BAR may appear at the bottom of the window. A STATUS-BAR may only be created in a DISPLAY, not the SCREEN SECTION.

A STATUS-BAR has no SIZE or LINES or position. It automatically places itself at the bottom of the window, using the entire width of the window.

A STATUS-BAR has several 'panels' which holds text and can be visually distinguished.

TITLE is synonymous with PANEL-TEXT.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
GRIP	boolean	boolean	Include a resizing GRIP in the corner of the window. No Java look and feel supports this feature, so it is currently syntax checked only.
PANEL-WIDTHS	integer	integer	Set the width of each panel in characters in turn.
PANEL-STYLE	integer	integer	Set the style of each panel in turn to one of the following styles: Value Meaning 0 Flat 1 Lowered 2 Raised
PANEL-TEXT	integer	integer	Set the text of each panel in turn.
PANEL-INDEX	integer	integer	Set the current panel index to the given integer.

ENTRY-FIELD

General information

Entry-fields accept text from the user. This component may also be known as a textfield or textbox. The entry-field is the graphical counterpart to acceptable text screen section entry.

Entry-fields do not use the TITLE property. They display and accept the VALUE property. They accept the VALUE IS MULTIPLE table method of settings values; this results in a multi-line entry-field, one line per table element.

The LINES scale is the height of the font. The SIZE scale is the width of '0' in the given font; this may be modified by the EF-UPPER-WIDE configuration variable in pixels for wide sizes. Wide sizes are used for the UPPER style or when the size is less than the configuration variable EF-WIDE-SIZE. This modification is to accommodate the extra visual overhead of entry-fields; in large fields, the overhead is a small percentage, but in small fields or fields with all uppercase, the overhead if a large percentage of the visual display.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
NUMERIC	boolean	boolean	Accept only numeric data.
NO-BOX	boolean	boolean	Do not display the box around the entry-field. Avoid this style if possible as it does not appear graphical and does not match other system applications.
BOXED	boolean	boolean	Display a box around the entry-field (default).
LEFT	boolean	boolean	Left-justify text within the entry-field (default).
RIGHT	boolean	boolean	Right-justify text within the entry-field.
CENTER	boolean	boolean	Center-justify text within the entry-field.
CENTERED	boolean	boolean	Center-justify text within the entry-field.
MULTILINE	boolean	boolean	Allow LINES to be greater than one, allowing multiple lines of text to be displayed and entered.
VSCROLL	boolean	boolean	Allow user to vertically scroll in a multi-line entry-field.
VSCROLL-BAR	boolean	boolean	Allow user to vertically scroll in a multi-line entry-field and place a vertical scroll bar within the entry-field to help the user do so.
USE-RETURN	boolean	boolean	Allow user to use the return/enter key within the

Name	MODIFY	INQUIRE	Descriptions
			entry-field rather than allowing it to terminate the accept.
USE-TAB	boolean	boolean	Allow user to use the tab key within the entry-field rather than allowing it to tab to the next component.
LOWER	boolean	boolean	Convert all text to lower-case.
UPPER	boolean	boolean	Convert all text to upper-case.
NO-AUTOSEL	boolean	boolean	Normally, a field's contents are automatically selected when the user clicks on the entry-field. This inhibits the automatic selection behavior.
READ-ONLY	boolean	boolean	Normally, a user may modify the text within an entry-field. The read-only style prevents the user from making changes to the entry-field. On many systems, read-only entry-fields are displayed differently, such as with a gray background as opposed to a white background.
SECURE	boolean	boolean	Allow secure entry into the entry-field. This is typically done for passwords or other sensitive data. An asterisk (*) is displayed in place of typed characters.
SPINNER	boolean	boolean	Include a spinner within the text-field. This spinner control includes an up and down arrow for the user to click and 'spin' the value of the entry-field. This generates the MSG-SPIN-UP and MSG-SPIN-DOWN events, but does not automatically change the value of the entry-field; this is left to user code.
AUTO-SPIN	boolean	boolean	This is like SPINNER, but does automatically change the value of the entry-field. The change in value will be checked against the MIN-VAL and MAX-VAL properties.
AUTO	boolean	boolean	Cause the entry-field to automatically terminate its input when filled with content by the user.
AUTOTERMINATE	boolean	boolean	Cause the entry-field to automatically terminate its input when filled with content by the user.

Name	MODIFY	INQUIRE	Descriptions
NOTIFY-CHANGE	boolean	boolean	Cause the entry-field to generate NTF-CHANGED events when changed by the user.
MAX-TEXT	integer	integer	Set the maximum number of characters which may be entered by the user.
MAX-LINES	integer	integer	Set the maximum number of lines which may be entered in a multi-line entry-field.
CURSOR	integer	integer	Sets/gets the cursor position within the entry-field.
CURSOR-COL	integer	integer	Sets/gets the cursor column position within the multi-line entry-field.
CURSOR-ROW	integer	integer	Sets/gets the cursor row position within the multi-line entry-field.
ACTION	integer	integer	Do a specific action on the entry-field. The integer must be one of the following: ACTION-CUT ACTION-COPY ACTION-PASTE ACTION-DELETE
DO-CUT		N/A	Cut the currently selected text.
DO-COPY		N/A	Copy the currently selected text to the clipboard.
DO-PASTE		N/A	Paste text from the clipboard to the currently selected text or cursor.
DO-DELETE		N/A	Delete the currently selected text.
MIN-VAL	integer	integer	Set the minimum value for spinning.
MAX-VAL	integer	integer	Set the maximum value for spinning.
SELECTION-TEXT	text		Replaces or gets the currently selected text.
AUTO-DECIMAL	integer	integer	Specify a minimum number of digits to the right of the decimal point for autotermination.
MASK-VALUE	text		Specify a mask value, a pattern into which the user types. Examples include phone numbers and social-security numbers. In the mask, the '?' character matches any character, '#' matches numeric digits, and any other character matches itself. So, '(###) ###-####' matches a long-distance

Name	MODIFY	INQUIRE	Descriptions
			phone number and '###-##-####' matches a social security number.
BLANK-WHEN-ZERO	boolean	boolean	Display the field as blank when the value is zero.
BLANK-WHEN-ZEROES	boolean	boolean	Display the field as blank when the value is zero.
BLANK-WHEN-ZEROS	boolean	boolean	Display the field as blank when the value is zero.
JUSTIFIED	boolean	boolean	Justify the text; if true, same as RIGHT; if false, same as LEFT.
JUST	boolean	boolean	Justify the text; if true, same as RIGHT; if false, same as LEFT.
BLINK	boolean	boolean	Apply the BLINKING style. This will most often be a color, not blinking text. This style is a holdover from textual screen section support and should be avoided in graphics.
BLINKING	boolean	boolean	Apply the BLINKING style. This will most often be a color, not blinking text. This style is a holdover from textual screen section support and should be avoided in graphics.
UNDERLINE	boolean	boolean	Apply the UNDERLINE style. Text within the entry-field will be underlined. This style is a holdover from textual screen section support and should be avoided in graphics.
UNDERLINED	boolean	boolean	Apply the UNDERLINE style. Text within the entry-field will be underlined. This style is a holdover from textual screen section support and should be avoided in graphics.
REQUIRED	boolean	boolean	Ensure that the field must have some data before being allowed to terminate.
EMPTY-CHECK	boolean	boolean	Ensure that the field must have some data before being allowed to terminate.
REVERSE-VIDEO	boolean	boolean	Display the text in reverse-video, with foreground and background interchanged.
REVERSE	boolean	boolean	Display the text in reverse-video, with foreground and background interchanged.
REVERSED	boolean	boolean	Display the text in reverse-video, with foreground and background interchanged.

Name	MODIFY	INQUIRE	Descriptions
REVERSE-VIDEO	boolean	boolean	Display the text in reverse-video, with foreground and background interchanged.
INVERSE	boolean	boolean	Display the text in reverse-video, with foreground and background interchanged.
OPEN-DIALOG	-	N/A	Create an open file chooser dialog.
SAVE-DIALOG	-	N/A	Create a save file chooser dialog.
FILENAME-FIELD	-	N/A	File chooser dialog allows only filesnames.
DIRECTORY-FIELD	-	N/A	File chooser dialog allows only directories.
FILENAME-DIRECTORY-FIELD	-	N/A	File chooser dialog allows filenames or directories.
CURRENT-DIRECTORY	text	text	File chooser's current directory.
APPROVE-BUTTON-TEXT	text	text	Text for approve button of file chooser.
APPROVE-BUTTON-MNEMONIC	text	text	Mnemonic for approve button of file chooser.
APPROVE-BUTTON-TOOLTIP-TEXT	text	text	Tooltip text for approve button of file chooser.
DIALOG-TITLE	text	text	File chooser's dialog title.
BROWSE-TEXT	text	text	Text for browse button of file chooser.

Events Table

Name	Value	Description
CMD-GOTO	3	User selected component for focus.
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
MSG-SPIN-UP	16416	User clicked up spinner.
MSG-SPIN-DOWN	16417	User clicked down spinner.
MSG-VALIDATE	16391	Allow program to validate data entered.
NTF-CHANGED	4100	User changed entry field (if NOTIFY-CHANGE). EVENT-DATA-1 is position of cursor (based at 1).

GRID

Synonym: GRID-CONTROL

General information

Grids display information in a two-dimensional manner similar to a spreadsheet. The information is presented in rows of columns with optional headers giving

information about the various rows and columns. The user may input information into the cells of the grid.

The grid rows are also termed records, and a group structure with a record defined in the same way as the grid may be used to add records at a time. Different properties manage the grid as a whole, just an individual row or column, or just an individual cell. The X and Y properties are the primary settings responsible for controlling which part of the grid is used.

Headers are always visible regardless of the user's scrolling actions. All scrolling is accomplished using the current look and feel's scrolling techniques, generally a smooth scroll; this means that partial grid cells may be available.

SIZE is the number of columns, and LINES is the number of lines in the grid.

The color and font used for each portion of the grid is chosen from the most specific to the least specific. The ordering is CURSOR, REGION, CELL, HEADING, ROW, COLUMN, ROW-PATTERN, COLUMN-PATTERN.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
ADJUSTABLE-COLUMNS	boolean	boolean	User may adjust column sizes by dragging column with mouse.
BOXED	boolean	boolean	Draw a border around the grid.
CENTERED-HEADINGS	boolean	boolean	Column headings are centered regardless of alignment.
COLUMN-HEADINGS	boolean	boolean	First record is column headings.
COLUMN-HEADERS	boolean	boolean	Synonym for COLUMN-HEADINGS.
HSCROLL	boolean	boolean	Grid may scroll horizontally.
NO-BOX	boolean	boolean	Grid does not have a border.
PAGED	boolean	boolean	Grid is of the paged style, deleting records above top and below bottom.
ROW-HEADINGS	boolean	boolean	Grid has row headings in column 1.
ROW-HEADERS	boolean	boolean	Synonym for ROW-HEADINGS.
TILED-HEADINGS	boolean	boolean	Headings drawn as shaded tiles.
USE-TAB	boolean	boolean	Tab may be used within grid.
VSCROLL	boolean	boolean	Grid may scroll vertically.
ACTION	integer	N/A	Invoke grid paging, use DO-commands instead.
DO-FIRST-PAGE	-	N/A	Invoke grid first page events.
DO-LAST-PAGE	-	N/A	Invoke grid last page events.
DO-CURRENT-PAGE	-	N/A	Invoke grid current page events.
ALIGNMENT	text	N/A	'L' for left, 'R' for right, 'C' for centered, 'U' for unaligned; all but unaligned remove spaces. This sets each column in turn.
BITMAP	integer	N/A	Place bitmap handle at X, Y.
BITMAP-NUMBER	integer	N/A	Set bitmap index in bitmap strip.
BITMAP-TRAILING	integer	N/A	Set to 1 for bitmap trailing text, 0 for bitmap leading text.
BITMAP-WIDTH	integer	N/A	Set width of bitmap in pixels.
BITMAP-VALUE	text	N/A	Set bitmap by name value.
CELL-COLOR	integer	N/A	Set the cell at X,Y to color number.

Name	MODIFY	INQUIRE	Descriptions
CELL-FOREGROUND	integer	integer	Set the cell foreground at X,Y to color number.
CELL-FOREGROUND-RGB	integer	integer	Set the cell foreground at X,Y to RGB value.
CELL-BACKGROUND	integer	integer	Set the cell background at X,Y to color number.
CELL-BACKGROUND-RGB	integer	integer	Set the cell background at X,Y to RGB value.
CELL-DATA	text	text	Visible cell data at X,Y This is the value of the cell.
CELL-FONT	integer	integer	Font number for cell at X,Y.
CELL-FONT-NAME	text	N/A	Font name for cell at X,Y.
COLUMN-COLOR	integer	integer	Column X is color number.
COLUMN-FOREGROUND	integer	integer	Column X foreground is color number.
COLUMN-FOREGROUND-RGB	integer	integer	Column X foreground is RGB value.
COLUMN-BACKGROUND	integer	integer	Column X background is color number.
COLUMN-BACKGROUND-RGB	integer	integer	Column X background is RGB value.
COLUMN-DIVIDERS	integer	N/A	Set width of column divider in pixels for each column in turn.
COLUMN-FONT	integer	N/A	Column X font is font handle number.
COLUMN-FONT-NAME	text	N/A	Column X font is font name.
CURSOR-COLOR	integer	integer	Cursor color.
CURSOR-FOREGROUND	integer	integer	Cursor foreground color.
CURSOR-FOREGROUND-RGB	integer	integer	Cursor foreground RGB value.
CURSOR-BACKGROUND			Cursor background color.
CURSOR-BACKGROUND-RGB			Cursor background RGB value.
EDITABLE	boolean	boolean	Cell at X,Y is editable.
EDITABLE-COLOR	integer	integer	Color of editing cell.
EDITABLE-FOREGROUND	integer	integer	Foreground color of editing cell.
EDITABLE-FOREGROUND-RGB	integer	integer	Foreground RGB value of editing cell.
EDITABLE-BACKGROUND	integer	integer	Background color of editing cell.
EDITABLE-BACKGROUND-RGB	integer	integer	Background RGB value of editing cell.
EDIT-CELL	-	N/A	Start editing at X,Y.
EDITING-COLUMN	integer	integer	Column being edited.
EDITING-ROW	integer	integer	Row being edited.
EDITING	N/A	boolean	1 if user editing, 0 if not.
CELL-SELECTED	N/A	boolean	1 if cells selected, 0 if not.
SELECTED-ROW-COUNT	N/A	integer	Number of selected rows.
SELECTED-COLUMN-	N/A	integer	Number of selected columns.

Name	MODIFY	INQUIRE	Descriptions
COUNT			
SELECT-ALL	-	N/A	Select all grid cells.
CLEAR-ALL	-	N/A	Clear all grid cells' selection.
CURSOR-FRAME-WIDTH	integer	integer	Thickness in pixels of cursor border.
CURSOR-X	integer	integer	Cursor X position .
CURSOR-Y	integer	integer	Cursor Y position.
DATA-COLUMNS	integer	integer	Internal storage record offset position within record for each column in turn. Each column number is given in turn.
DATA-SIZES	integer	integer	Data-sizes is preferred form.
DATA-TYPES	text	N/A	Internal storage size of each column within record for each column in turn.
DISPLAY-COLUMNS	integer	integer	Externally visible positioning within record for each column in turn. Generally this is at least as wide as DATA-SIZES or sizes implied by DATA-COLUMNS.
DIVIDER-COLOR	integer	integer	Color of row and column dividers.
DIVIDER-COLOR-RGB	integer	integer	RGB value of row and color dividers.
DIVIDER-FOREGROUND	integer	integer	Foreground color of row and column dividers.
DIVIDER-FOREGROUND-RGB	integer	integer	RGB value of row and column dividers.
END-COLOR	integer	integer	Color of end of grid where no data is available.
END-COLOR-RGB	integer	integer	RGB value of end of grid where no data is available.
END-BACKGROUND	integer	integer	Background color of end of grid.
END-BACKGROUND-RGB	integer	integer	RGB background value of end of grid.
FILE-POS	integer	integer	Only for paged grids, record number matching file position. Special values are PAGED-AT-START (2147418113), PAGED-AT-END (2147418114), PAGED-EMPTY (214748115).
HEADING-COLOR	integer	integer	Heading color value.
HEADING-BACKGROUND	integer	integer	Heading background color value.
HEADING-BACKGROUND-RGB	integer	integer	Heading RGB background value.
HEADING-FOREGROUND	integer	integer	Heading foreground color value.
HEADING-FOREGROUND-RGB	integer	integer	Heading RGB foreground value.
HEADING-DIVIDER-COLOR	integer	integer	Heading divider color, separating one header from another.
HEADING-DIVIDER-COLOR-RGB	integer	integer	Heading divider RGB value, separating one header from another.

Name	MODIFY	INQUIRE	Descriptions
HEADING-FONT	integer	integer	Heading font handle number.
HEADING-FONT-NAME	text	text	Heading font name.
HIDDEN-DATA	text	text	Hidden data in cell at X,Y.
HSCROLL-POS	N/A	integer	Horizontal scroll position, the leftmost visible column.
INSERT-ROWS	integer	N/A	Insert given number of blank records at INSERTION-INDEX, then set INSERTION-INDEX to 0.
INSERTION-INDEX	integer	integer	Row number for new records, 0 for end.
LAST-ROW	integer	integer	Record number of last non-blank record.
MASS-UPDATE	integer	integer	0 for instant update, 1 for massed updates.
NUM-COL-HEADINGS	integer	integer	Number of column headings can be 0 or 1.
NUM-ROW-HEADINGS	integer	integer	Number of row headings can be 0 or 1.
NUM-ROWS	integer	integer	Number of rows in the grid.
NUM-COLUMNS	integer	integer	Number of columns in the grid, may be implicitly defined by other properties.
RECORD-DATA	text	text	Using DATA-COLUMNS or DATA-SIZES, add an entire record of data at the Y row.
RECORD-TO-ADD	text	text	Using DATA-COLUMNS or DATA-SIZES, add an entire record of data at INSERTION-INDEX.
RECORD-TO-DELETE	integer	integer	Delete given row of data.
REGION-COLOR	integer	integer	Set color for region from (START-X,START-Y) through (X,Y). This is generally handled automatically.
REGION-FOREGROUND	integer	integer	Region foreground color.
REGION-FOREGROUND-RGB	integer	integer	Region RGB foreground value.
REGION-BACKGROUND	integer	integer	Region background color.
REGION-BACKGROUND-RGB	integer	integer	Region RGB background value.
RESET-GRID	integer	N/A	Set to non-zero to clear data and attributes.
RESET	integer	N/A	Synonym for RESET-GRID
ROW-COLOR	integer	integer	Color for row Y.
ROW-FOREGROUND	integer	integer	Foreground color for row Y.
ROW-FOREGROUND-RGB	integer	integer	Foreground RGB value for row Y.
ROW-BACKGROUND	integer	integer	Background color for row Y.
ROW-BACKGROUND-RGB	integer	integer	Background RGB value for row Y.
RECORD-COLOR	integer	integer	Synonym for RECORD-COLOR
RECORD-FOREGROUND	integer	integer	Synonym for ROW-FOREGROUND
RECORD-FOREGROUND-RGB	integer	integer	Synonym for ROW-FOREGROUND-RGB

Name	MODIFY	INQUIRE	Descriptions
RECORD-BACKGROUND	integer	integer	Synonym for ROW-BACKGROUND
RECORD-BACKGROUND-RGB	integer	integer	Synonym for ROW-BACGROUND-RGB
ROW-COLOR-PATTERN	integer	N/A	Apply repeatedly to create color pattern for rows.
ROW-FOREGROUND-PATTERN	integer	N/A	Apply repeatedly to create foreground color pattern for rows.
ROW-FOREGROUND-RGB-PATTERN	integer	N/A	Apply repeatedly to create foreground RGB value pattern for rows.
ROW-BACKGROUND-PATTERN	integer	N/A	Apply repeatedly to create background color pattern.
ROW-BACKGROUND-RGB-PATTERN	integer	N/A	Apply repeatedly to create background RGB value pattern.
ROW-DIVIDERS	integer	N/A	Apply repeatedly to establish row divider thickness in pixels.
ROW-FONT	integer	integer	Font handle number for row Y.
ROW-FONT-NAME	text	text	Font name for row Y.
SEARCH-OPTIONS	text	N/A	Set to group to define how searches are performed.
SEARCH-TEXT	text	integer	Assign text to this to perform search using SEARCH-OPTIONS. X and Y return the location if successful. This returns a 0 for no data found, 1 for success, 2 for success with wrap.
SEPARATION	integer	integer	Amount of whitespace between columns in tenths of character
START-X	integer	integer	Starting column for selection.
START-Y	integer	integer	Starting row for selection.
VPADDING	integer	integer	Vertical whitespace as integer percentage to add to row height.
VSCROLL-POS	integer	integer	Uppermost visible record in grid.
X	integer	integer	Column position for other properties to use.
Y	integer	integer	Row position for other properties to use.
DRAG-EVENTS	boolean	boolean	Enable or disable events ending in _DRAG, by default enabled.

Events Table

Name	Value	Description
CMD-GOTO	3	User selected component for focus.
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
MSG-PAGED-NEXT	16419	User clicked next button on paged grid. EVENT-DATA-2 is page-pos.
MSG-PAGED-PREV	16420	User clicked previous button on paged grid. EVENT-DATA-2 is page-pos.
MSG-PAGED-NEXTPAGE	16421	User clicked next page button on paged grid.

Name	Value	Description
MSG-PAGED-PREVPAGE	16422	User clicked previous page button on paged grid.
MSG-PAGED-FIRST	16423	User clicked first page button on paged grid.
MSG-PAGED-LAST	16424	User clicked last page button on paged grid.
MSG-COL-WIDTH-CHANGED	16410	Column width has changed. EVENT-DATA-1 is column, EVENT-DATA-2 is new width.
MSG-GOTO-CELL	16395	User clicked or keyboarded to cell.
MSG-GOTO-CELL-DRAG	16404	User drag selected to cell.
MSG-BEGIN-DRAG	16406	User beginning drag operation.
MSG-END-DRAG	16407	User ending drag operation.
MSG-BEGIN-ENTRY	16392	User begins entry into cell. EVENT-DATA-1 is column, EVENT-DATA-2 is row.
MSG-FINISH-ENTRY	16393	User ends entry into cell.
MSG-CANCEL-ENTRY	16394	User entry canceled. This event is not sent by most Java systems.
MSG-GRID-RBUTTON-DOWN	16426	Right button pressed in grid.
MSG-GRID-RBUTTON-UP	16427	Right button released in grid.
MSG-HEADING-CLICKED	16402	Heading clicked. EVENT-DATA-1 is column, EVENT-DATA-2 is row.
MSG-HEADING-DBLCLICK	16403	Heading double-clicked. EVENT-DATA-1 is column, EVENT-DATA-2 is row.

The SEARCH-OPTIONS group structure is:

```
01 SEARCH-OPTIONS.
   05 SO-DIRECTION PIC 9.
      88 SO-FORWARDS VALUE 0.
      88 SO-BACKWARDS VALUE 1.
   05 SO-WRAP PIC 9.
      88 SO-DO-WRAP VALUE 0.
      88 SO-NO-WRAP VALUE 1.
   05 SO-CASE PIC 9.
      88 SO-CASE-INDEPENDENT VALUE 0.
      88 SO-CASE-DEPENDENT VALUE 1.
   05 SO-MATCH PIC 9.
     88 SO-ANY VALUE 0.
      88 SO-LEADING VALUE 1.
      88 SO-ALL VALUE 2.
   05 SO-LOCATION PIC 9.
      88 SO-VISIBLE-VALUES VALUE 0.
      88 SO-HIDDEN-VALUES VALUE 1.
      88 SO-ALL-VALUES VALUE 2.
   05 SO-SKIP PIC 9.
      88 SO-SKIP-CURRENT-CELL VALUE 0.
      88 SO-SEARCH-CURRENT-CELL VALUE 1.
   05 SO-CURSOR PIC 9.
      88 SO-MOVE-CURSOR-TO-RESULT VALUE 0.
      88 SO-RETAIN-CURSOR VALUE 1.
   05 SO-COLUMN PIC 9(5).
      88 SO-ALL-COLUMNS VALUE 0.
```

PUSH-BUTTON

A push-button is a graphical button, which the user may push to select, such as an OK or Cancel button.

Push-buttons generate events when clicked, or generate termination- or exception-values to terminate the accept.

The TITLE property determines the text on the face of the button. Push-buttons do not use the VALUE property.

The LINES scale is the height of the font. The SIZE scale is the width of '0' in the given font. When the BITMAP style is used, LINES and SIZE are in pixels instead of control units.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
DEFAULT-BUTTON	boolean	boolean	Set the button to be the default button for the window. This is the button which will be pushed when pressing the Enter/Return key, keycode 13. The default button may be displayed differently by the graphical look and feel.
ESCAPE-BUTTON	boolean	boolean	Set the button to correspond to the escape key, keycode 27.
NO-AUTO-DEFAULT	boolean	boolean	Prevent the button from becoming the default button upon activation.
OK-BUTTON	boolean	boolean	The OK-BUTTON style is a group of settings corresponding to TITLE "OK", DEFAULT-BUTTON, TERMINATION-VALUE 13.
CANCEL-BUTTON	boolean	boolean	The CANCEL-BUTTON style is a group of settings corresponding to TITLE "Cancel", ESCAPE-BUTTON, EXCEPTION-VALUE 27.
BITMAP	boolean	boolean	Display the button's face using a bitmap rather than the title text.
FRAMED	boolean	boolean	Only for BITMAP buttons, draw a thin frame around the button.
UNFRAMED	boolean	boolean	Only for BITMAP buttons, do not draw a thin frame around the button.
SQUARE	boolean	boolean	Only for BITMAP FRAMED buttons, force square corners.
FLAT	boolean	boolean	Only for BITMAP buttons, have no visible borders.
BITMAP-NUMBER	integer	integer	Identify the tile within the bitmap to display as the face.

Name	MODIFY	INQUIRE	Descriptions
BITMAP-HANDLE	integer	integer	Identify the bitmap to display for the face of the button.
BITMAP-VALUE	text	N/A	Identify the bitmap resource by name rather than by handle; any graphical bitmap resource is acceptable for the text value.

Events Table

Name	Value	Description
CMD-GOTO	3	User selected component for focus.
CMD-CLICKED	4	User clicked component.
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
MSG-VALIDATE	16391	Allow program to validate data entered.

CHECK-BOX

A check-box is a button which may be checked or unchecked. Visually, it may be represented by an actual checkmark, an 'X' character, filled-in oval, etc. A check-box is generally an independent component. For grouped check-boxes, see RADIO-BUTTON.

A TITLE describing the check-box may be included; it will be displayed next to the check-box. The value of the check-box is its checked state; a 0 represents unchecked, a 1 represents checked.

The LINES scale is the height of the font. The SIZE scale is the width of '0' in the given font. When the BITMAP style is used, LINES and SIZE are in pixels instead of control units.

A CHECK-BOX uses both foreground- and background-color if specified. If not specified, the background is the background color of the window.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
BITMAP	boolean	boolean	Display the check-box's face using a bitmap rather than the title text.
FRAMED	boolean	boolean	Only for BITMAP check-boxes, draw a thin frame around the check-box.
UNFRAMED	boolean	boolean	Only for BITMAP check-boxes, do not draw a thin frame around the check-box.
SELF-ACT	boolean	boolean	Only for BITMAP FRAMED check-boxes, force square corners.
NOTIFY	boolean	boolean	
SQUARE	boolean	boolean	

Name	MODIFY	INQUIRE	Descriptions
LEFT-TEXT	boolean	boolean	Display title text to the left rather than the right of the check-box.
FLAT	boolean	boolean	Only for BITMAP check-boxes, have no visible borders.
BITMAP-NUMBER	integer	integer	Identify the tile within the bitmap to display as the face.
BITMAP-HANDLE	integer	integer	Identify the bitmap to display for the face of the button.
BITMAP-VALUE	text	N/A	Identify the bitmap resource by name rather than by handle; any graphical bitmap resource is acceptable for the text value.
TERMINATION-VALUE	integer	integer	
EXCEPTION-VALUE	integer	integer	

Events Table

Name	Value	Description
CMD-GOTO	3	User selected component for focus.
CMD-CLICKED	4	User clicked component.
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
MSG-VALIDATE	16391	Allow program to validate data entered.

RADIO-BUTTON

A radio-button is a button which may be selected or unselected. Unlike a check-box, the radio-button is generally grouped and has only one selected member at a time; this makes it useful for selecting features exclusive of one another.

A TITLE describing the radio-button may be included; it will be displayed next to the radio-button. The value of the radio-button is its selected state; a 0 represents unchecked, a 1 represents checked.

The LINES scale is the height of the font. The SIZE scale is the width of '0' in the given font. When the BITMAP style is used, LINES and SIZE are in pixels instead of control units.

A RADIO-BUTTON uses both foreground- and background-color if specified. If not specified, the background is the background color of the window.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
BITMAP	text	N/A	Identify the bitmap resource by name rather than by handle; any graphical bitmap resource is acceptable for the text value.

FRAMED			Only for BITMAP radio-buttons, draw a thin frame around the radio-button.
UNFRAMED			Only for BITMAP radio-buttons, do not draw a thin frame around the radio-button.
SQUARE	boolean		Only for BITMAP FRAMED radio-buttons, force square corners.
NO-GROUP-TAB	boolean	boolean	Radio-buttons in a group handle tab in a special manner, internal to the radio-button group. NO-GROUP-TAB suppresses the group's handling of the tab keys.
LEFT-TEXT	boolean	boolean	Display title text to the left rather than the right of the radio-button.
FLAT	boolean	boolean	Only for BITMAP radio-buttons, have no visible borders.
BITMAP-NUMBER	integer	integer	Identify the tile within the bitmap to display as the face.
BITMAP-HANDLE	integer	integer	Identify the bitmap to display for the face of the button.
GROUP	integer	integer	Define the group to which the radio-button belongs by number; all radio-buttons belonging to the same group are exclusive of one another, unless they are in group 0. Group 0 radio-buttons behave as if they were check-boxes. The default GROUP is 1.
GROUP-VALUE	integer	integer	The GROUP-VALUE property simplifies the handling of multiple radio-buttons. Assign each radio-button in a group a different group-value number, and assign the VALUE of each radio-button to the same identifier. The identifier named by VALUE will contain the group-value of the currently selected radio-button.
BITMAP-VALUE	text	N/A	

Events Table

Name	Value	Description
CMD-GOTO	3	User selected component for focus.
CMD-CLICKED	4	User clicked component.
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
MSG-VALIDATE	16391	Allow program to validate data entered.

Example

```
77 group-check pic 99 value 0.

screen section.
01  screen-1.
   03 radio-button "One"
      column 2, line 2,
      group = 1, group-value = 1,
      value group-check.
   03 radio-button "Two"
      column 2, line 4,
      group = 1, group-value = 2,
      value group-check.
   03 radio-button "Three"
      column 2, line 6,
      group = 1, group-value = 3,
      value group-check.
```

If group-check is 1, then "One" is selected; if 2, then "Two"; if 3, then "Three".

LIST-BOX

A LIST-BOX is a list of text selections presented in a vertically scrollable list.

The TITLE is not used. The VALUE of the LIST-BOX is its currently selected text item.

LINES is the number of text lines to show. The SIZE scale is the width of '0' in the given font.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
UNSORTED	boolean	boolean	Prevent the list items from being sorted; UNSORTED shows items in the order added.
SORTED	boolean	boolean	Show the list items in sorted order (default).
PAGED	boolean	boolean	Create a PAGED LIST-BOX, instead of a normal LIST-BOX. This style must be specified at creation time, not through MODIFY. The PAGED style generates events when scrolling beyond the visible page size, allowing very large lists to be used by dynamically modifying the LIST-BOX contents.
NO-BOX	boolean	boolean	Do not display the box around the list-box.
BOXED	boolean	boolean	Display the box around the list-box (default).

Name	MODIFY	INQUIRE	Descriptions
UPPER	boolean	boolean	
LOWER	boolean	boolean	
ENTRY-FIELD-VISIBLE	boolean	boolean	
NOTIFY-DBLCLICK	boolean	boolean	Generate double-click events when the user double-clicks an item.
NOTIFY-SELCHANGE	boolean	boolean	Generate events when the selection is changed.
NO-SEARCH	boolean	boolean	Only for PAGED LIST-BOX, suppress the search capability.
ITEM-TO-ADD	text	text	Set to the text item to add to the list-box. Repeatedly setting this property fills the list-box.
RESET-LIST	integer	integer	Empty a list-box if integer is not 0.
MASS-UPDATE	integer	integer	Setting MASS-UPDATE to 1 suppresses visual updates, allowing a massive amount of data to be added to the list-box efficiently; setting MASS-UPDATE to 0 restores normal visual update behavior.
ITEM-TO-DELETE	integer	integer	Setting ITEM-TO-DELETE to a value greater than 0 deletes the corresponding item number index from the LIST-BOX.
INSERTION-INDEX	integer	integer	When set to 0, items are added in sorted order if SORTED, or to the end if UNSORTED. When set to a positive value, items are added at the given integer index.
SEARCH-TEXT	text	text	Only for PAGED LIST-BOX, contains the search text requested by the user in the NTF-PL-SEARCH event. The program should search for the given SEARCH-TEXT and display the result in the PAGED LIST-BOX.
DATA-COLUMNS	integer	integer	Specify each data column in turn for record-oriented data displayed in a list-box, starting at column 1. Set DATA-COLUMNS to 0 to reset the list.
DISPLAY-COLUMNS	integer	integer	Specify each visual column in turn for record-oriented data displayed in a list-box, starting at column 1. Set DISPLAY-COLUMNS to 0 to reset the list.

Name	MODIFY	INQUIRE	Descriptions
ALIGNMENT	text	text	Specify each visual column alignment in turn for record-oriented data displayed in a list-box, starting at column 1. Set ALIGNMENT to SPACES to reset the list. The following alignment settings are available for each column: L Left aligned after removing spaces R Right aligned after removing spaces C Center aligned after removing spaces U Unaligned, keep all spaces
SEPARATION	integer	integer	Specify each visual column spatial separation in turn for record-oriented data displayed in a list-box, starting at column 1. Set SEPARATION to -1 to reset the list. Each separation value is 1/10th of a character.
DIVIDERS	integer	integer	Specify each visual column divider's width in turn for record-oriented data displayed in a list-box, starting at column 1. Set DIVIDERS to -1 to reset the list. Each value is a width of the divider line in pixels.
SELECTION-INDEX	integer	integer	Set/get the currently selected index within the list-box.
THUMB-POSITION	integer	integer	Set the list-box to display the line number of the item at the top of the list-box, or get the line number currently displayed at the top of the list-box. This property is not implemented.
QUERY-INDEX	integer	integer	Set the index to query for use with the ITEM-VALUE property.
ITEM-VALUE	text	text	Set or get the value of the item at the index designated by QUERY-INDEX.
SORT-ORDER	integer	integer	Only for PAGED LIST-BOX, determines the sort order as one of the following: Mnemonic PL-SORT-DEFAULT 0 PL-SORT-NATIVE-IGNORE-CASE 1

Name	MODIFY	INQUIRE	Descriptions
			PL-SORT-NONE 1 PL-SORT-NATIVE 2 PL-SORT-NATIVE-IGNORE-CASE 3

Events Table

Name	Value	Description
CMD-GOTO	3	User selected component for focus.
CMD-DBLCLICK	5	User double-clicked component
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
MSG-VALIDATE	16391	Allow program to validate data entered.
NTF-SELCHANGE	4099	User selected item (if NOTIFY-SELCHANGE). EVENT-DATA-1 is selected item (based at 1).
NTF-PL-FIRST	4105	PAGED LIST-BOX; user scrolls to top.
NTF-PL-LAST	4106	PAGED LIST-BOX; user scrolls to bottom.
NTF-PL-NEXT	4101	PAGED LIST-BOX; user scrolls one record down.
NTF-PL-PREV	4102	PAGED LIST-BOX; user scrolls one record up.
NTF-PL-NEXTPAGE	4103	PAGED LIST-BOX; user scrolls one page down.
NTF-PL-PREVPAGE	4104	PAGED LIST-BOX; user scrolls one page up.
NTF-PL-SEARCH	4107	PAGED LIST-BOX: user requests search. EVENT-DATA-1 contains length of search text. INQUIRE SEARCH-TEXT contains search text.

COMBO-BOX

A COMBO-BOX combines an ENTRY-FIELD and a LIST-BOX.

The VALUE represents the data in the ENTRY-FIELD portion. For a DROP-LIST, this represents the selected item text.

SIZE is interpreted like ENTRY-FIELD. LINES is the number of lines to show, including the ENTRY-FIELD portion.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
DROP-DOWN	boolean	boolean	Hide the list portion of the component unless the user presses the down button to reveal it (default).
STATIC-LIST	boolean	boolean	Always display the list portion of the component.

Name	MODIFY	INQUIRE	Descriptions
DROP-LIST	boolean	boolean	Same as DROP-DOWN, but the user may not enter data into the ENTRY-FIELD portion of the component
UNSORTED	boolean	boolean	Same as LIST-BOX.
SORTED	boolean	boolean	Same as LIST-BOX.
LOWER	boolean	boolean	Convert all text to lower-case.
UPPER	boolean	boolean	Convert all text to upper-case.
NOTIFY-DBLCLICK	boolean	boolean	Generate CMD-DBLCLICK events when the user double-clicks an item; this is not available in all style combinations and should be avoided.
NOTIFY-SELCHANGE	boolean	boolean	Generate NTF-SELCHANGE events when the selection is changed.
MAX-TEXT	integer	integer	Same as ENTRY-FIELD.
ITEM-TO-ADD	text		Same as LIST-BOX.
MASS-UPDATE	integer	integer	Same as LIST-BOX.
RESET-LIST	integer	integer	Same as LIST-BOX.
ITEM-TO-DELETE	integer	integer	Same as LIST-BOX.
INSERTION-INDEX	integer	integer	Same as LIST-BOX.

Events Table

Name	Value	Description
CMD-GOTO	3	User selected component for focus.
CMD-DBLCLICK	5	User double-clicked component
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
MSG-VALIDATE	16391	Allow program to validate data entered.
NTF-SELCHANGE	4099	User selected item (if NOTIFY-SELCHANGE). EVENT-DATA-1 is selected item (based at 1).

FRAME

Synonym: GROUP-FRAME

A FRAME is a visual grouping of components on the screen. This is purely a visual grouping and has no effect on the components themselves. This is also used for expressing a progress-bar component.

A FRAME may have a TITLE, giving a name to the visual grouping. A FRAME does not have a VALUE.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
HEAVY	boolean	boolean	Cause the frame's thickness to be thicker than normal.

Name	MODIFY	INQUIRE	Descriptions
VERY-HEAVY	boolean	boolean	Cause the frame's thickness to be much thicker than normal; avoid when using 3-D unless using ALTERNATE.
ALTERNATE	boolean	boolean	Cause an alternate frame appearance.
RAISED	boolean	boolean	Cause the frame to have a raised appearance, a 3-D style.
LOWERED	boolean	boolean	Cause the frame to have a lowered appearance, a 3-D style.
ENGRAVED	boolean	boolean	Cause the frame to have an engraved appearance, a 3-D style.
RIMMED	boolean	boolean	Cause the frame to have a rimmed appearance, a 3-D style.
FULL-HEIGHT	boolean	boolean	Cause the top of the frame to be at the exact given location, rather than lowered somewhat as is the default.
HIGH-COLOR	integer	integer	Specify the highlight or brighter color for 3-D styles.
LOW-COLOR	integer	integer	Specify the lowlight or darker color for 3-D styles.
FILL-COLOR	integer	integer	Set the fill color for the interior of the frame; 0 (default) indicates no fill color. A fill color is used for progress bars.
FILL-PERCENT	integer	integer	Set the fill percentage for the interior of the frame.
FILL-COLOR2	integer	integer	Set the fill color for the unfilled portion of the frame; 0 (default) indicates no fill color for the remaining portion of the progress bar. Example: At 75%, fill-color is expressed by 1, and fill-color2 is expressed by 2. [11111111111111122222]
TITLE-POSITION	integer	integer	Set the position of the visible title if TITLE is set to the following positions: 1 Top Left 2 Top Center 2 Top Right 3 Bottom Left 4 Bottom Center 5 Bottom Right

Name	MODIFY	INQUIRE	Descriptions
			6 Centered Vertically and 7 Horizontally
HORIZONTAL	boolean	boolean	Set the orientation of the progress bar to the horizontal.
VERTICAL	boolean	boolean	Set the orientation of the progress bar to the vertical.

Events Table

Name	Value	Description
CMD-GOTO	3	User selected component for focus.
CMD-DBLCLICK	5	User double-clicked component
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
MSG-VALIDATE	16391	Allow program to validate data entered.

BAR

A BAR is a graphical bar, either horizontal or vertical, which may be used for graphical drawing in a variety of widths and styles.

A BAR has neither TITLE nor VALUE.

SIZE and LINES are in window cells. If SIZE is zero, then the BAR is vertical; if LINES is zero, then the BAR is horizontal. Both SIZE and LINES cannot be zero, but one must be.

The bar is drawn in the foreground-color.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
DOTTED	boolean	boolean	Create a dotted line.
DASHED	boolean	boolean	Create a dashed line.
DOT-DASH	boolean	boolean	Create a line with alternating dots and dashes.
WIDTH	integer	integer	Specify the WIDTH of the bar in pixels.
COLORS	integer	integer	Specify the color of each pixel in the WIDTH in turn. Setting COLORS to 999 or higher resets the color list.
SHADING	integer	integer	Specify the shading of each pixel in the WIDTH in turn. The following settings are available: -2 Color 16 / Bright White -1 Brighter Color 0 Normal Color 1 Darker Color 2 Color 1 / Black

			Any invalid value resets the shading list.
POSITION-SHIFT	integer	integer	Adjust the bar down or to the right by the given number of pixels.
TRAILING-SHIFT	integer	integer	Taper the trailing end of the bar by specifying the trailing shift amount of each pixel in the WIDTH in turn. Setting TRAILING-SHIFT to 999 or higher resets the TRAILING-SHIFT list. Positive values extend the line, negative values retract the line.
LEADING-SHIFT	integer	integer	Taper the leading end of the bar by specifying the leading shift amount of each pixel in the WIDTH in turn. Setting LEADING-SHIFT to 999 or higher resets the LEADING-SHIFT list. Positive values shorten the line, negative values lengthen the line.

SCROLL-BAR

A SCROLL-BAR allows the user to scroll through a number of values. A SCROLL-BAR is typically part of a scrollable base component, but is also available separately.

A SCROLL-BAR does not use the TITLE property. The VALUE is the thumb's position within the SCROLL-BAR; the range is limited by MIN-VAL and MAX-VAL.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
HORIZONTAL	boolean	boolean	Create a HORIZONTAL SCROLL-BAR rather than a vertical SCROLL-BAR.
TRACK-THUMB	boolean	boolean	Generate MSG-SB-THUMBTRACK messages while moving the thumb control.
MIN-VAL	integer	integer	Set the minimum value for the SCROLL-BAR.
MAX-VAL	integer	integer	Set the maximum value for the SCROLL-BAR.
PAGE-SIZE	integer	integer	Set the number of conceptual scrollable elements which the SCROLL-BAR scrolls through.

Events Table

Name	Value	Description
CMD-DBLCLICK	5	User double-clicked component
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
MSG-VALIDATE	16391	Allow program to validate data entered.
MSG-SB-NEXT	16385	User clicked down/right unit.
MSG-SB-PREV	16386	User clicked up/left unit.
MSG-SB-NEXTPAGE	16387	User clicked down/right page.
MSG-SB-PREVPAGE	16388	User clicked up/left page.
MSG-SB-THUMB	16389	User repositioned thumb control. EVENT-DATA-2 is new position.
MSG-SB-THUMBTRACK	16390	User respositioning thumb (if TRACK-THUMB). EVENT-DATA-2 is new position.

SLIDER

A SLIDER is similar to a SCROLL-BAR, but is intended for direct use by the user. The SLIDER may represent its numerical range visually, and allow the user finer control over the values.

A SLIDER does not use the TITLE property. The VALUE is the thumb's position within the SLIDER; the range is limited by MIN-VAL and MAX-VAL.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
HORIZONTAL	boolean	boolean	Create a HORIZONTAL SCROLL-BAR rather than a vertical SCROLL-BAR.
TRACK-THUMB	boolean	boolean	Generate MSG-SB-THUMBTRACK messages while moving the thumb control.
MIN-VAL	integer	integer	Set the minimum value for the SCROLL-BAR.
MAX-VAL	integer	integer	Set the maximum value for the SCROLL-BAR.
PAGE-SIZE	integer	integer	Set the number of conceptual scrollable elements which the SCROLL-BAR scrolls through.
PAINT-LABELS	integer	integer	Request that the labels be visually painted; 1 is on, 0 is off.
PAINT-TICKS	integer	integer	Request that the tick marks be visually painted; 1 is on, 0 is off.
PAINT-TRACK	integer	integer	Request that the main track be visually painted; 1 is on, 0 is off.
SNAP-TO-TICKS	integer	integer	Request that the thumb automatically snap to the

			nearest tick mark.
INVERTED	integer	integer	Request that the numerical range be presented in reverse order; 1 is inverted, 0 is normal.
MAJOR-TICK	integer	integer	Set the major tick interval.
MINOR-TICK	integer	integer	Set the minor tick interval.

Events Table

Name	Value	Description
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
MSG-VALIDATE	16391	Allow program to validate data entered.
MSG-SB-NEXT	16385	User clicked down/right unit.
MSG-SB-PREV	16386	User clicked up/left unit.
MSG-SB-NEXTPAGE	16387	User clicked down/right page.
MSG-SB-PREVPAGE	16388	User clicked up/left page.
MSG-SB-THUMB	16389	User repositioned thumb control. EVENT-DATA-2 is new position.
MSG-SB-THUMBTRACK	16390	User respositioning thumb (if TRACK-THUMB). EVENT-DATA-2 is new position.

TAB-CONTROL

A TAB-CONTROL is used to visually represent a single tab of information to the user at a time, allowing more information to be presented in less space.

The TITLE property is not used. The VALUE is the currently selected tab.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
MULTILINE	boolean	boolean	Allow the tabs to occupy multiple lines. This is useful for when there is a large number of tabs.
BUTTONS	boolean	boolean	Make the tabs appear like buttons. This style is not implemented.
FIXED-WIDTH	boolean	boolean	Force each tab to use the same width.
TAB-TO-ADD	text	text	Set this to each tab name to add in turn.
TAB-TO-DELETE	integer	integer	Set this to the index of the tab to delete.
RESET-TABS	integer	integer	Set RESET-TABS to non-zero to reset the tab list.
BITMAP-HANDLE	integer	integer	Set the bitmap resource to use by handle number. A single bitmap includes fixed width tiles within it, each tile capable

			of being displayed as the tab's graphic representation.
BITMAP-WIDTH	integer	integer	Set the width of each tile in the bitmap.
BITMAP-NUMBER	integer	integer	Set each tab in turn to use the given integer tile number within the bitmap.
BITMAP-VALUE	text	N/A	Set the bitmap resource to use by name rather than by handle number.

Events Table

Name	Value	Description
CMD-TABCHANGED	7	User selects different tab. EVENT-DATA-1 is selected tab.
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
MSG-VALIDATE	16391	Allow program to validate data entered.

BITMAP

A BITMAP is a graphic image capable of being displayed on screen. The image must be in a format recognized by the Elastic COBOL runtime or Java Virtual Machine. This currently includes JPG, GIF for all JDK's, BMP for the Elastic COBOL runtime, and PNG for JDK 1.3+.

SIZE and LINES are expressed in pixels.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
TILED	boolean	boolean	Specify that the image should be repeated in a tiled fashion to completely fill the given area.
CENTER	boolean	boolean	Specify that the image should be centered within the given area.
CENTERED	boolean	boolean	Specify that the image should be centered within the given area.
SCALED	boolean	boolean	Specify that the image should be scaled to match the given area.
BITMAP-HANDLE	integer	integer	Specify the bitmap by handle number, as loaded by the W$BITMAP routine.
BITMAP-NUMBER	integer	integer	Specify the tile number within the bitmap to display.
BITMAP-START	integer	integer	Specify the starting tile for animation, interacting with BITMAP-END and BITMAP-

Name	MODIFY	INQUIRE	Descriptions
			TIME.
BITMAP-END	integer	integer	Specify the ending tile for animation, interacting with BITMAP-START and BITMAP-TIME.
BITMAP-TIMER	integer	integer	Specify the time for each tile in animation in hundredths of a second, interacting with BITMAP-START and BITMAP-END.
TILE-OFFSET	integer	integer	Specify an offset to add to each row when the image is TILED to create a more aesthetic effect.
BORDER-WIDTH	integer	integer	Specify the width of the border around the image.
BORDER-HEIGHT	integer	integer	Specify the height of the border around the image.
IMAGE	text	text	Create an image from text, a line at a time by specifying each IMAGE line in turn. The image is actually created when the final line is set. This is useful for creating small images for icons inline. The characters in the text to set represent colors from the following table: Character — Color Space — Black B — Blue C — Cyan < — Dark Gray | — Gray G — Green > — Light Gray M — Magenta O — Orange P — Pink R — Red W — White Y — Yellow V — Violet
BITMAP-VALUE	text	N/A	Specify the bitmap by resource name directly.

CALENDAR-DISPLAY

General information

A CALENDAR-DISPLAY is a month by month calendar, visible one month at a time. It can be used for selecting dates, but it requires a large amount of screen real estate.

Many properties are the same between CALENDAR-DISPLAY and CALENDAR-FIELD; many are useful generally in INQUIRE.

Properties Table

Name	Modify	Inquire	Description
CELL-BACKGROUND	integer	N/A	Cell background color
CELL-FOREGROUND	integer	N/A	Cell foreground color
CELL-UNSELECTED	integer	N/A	Cell unselected color
DAY-OF-MONTH	integer	integer	Selected day of month.
DATE	integer	integer	Synonym for DAY-OF-MONTH.
DAY-OF-WEEK	integer	integer	Selected day of week.
DAY-OF-WEEK-IN-MONTH	integer	integer	Selected day of week in month.
DAY-OF-YEAR	integer	integer	Selected day of year.
JULIAN-DAY	integer	integer	Synonym for DAY-OF-YEAR.
DST-OFFSET	integer	integer	Selected DST offset.
ERA	integer	integer	Selected era.
MONTH	integer	integer	Selected month.
WEEK-OF-MONTH	integer	integer	Selected week of month.
WEEK-OF-YEAR	integer	integer	Selected week of year.
YEAR	integer	integer	Selected year.
ZONE-OFFSET	integer	integer	Selected zone offset.
GO-NEXT-MONTH	boolean	N/A	Advance display to the next month.
GO-PREVIOUS-MONTH	boolean	N/A	Advance display to the previous month.
GO-NEXT-YEAR	boolean	N/A	Advance display to the next year.
GO-PREVIOUS-YEAR	boolean	N/A	Advance display to the previous year.

CALENDAR-FIELD

General information

A CALENDAR-FIELD is a textfield with a popup displaying a calendar, as in the CALENDAR-DISPLAY component. This allows the user to enter a date by typing it or selecting it from a visual calendar.

Many properties are the same between CALENDAR-DISPLAY and CALENDAR-FIELD; many are useful generally in INQUIRE.

Properties Table

Name	Modify	Inquire	Description
CELL-BACKGROUND	integer	N/A	Cell background color
CELL-FOREGROUND	integer	N/A	Cell foreground color
CELL-UNSELECTED	integer	N/A	Cell unselected color
DAY-OF-MONTH	integer	integer	Selected day of month.
DAY-OF-WEEK	integer	integer	Selected day of week.
DAY-OF-WEEK-IN-MONTH	integer	integer	Selected day of week in month.
DAY-OF-YEAR	integer	integer	Selected day of year.
JULIAN-DAY	integer	integer	Synonym for DAY-OF-YEAR.
DST-OFFSET	integer	integer	Selected DST offset.
ERA	integer	integer	Selected era.
MONTH	integer	integer	Selected month.
WEEK-OF-MONTH	integer	integer	Selected week of month.
WEEK-OF-YEAR	integer	integer	Selected week of year.
YEAR	integer	integer	Selected year.
ZONE-OFFSET	integer	integer	Selected zone offset.
GO-NEXT-MONTH	boolean	N/A	Advance display to the next month.
GO-PREVIOUS-MONTH	boolean	N/A	Advance display to the previous month.
GO-NEXT-YEAR	boolean	N/A	Advance display to the next year.
GO-PREVIOUS-YEAR	boolean	N/A	Advance display to the previous year.

TREE-VIEW

A TREE-VIEW presents hierarchical information in a tree, where the user may expand and collapse nodes in the tree to expose children which may in turn be nodes.

Each item in the tree is assigned an ID which is an integer identifier; the TREE-VIEW ID may be declared as any numeric integer type. (As a special case, this may be the POINTER type; in such a case, only the address is used as the integer identifier.)

Because of the complexity of the TREE-VIEW, a special identifier called ITEM points to the current ITEM affected by the operation. Each item other than the root item has exactly one parent.

A TREE-VIEW does not have a TITLE. The VALUE is the currently selected ID.

Properties Table

Name	MODIFY	INQUIRE	Descriptions
BOXED	boolean	boolean	Set the TREE-VIEW to show a visible surrounding box

Name	MODIFY	INQUIRE	Descriptions
			(default).
BUTTONS	boolean	boolean	Show visible buttons to the left of the item, indicating a '+' and '-' for expanding and collapsing the tree.
LINES-AT-ROOT	boolean	boolean	Set SHOW-LINES and BUTTON to apply to the root.
NO-BOX	boolean	boolean	Do not show a visible surrounding box.
SHOW-LINES	boolean	boolean	Draw lines between items to make explicit the familial relationship.
SHOW-SEL-ALWAYS	boolean	boolean	Always show the currently selected node even when not in focus.
BITMAP-HANDLE	integer	integer	Set the bitmap handle to use for the graphics to show as buttons.
BITMAP-NUMBER	integer	integer	Set the tile number within the bitmap. This actually assigns the bitmap to the current item.
BITMAP-WIDTH	integer	integer	Set the width of tiles in the bitmap.
ENSURE-VISIBLE	integer	integer	Ensure that the given ID number is visible in the component.
EXPAND	integer	integer	Expand the given ID number.
HIDDEN-DATA	text	text	Get or set hidden data associated with the item.
HAS-CHILDREN	integer	integer	If not zero, then the item is a parent node even if it has no children currently added; this allows the item to be collapsed and expanded. If zero, then the actual children are used to determine if the item has children.
ITEM	integer	integer	Set the current item to the given integer ID.
ITEM-TEXT	text	text	Set the item's text to the given text.
ITEM-TO-ADD	text	text	Add a new node called 'text' using the PARENT and PLACEMENT properties, returning its ID. If successfully added, ITEM is left pointing at the new node.
ITEM-TO-DELETE	integer	integer	Delete the given ID.
ITEM-TO-EMPTY	integer	integer	Delete all children of ID.
NEXT-ITEM	integer	integer	Traverse the tree from the current item ID. Set the integer to the manner in which to traverse from the following list:

Name	MODIFY	INQUIRE	Descriptions
			TVNI-CHILD First child of current. TVNI-FIRST-VISIBLE First item currently visible. TVNI-NEXT Next sibling of current. TVNI-NEXT-VISIBLE Next visible item after current. TVNI-PARENT Parent item of current. TVNI-PREVIOUS Previous sibling of current. TVNI-PREVIOUS-VISIBLE Previous visible item before current. TVNI-ROOT Root of entire tree.
PARENT	integer	integer	Set item which is to be parent of future items to add. Setting parent to 0 resets the parent to the root.
PLACEMENT	integer	integer	Set where to place new items under the parent. May be set to a particular item ID or to one of the special items: TVPLACE-FIRST Place first in list. TVPLACE-LAST Place last in list. TVPLACE-SORT Place in list in sorted order.
RESET-LIST	integer	integer	When set to non-zero value, removes all items from the TREE-VIEW.
BITMAP-VALUE	text	N/A	Specify the bitmap by name rather than by bitmap handle. (see example below)

Events Table

Name	Value	Description
CMD-GOTO	3	User selected component for focus.
CMD-HELP	8	N/A. Help requested for component. EVENT-DATA-2 is help-id.
MSG-VALIDATE	16391	Allow program to validate data entered.
MSG-TV-SELCHANGING	16411	TREE-VIEW selection about to change. EVENT-DATA-1 is reason for change.
MSG-TV-SELCHANGE	16412	TREE-VIEW selection has changed. EVENT-DATA-1 is reason for change. EVENT-DATA-2 is ID of item selected.
MSG-TV-EXPANDING	16413	TREE-VIEW item about to expand/collapse. EVENT-DATA-1 is

Name	Value	Description
		TVFLAG-EXPAND (2)or TVFLAG-COLLAPSE (1). EVENT-DATA-2 is ID or item.
MSG-TV-EXPANDED	16414	TREE-VIEW item has expanded/collapsed. EVENT-DATA-1 is TVFLAG-EXPAND (2) or TVFLAG-COLLAPSE (1). EVENT-DATA-2 is ID or item.
MSG-TV-DBLCLICK	16428	User double-clicked item with no children. EVENT-DATA-2 is ID of item.

Example

```
77 PARENT-1 SIGNED-INT.
77 PARENT-2 SIGNED-INT.
77 PARENT-3 SIGNED-INT.

MODIFY MY-TREE-VIEW
        ITEM-TO-ADD = "John Adams" GIVING PARENT-1
                PARENT = PARENT-1
                ITEM-TO-ADD = "John Quincy Adams"
                PARENT = 0
        ITEM-TO-ADD = "George H. W. Bush" GIVING PARENT-2
        PARENT = PARENT-2
                ITEM-TO-ADD = "George W. Bush"
                ITEM-TO-ADD = "Jeb Bush"
                PARENT = 0
        ITEM-TO-ADD = "William J. Clinton" GIVING PARENT-3
                PARENT = PARENT-3
                ITEM-TO-ADD = "Chelsea Clinton"
                PARENT = 0
```

Creates the following tree structure

John Adams

    John Quincy Adams

George H. W. Bush

    George W. Bush

    Jeb Bush

William J. Clinton

    Chelsea Clinton

MENU

A MENU provides a menu structure at the top of the window or a pop-up menu for an individual component.

There is a function call W$MENU which may be used to create menus, but it should be avoided. The MENU component functionality provides a generally superior interface.

Properties Table

	MODIFY	INQUIRE	Descriptions
CHECKABLE	boolean	boolean	Set if the next item should be checkable.
UNCHECKED	boolean	boolean	Set a checkable to unchecked.
CHECKED	boolean	boolean	Set a checkable to checked.
BLOCKED	boolean	boolean	Set the menu to block input.
POPUP-TYPE	boolean	boolean	Set the menu to be a pop-up type.
BEGIN-MENU	text	N/A	Start describing a submenu (at any level). Menus may be nested. The text is the text of the menu item; it may include an ampersand (&) to mark the hotkey. The submenu is continued until END-MENU. Every BEGIN-MENU have a following END-MENU.
END-MENU	N/A	N/A	Stop describing a submenu. Every BEGIN-MENU have a following END-MENU.
ITEM-TO-ADD	text	N/A	Same as TREE-VIEW.
ITEM-TO-DELETE	text	N/A	Same as TREE-VIEW.
ITEM	text	text	Same as TREE-VIEW.
SEPARATOR	N/A	N/A	Add a SEPARATOR line to the menu.
ACCELERATOR	text	text	Set the ACCELERATOR for the current item to the given text. The text is a KeyStroke value.
PARENT-ITEM	text	text	Same as TREE-VIEW.
PARENT	text	text	Same as TREE-VIEW.
TERMINATION-TRUE	text	text	Set the current item to terminate with the given termination value if the checked state is true.
TERMINATION-FALSE	text	text	Set the current item to terminate with the given termination value if the checked state is false.
EXCEPTION-TRUE	text	text	Set the current item to terminate with the given exception value if the checked state is true.
EXCEPTION-FALSE	text	text	Set the current item to terminate with the given exception value if the checked state is false.
SELECTED-USING	identifier	N/A	Set identifier to item's value upon selection.
SELECTED-USING-TRUE	identifier	N/A	Set identifier to item's value upon selection if state is true.
SELECTED-USING-FALSE	text	N/A	Set identifier to item's value upon selection if state is false.
SELECTED-VALUE	text	N/A	Set identifier to item's value

	MODIFY	INQUIRE	Descriptions
			upon selection.
SELECTED-VALUE-TRUE	text	N/A	Set identifier to item's value upon selection if state is true.
SELECTED-VALUE-FALSE	N/A	N/A	Set identifier to item's value upon selection if state is false.
RESET-MENU	N/A	N/A	Reset the menu contents if set to non-zero.
NOTIFY	boolean	boolean	Send the notification values for the item.
HELP-MENU	N/A	N/A	Set the current item to be a help menu.

Events Table

Name	Value	Description
MSG-INIT-MENU	16398	N/A. Component pop-up to be displayed. EVENT-DATA-2 is menu handle.
MSG-MENU-INPUT	16397	N/A. User selected item from pop-up. EVENT-DATA-2 is menu ID.
MSG-END-MENU	16399	N/A. Pop-up is no longer visible. EVENT-DATA-2 is menu handle.

Example

```
01 MENU-ITEMS
     03 MENU
          BEGIN-MENU = "&File"

           ITEM-TO-ADD = "&Open"
           ACCELERATOR = "control O"
           SELECTED-USING OPTION-1 NOTIFY

           ITEM-TO-ADD = "&Save "
           ACCELERATOR = "control S"
           SELECTED-USING OPTION-2 NOTIFY

           ITEM-TO-ADD = "Save &As…"
           ACCELERATOR = "control A"
           SELECTED-USING OPTION-3 NOTIFY

           CHECKABLE ITEM-TO-ADD = "&Property"
           ACCELERATOR = "control P"
           SELECTED-USING OPTION-4 NOTIFY

           SEPARATOR

           ITEM-TO-ADD = "E&xit"
           ACCELERATOR = "alt shift X"
           TERMINATION-VALUE 13

          END-MENU
          .
```

WEB-BROWSER

The WEB-BROWSER component is used for displaying and browsing HTML documents.

Currently, the WEB-BROWSER component uses the HTML rendering pane of Java which is suitable for displaying HTML, but it is not a full-featured web browser.

The WEB-BROWSER does not use the TITLE property, but the VALUE property is the value of the displayed URL.

Properties Table

	MODIFY	INQUIRE	Descriptions
GO-BACK	integer	integer	Navigate to the previous item in the history.
GO-FORWARD	integer	integer	Navigate to the next item in the history.
GO-HOME	text	text	Navigate to the home page.
GO-SEARCH	text	text	Navigate to the search page.
REFRESH	integer	integer	Refresh the current page.
REFRESH-BROWSER	integer	integer	Refresh the current page.
STOP	integer	integer	Stop loading the current page.
STOP-BROWSER	integer	integer	Stop loading the current page.
BUSY	integer	integer	Obtain the browser's busy state.
TYPE	N/A	integer	Type of content for the current page.
STATUS-TEXT	text	text	Text displayed in the status bar.
NAVIGATE-URL	text	text	The web page URL to which to navigate.
PROGRESS	integer	integer	Current amount of progress.
MAX-PROGRESS	integer	integer	Maximum amount of progress.

Events Table

Name	Value	Description
MSG-WB-BEFORE-NAVIGATE	16429	N/A. When about to navigate to new URL.
MSG-WB-DOWNLOAD-BEGIN	16431	N/A. When about to download.
MSG-WB-DOWNLOAD-COMPLETE	16432	N/A. When download is complete.
MSG-WB-NAVIGATE-COMPLETE	16430	N/A. When navigation is complete.
MSG-WB-PROGRESS-CHANGE	16433	N/A. When progress amount is changed.
MSG-WB-STATUS-TEXT-CHANGE	16434	N/A. When the status text is changed.
MSG-WB-TITLE-CHANGE	16435	N/A. When the title is changed.

WINDOW

The WINDOW is the top-level component. All other components are placed on the WINDOW. It must be created only with a DISPLAY statement, not within the screen section.

Properties Table

Name	Modify	Inquire	Description
MODELESS	boolean	boolean	A modal window blocks all other application windows from input.
MODAL	boolean	boolean	A modal window blocks all other application windows from input.
FLOATING	boolean	boolean	Specify that this is a floating window.
BIND TO THREAD	boolean	boolean	Request the runtime switch threads of execution automatically depending on controlling thread. This is ignored; Elastic COBOL has a more fully threaded environment which does not require this property. Bind to thread is identical to link to thread, except it requests the window automatically be destroyed when the controlling thread terminates. This is ignored.
LINK TO THREAD	boolean	boolean	Request the runtime switch threads of execution automatically depending on controlling thread. This is ignored; Elastic COBOL has a more fully threaded environment which does not require this property. Bind to thread is identical to link to thread, except it requests the window automatically be destroyed when the controlling thread terminates. This is ignored.
SHADOW	boolean	boolean	Request a shadow be shown under the window to create a three-dimensional effect. This is handled by the window manager and so ignored.
ACTION	integer	integer	Set to ACTION-MAXIMIZE, ACTION-MINIMIZE, ACTION-RESTORE to perform an action on the window. These

Name	Modify	Inquire	Description
			options are not supported on all Java Virtual Machines.
INITIAL	N/A	N/A	Specify that this is the main, initial window.
INDEPENDENT	N/A	N/A	Imply CELL SIZE = LABEL FONT, making the vertical height for a cell the same as the default height for an ENTRY-FIELD, which is taller than the normal default height for a LABEL.
GRAPHICAL	boolean	boolean	Imply CELL SIZE = LABEL FONT, making the vertical height for a cell the same as the default height for an ENTRY-FIELD, which is taller than the normal default height for a LABEL.
BOXED	boolean	boolean	Request a box be drawn around the window. This is handled by the window manager and so ignored.
REVERSE-VIDEO	N/A	N/A	Request reverse-video for the window.
REVERSE	N/A	N/A	Request reverse-video for the window.
REVERSED	N/A	N/A	Request reverse-video for the window.
USER-GRAY	N/A	N/A	The USER- phrases map colors to the system default color scheme, causing the graphical screen section support to match the surrounding environment better. USER-GRAY maps color number 8 to the 3-D object color. The way to properly use these settings is to have color 8 as the background for graphical component area and color 16 for plain text background areas, the application will better match the surrounding environment.
USER-WHITE	N/A	N/A	The USER- phrases map colors to the system default color scheme, causing the graphical screen section support to match the surrounding environment better. USER-WHITE maps color number 16 to the normal background color for

Name	Modify	Inquire	Description
			application windows. The way to properly use these settings is to have color 8 as the background for graphical component area and color 16 for plain text background areas, the application will better match the surrounding environment.
USER-COLORS	N/A	N/A	The USER- phrases map colors to the system default color scheme, causing the graphical screen section support to match the surrounding environment better. USER-COLORS implies USER-GRAY and USER-WHITE. The way to properly use these settings is to have color 8 as the background for graphical component area and color 16 for plain text background areas, the application will better match the surrounding environment.
TITLE-BAR	N/A	N/A	Request that the window possess a title-bar.
AUTO-RESIZE	N/A	N/A	Request that the window be capable of resizing itself upon user request.
RESIZABLE	N/A	N/A	Request that the window be resizable.
MIN-SIZE	integer	N/A	Request a minimum and maximum size for the window. This is handled by the window manager and so ignored.
MAX-SIZE	integer	N/A	Request a minimum and maximum size for the window. This is handled by the window manager and so ignored.
MIN-LINES	integer	N/A	Request a minimum and maximum size for the window. This is handled by the window manager and so ignored.
MAX-LINES	integer	N/A	Request a minimum and maximum size for the window. This is handled by the window manager and so ignored.
AUTO-MINIMIZE	N/A	N/A	Request that minimize be automatically handled.
CONTROLS-UNCROPPED	N/A	N/A	Request that components not be cropped to meet the current windows settings. This property is not necessary in

Name	Modify	Inquire	Description
			Elastic COBOL and so ignored.
NO SCROLL	N/A	N/A	Request automatic scrolling be disabled for the window. This is ignored.
NO WRAP	N/A	N/A	Request wrapping be disabled for the window. This is ignored.
LABEL-OFFSET	integer	integer	Request a default label offset property for the window.
CONTROL-FONT	integer	N/A	Specify the default font for graphical components. The integer is the font handle.
BACKGROUND-IMAGE	text	text	Set a background image for the window, optionally scaling it to fill the window or tiling it repeatedly to fill the window. The text is the image resource name (such as filename.gif) to use for the image. Note that a background image precludes plain text output on the window, but does allow graphical components on the window. (The background image and pain text input/output are at same graphical layer and mutually exclusive.)
BACKGROUND-IMAGE-SCALED	text	text	Set a background image for the window, optionally scaling it to fill the window or tiling it repeatedly to fill the window. The text is the image resource name (such as filename.gif) to use for the image. Note that a background image precludes plain text output on the window, but does allow graphical components on the window. (The background image and pain text input/output are at same graphical layer and mutually exclusive.)
BACKGROUND-IMAGE-TILED	text	text	Set a background image for the window, optionally scaling it to fill the window or tiling it repeatedly to fill the window. The text is the image resource name (such as filename.gif) to use for the image. Note that a background image precludes plain text output on

Name	Modify	Inquire	Description
			the window, but does allow graphical components on the window. (The background image and pain text input/output are at same graphical layer and mutually exclusive.)
ICON-IMAGE-NAME	text	text	Set the icon-image (upper-left corner in most operating systems) to the given name, where the name is the filename of a supported image
GRADIENT	boolean	boolean	Enable the color gradient, such as used by installers. This gradient precludes use of plain text display/accept in the window.
GRADIENT-FROM	integer	integer	Color number from which gradient starts.
GRADIENT-TO	integer	integer	Color number to which gradient proceeds.
GRADIENT-BORDER-WIDTH	integer	integer	Width in pixels of border around gradient.
GRADIENT-BORDER	boolean	boolean	Set the gradient border on or off.
GRADIENT-BORDER-COLOR	integer	integer	Set the color of the gradient border.
DO-MINIMIZE	-	N/A	Cause window to be minimized.
DO-MAXIMIZE	-	N/A	Cause window to be maximized.
DO-RESTORE	-	N/A	Cause window to be restored.
DO-TO-FRONT	-	N/A	Cause window to go to front.
DO-TO-BACK	-	N/A	Cause window to go to back.
CENTERED	-	N/A	Create window in center of screen.
TILED	boolean	N/A	Create window in tiled position on screen.

TOOLBAR

A TOOLBAR is a component which is present only at the top position in the window. It displays and accepts choices, typically small icon images. It must be created only with a DISPLAY statement, not within the screen section.

The TOOLBAR has no additional properties or events of its own. It is merely a holding place for other components to be displayed upon.

Properties Table

NONE

Events Table

NONE

MESSAGE BOX

A MESSAGE BOX is a component which is displayed and then removed. It displays a message to the user which is then dismissed by the user. It must be created only with a DISPLAY statement, not within the screen section.

Properties Table

	MODIFY	INQUIRE	Descriptions
TYPE	integer	integer	Set the type of the message box. (OK is 1, YES-NO is 2, OK-CANCEL is 3, YES-NO-CANCEL is 4.)
ICON	integer	integer	Set the icon of the message box.
DEFAULT	integer	integer	Set the default selection of the message box.

Events Table

NONE

Sample

```
id division.
program-id. message-box.

data division.
working-storage section.
77 message-box-value pic 9.

procedure division.
main-paragraph.

    display message box
        "This is text for the user"
         title "Title of Message Box"
         type 4
         giving message-box-value

    if message-box-value = 1 then
      display "yes" upon sysout
    end-if

    if message-box-value = 2 then
      display "no" upon sysout
    end-if

    if message-box-value = 3 then
      display "cancel" upon sysout
    end-if
```

```
stop all run
.
```

Appendix F - Internal Copy Files

Internal copy files are present for wider compatibility with the graphical screen section support. An internal copy file is not an actual file, but rather a signal to the compiler to include certain constants which otherwise would not be defined. An internal copy file will always be overridden by an actual copy file by the same name if present. If an internal copy file is used, a warning will be issued.

ELASTIC COBOL -ACTIVEX.DEF and ACTIVEX.DEF

Constant Name	Value	Notes
OLE-BLACK	0	OLE COLOR VALUES
OLE-RED	255	
OLE-GREEN	65280	
OLE-YELLOW	65535	
OLE-BLUE	16711680	
OLE-MAGENTA	16711935	
OLE-CYAN	16776960	
OLE-WHITE	16777215	
CRESOURCE-LOAD	1	C$RESOURCE
CRESOURCE-DESTROY	2	

ELASTIC COBOL .DEF and ACUCOBOL.DEF

Constant Name	Value	Notes
BLACK	1	
BLUE	2	
GREEN	3	
CYAN	4	
RED	5	
MAGENTA	6	
BROWN	7	
WHITE	8	
DARK-GRAY	9	
BRIGHT-BLUE	10	
BRIGHT-GREEN	11	
BRIGHT-CYAN	12	
BRIGHT-RED	13	
BRIGHT-MAGENTA	14	
YELLOW	15	
BRIGHT-WHITE	16	
BCKGRND-BLACK	32	
BCKGRND-BLUE	64	

Constant Name	Value	Notes
BCKGRND-GREEN	96	
BCKGRND-CYAN	128	
BCKGRND-RED	160	
BCKGRND-MAGENTA	192	
BCKGRND-BROWN	224	
BCKGRND-WHITE	256	
BCKGRND-DARK-GRAY	288	
BCKGRND-BRIGHT-BLUE	320	
BCKGRND-BRIGHT-GREEN	352	
BCKGRND-BRIGHT-CYAN	384	
BCKGRND-BRIGHT-RED	416	
BCKGRND-BRIGHT-MAGENTA	448	
BCKGRND-YELLOW	480	
BCKGRND-BRIGHT-WHITE	512	
COLOR-REVERSE	1024	
FRGRND-LOW	2048	
FRGRND-HIGH	4096	
COLOR-UNDERLINE	8192	
COLOR-BLINK	16384	
COLOR-PROTECTED	32768	
BCKGRND-LOW	65536	
BCKGRND-HIGH	131072	
WINDOW-BRIGHT-WHITE	131328	
GET-FILE-STATUS	1	C$RERR
GET-TRANSACTION-STATUS	2	
START-FILESYSTEM-LIST	0	C$FILESYS
CONTINUE-FILESYSTEM-LIST	1	
CHECK-FOR-FILESYSTEM	2	
NUMBER-OF-FILESYSTEMS	3	
CSYS-ASYNC	1	C$SYSTEM
CSYS-NO-IO	2	
CSYS-MAXIMIZED	4	
CSYS-MINIMIZED	8	
CSYS-COMPATIBILITY	16	
CSYS-HIDDEN	32	
CSYS-SHELL	64	

ELASTIC COBOL GUI.DEF and ACUGUI.DEF

Constant Name	Value	Notes
TEST-MOUSE-PRESENCE	0	W$MOUSE
GET-MOUSE-STATUS	1	
GET-MOUSE-SCREEN-STATUS	2	
SET-MOUSE-POSITION	3	
SET-MOUSE-SCREEN-POSITION	4	
SET-MOUSE-SHAPE	5	

Constant Name	Value	Notes
SET-DELAYED-MOUSE-SHAPE	6	
GET-MOUSE-SHAPE	7	
CAPTURE-MOUSE	8	
RELEASE-MOUSE	9	
ENABLE-MOUSE	10	
SET-MOUSE-HELP	19	
ARROW-POINTER	1	
BAR-POINTER	2	
CROSS-POINTER	3	
WAIT-POINTER	4	
HELP-POINTER	5	
AUTO-MOUSE-HANDLING	1	MOUSE-FLAGS
ALLOW-LEFT-DOWN	2	
ALLOW-LEFT-UP	4	
ALLOW-LEFT-DOUBLE	8	
ALLOW-MIDDLE-DOWN	16	
ALLOW-MIDDLE-UP	32	
ALLOW-MIDDLE-DOUBLE	64	
ALLOW-RIGHT-DOWN	128	
ALLOW-RIGHT-UP	256	
ALLOW-RIGHT-DOUBLE	512	
ALLOW-MOUSE-MOVE	1024	
ALWAYS-ARROW-CURSOR	2048	
ALLOW-ALL-SCREEN-ACTIONS	16384	
WMENU-NEW	1	W$MENU OPCODES
WMENU-DESTROY	2	
WMENU-ADD	3	
WMENU-CHANGE	4	
WMENU-DELETE	5	
WMENU-CHECK	6	
WMENU-UNCHECK	7	
WMENU-ENABLE	8	
WMENU-DISABLE	9	
WMENU-SHOW	10	
WMENU-GET-MENU	11	
WMENU-INPUT	12	
WMENU-BLOCK	13	
WMENU-UNBLOCK	14	
WMENU-GET-BLOCK	15	
WMENU-SET-BLOCK	16	
WMENU-RELEASE	17	
WMENU-GET-CONFIGURATION	18	
WMENU-SET-CONFIGURATION	19	
WMENU-REFRESH	20	
WMENU-DESTROY-DELAYED	21	
WMENU-GET-DELAYED-FLAG	22	
WMENU-SET-DELAYED-FLAG	23	
WMENU-NEW-POPUP	26	

Constant Name	Value	Notes
WMENU-POPUP	27	
W-UNCHECKED	0	W$MENU
W-CHECKED	1	
W-ENABLED	0	
W-DISABLED	16	
W-SEPARATOR	256	
MB-OK	1	MESSAGE BOX
MB-YES-NO	2	
MB-OK-CANCEL	3	
MB-YES-NO-CANCEL	4	
MB-YES	1	MESSAGE BOX RESPONSE
MB-NO	2	
MB-CANCEL	3	
MB-DEFAULT-ICON	1	MESSAGE BOX ICON
MB-WARNING-ICON	2	
MB-ERROR-ICON	3	
WBITMAP-DISPLAY	1	W$BITMAP OPCODES
WBITMAP-DESTROY	2	
WBITMAP-LOAD	3	
WBITMAP-NO-FILL	1	W$BITMAP FLAGS
WBERR-UNSUPPORTED	0	W$BITMAP ERRORS
WBERR-FILE-ERROR	-1	
WBERR-NO-MEMORY	-2	
WBERR-NOT-BITMAP	-3	
WBERR-FORMAT-UNSUPPORTED	-4	
WBERR-MISSING-DLL	-5	
EVENT-ACTION-NORMAL	0	EVENT ACTION
EVENT-ACTION-TERMINATE	1	
EVENT-ACTION-CONTINUE	2	
EVENT-ACTION-IGNORE	3	
EVENT-ACTION-FAIL	4	
EVENT-ACTION-COMPLETE	5	
EVENT-ACTION-FAIL-TERMINATE	7	
W-TIMEOUT	99	STANDARD EXCEPTION
W-CONVERSION-ERROR	98	
W-NO-FIELDS	97	
W-EVENT	96	
W-MESSAGE	95	
CMD-CLOSE	1	COMMANDS
CMD-GOTO	3	
CMD-CLICKED	4	
CMD-DBLCLICK	5	
CMD-ACTIVATE	6	
CMD-TABCHANGED	7	
CMD-HELP	8	
NTF-SELCHANGE	4099	NOTIFICATION
NTF-CHANGED	4100	
NTF-PL-NEXT	4101	
NTF-PL-PREV	4102	
NTF-PL-NEXTPAGE	4103	
NTF-PL-PREVPAGE	4104	

Constant Name	Value	Notes
NTF-PL-FIRST	4105	
NTF-PL-LAST	4106	
NTF-PL-SEARCH	4107	
NTF-RESIZED	4114	
MSG-SB-NEXT	16385	MESSAGE EVENTS
MSG-SB-PREV	16386	
MSG-SB-NEXTPAGE	16387	
MSG-SB-PREVPAGE	16388	
MSG-SB-THUMB	16389	
MSG-SB-THUMBTRACK	16390	
MSG-VALIDATE	16391	
MSG-BEGIN-ENTRY	16392	
MSG-FINISH-ENTRY	16393	
MSG-CANCEL-ENTRY	16394	
MSG-GOTO-CELL	16395	
MSG-GOTO-CELL-MOUSE	16396	
MSG-MENU-INPUT	16397	
MSG-INIT-MENU	16398	
MSG-END-MENU	16399	
MSG-BITMAP-CLICKED	16400	
MSG-BITMAP-DBLCLICK	16401	
MSG-HEADING-CLICKED	16402	
MSG-HEADING-DBLCLICK	16403	
MSG-GOTO-CELL-DRAG	16404	
MSG-HEADING-DRAGGED	16405	
MSG-BEGIN-DRAG	16406	
MSG-END-DRAG	16407	
MSG-BEGIN-HEADING-DRAG	16408	
MSG-END-HEADING-DRAG	16409	
MSG-COL-WIDTH-CHANGED	16410	
MSG-TV-SELCHANGING	16411	
MSG-TV-SELCHANGE	16412	
MSG-TV-EXPANDING	16413	
MSG-TV-EXPANDED	16414	
MSG-CLOSE	16415	
MSG-SPIN-UP	16416	
MSG-SPIN-DOWN	16417	
MSG-PAGED-NEXT	16419	
MSG-PAGED-PREV	16420	
MSG-PAGED-NEXTPAGE	16421	
MSG-PAGED-PREVPAGE	16422	
MSG-PAGED-FIRST	16423	
MSG-PAGED-LAST	16424	
MSG-GRID-RBUTTON-DOWN	16426	
MSG-GRID-RBUTTON-UP	16427	
MSG-TV-DBLCLICK	16428	
MSG-WB-BEFORE-NAVIGATE	16429	
MSG-WB-NAVIGATE-COMPLETE	16430	
MSG-WB-DOWNLOAD-BEGIN	16431	
MSG-WB-DOWNLOAD-COMPLETE	16432	
MSG-WB-PROGRESS-	16433	

Constant Name	Value	Notes
CHANGE		
MSG-WB-STATUS-TEXT-CHANGE	16434	
MSG-AX-EVENT	16436	
ACTION-CUT	1	ENTRY-FIELD ACTIONS
ACTION-COPY	2	
ACTION-PASTE	3	
ACTION-DELETE	4	
ACTION-UNDO	5	
ACTION-FIRST-PAGE	10	PAGED-GRID ACTION
ACTION-LAST-PAGE	11	
ACTION-CURRENT-PAGE	12	
ACTION-MINIMIZE	20	WINDOW ACTION
ACTION-MAXIMIZE	21	
ACTION-RESTORE	22	
ACTION-TOFRONT	1031	
ACTION-TOBACK	1032	
PAGED-AT-START	2147418113	PAGED GRID FILE POSITION
PAGED-AT-END	2147418114	
PAGED-EMPTY	2147418115	
GRDSRCH-NOT-FOUND	0	GRID SEARCH STATUS
GRDSRCH-FOUND	1	
GRDSRCH-WRAPPED	2	
SND-SYNC	0	WIN$PLAYSOUND
SND-ASYNC	1	
SND-LOOP	8	
SND-NOSTOP	16	
TVPLACE-FIRST	4294901761	TREE-VIEW
TVPLACE-LAST	0	
TVPLACE-SORT	4294901763	
TVNI-CHILD	1	TREE-VIEW NEXT ITEM
TVNI-FIRST-VISIBLE	2	
TVNI-NEXT	3	
TVNI-NEXT-VISIBLE	4	
TVNI-PARENT	5	
TVNI-PREVIOUS	6	
TVNI-PREVIOUS-VISIBLE	7	
TVNI-ROOT	8	
TVFLAG-COLLAPSE	1	TREE-VIEW MISCELLANEOUS
TVFLAG-EXPAND	2	
TVFLAG-PROGRAM	0	
TVFLAG-MOUSE	1	
TVFLAG-KEYBOARD	2	
TVFLAG-NORMAL	1	
TVFLAG-BOLD	2	
PL-SORT-DEFAULT	0	PAGED LIST-BOX SORT ORDER
PL-SORT-NONE	1	
PL-SORT-NATIVE	2	

Constant Name	Value	Notes
PL-SORT-NATIVE-IGNORE-CASE	3	
HKEY-CLASSES-ROOT-31	1	WINDOWS REGISTRY
HKEY-CLASSES-ROOT	2147483648	
HKEY-CURRENT-USER	2147483649	
HKEY-LOCAL-MACHINE	2147483650	
HKEY-USERS	2147483651	
HKEY-PERFORMANCE-DATA	2147483652	
HKEY-CURRENT-CONFIG	2147483653	
HKEY-DYN-DATA	2147483654	
REG-NONE	0	WINDOWS REGISTRY TYPES
REG-SZ	1	
REG-EXPAND-SZ	2	
REG-BINARY	3	
REG-DWORD	4	
REG-DWORD-LITTLE-ENDIAN	5	
REG-DWORD-BIG-ENDIAN	6	
REG-LINK	7	
REG-MULTI-SZ	8	
REG-RESOURCE-LIST	9	
REG-FULL-RESOURCE-DESCRIPTOR	10	
REG-RESOURCE-REQUIREMENTS-LIST	11	
KEY-QUERY-VALUE	1	WINDOWS REGISTRY ACCESS RIGHTS
KEY-SET-VALUE	2	
KEY-CREATE-SUB-KEY	4	
KEY-ENUMERATE-SUB-KEYS	8	
KEY-NOTIFY	16	
KEY-CREATE-LINK	32	
KEY-WRITE	131078	
KEY-READ	131097	
KEY-EXECUTE	131097	
KEY-ALL-ACCESS	983103	
REG-OPENED-NEW-KEY	1	WINDOWS REGISTRY DISPOSITION
REG-OPENED-EXISTING-KEY	2	

ELASTIC COBOL -CONTROLS.DEF and CONTROLS.DEF

Constant Name	Value	Notes
CTL-LABEL	1	
CTL-ENTRY-FIELD	2	
CTL-PUSH-BUTTON	3	
CTL-CHECK-BOX	4	
CTL-RADIO-BUTTON	5	
CTL-SCROLL-BAR	6	
CTL-LIST-BOX	7	
CTL-COMBO-BOX	8	
CTL-FRAME	9	
CTL-TAB	10	
CTL-BAR	11	
CTL-GRID	12	
CTL-BITMAP	13	
CTL-TREE-VIEW	14	
CTL-WEB-BROWSER	15	
CTL-ACTIVE-X	16	
S-PERMANENT	1073741824	
S-TEMPORARY	536870912	
S-NOTAB	268435456	
S-HEIGHT-IN-CELLS	134217728	
S-WIDTH-IN-CELLS	67108864	
S-3D	33554432	
S-OVERLAP-LEFT	16777216	
S-OVERLAP-TOP	8388608	
S-SELF-ACT	4194304	
S-NOTIFY	2097152	
P-TERMINATION-VALUE	1	
P-EXCEPTION-VALUE	2	
LS-LEFT	1	LABEL STYLES
LS-RIGHT	2	
LS-CENTER	4	
LS-NO-KEY-LETTER	8	
LS-TRANSPARENT	16	
LP-LABEL-OFFSET	1	LABEL PROPERTIES
EFS-LEFT	1	ENTRY-FIELD STYLES
EFS-RIGHT	2	
EFS-CENTER	4	
EFS-BOX	8	
EFS-NO-BOX	16	
EFS-MULTILINE	32	
EFS-VSCROLL	96	
EFS-VSCROLL-BAR	224	
EFS-USE-RETURN	256	
EFS-USE-TAB	512	

Constant Name	Value	Notes
EFS-UPPER	1024	
EFS-LOWER	2048	
EFS-NO-AUTOSEL	4096	
EFS-READ-ONLY	8192	
EFS-AUTOTERMINATE	16384	
EFS-NOTIFY-CHANGE	32768	
EFS-SECURE	65536	
EFS-NUMERIC	131072	
EFS-SPINNER	262144	
EFS-AUTO-SPIN	262208	
EFP-MAX-TEXT	3	ENTRY-FIELD PROPERTIES
EFP-MAX-LINES	4	
EFP-MIN-VAL	5	
EFP-MAX-VAL	6	
EFP-AUTO-DECIMAL	7	
EFP-CURSOR-ROW	8	
EFP-CURSOR	4097	
EFP-ACTION	4098	
EFP-SELECTION-TEXT	4099	
EFP-CURSOR-COL	4100	
PBS-DEFAULT-BUTTON	1	PUSH-BUTTON STYLES
PBS-ESCAPE-BUTTON	2	
PBS-OK-BUTTON	4	
PBS-CANCEL-BUTTON	8	
PBS-NO-AUTO-DEFAULT	16	
PBS-BITMAP	32768	
PBS-SQUARE	16384	
PBS-FRAMED	8192	
PBS-UNFRAMED	4096	
PBS-FLAT	2048	
PBP-BITMAP-NUMBER	3	PUSH-BUTTON PROPERTIES
PBP-BITMAP-HANDLE	4	
CBS-BITMAP	32768	CHECK-BOX STYLES
CBS-SQUARE	16384	
CBS-FRAMED	8192	
CBS-UNFRAMED	4096	
CBS-FLAT	2048	
CBS-LEFT-TEXT	2	
CBP-BITMAP-NUMBER	3	CHECK-BOX PROPERTIES
CBP-BITMAP-HANDLE	4	
RBS-NO-GROUP-TAB	1	RADIO-BUTTON STYLES
RBS-LEFT-TEXT	2	
RBS-BITMAP	32768	
RBS-SQUARE	16384	
RBS-FRAMED	8192	
RBS-UNFRAMED	4096	
RBS-FLAT	2048	
RBP-BITMAP-NUMBER	3	RADIO-BUTTON PROPERTIES
RBP-BITMAP-HANDLE	4	
RBP-GROUP	5	
RBP-GROUP-VALUE	6	
SBS-HORIZONTAL	1	SCROLLBAR STYLES

Constant Name	Value	Notes
SBS-TRACK-THUMB	2	
SBP-MIN-VAL	1	SCROLLBAR PROPERTIES
SBP-MAX-VAL	2	
SBP-PAGE-SIZE	3	
SLS-HORIZONTAL	1	SLIDER STYLES
SLS-TRACK-THUMB	2	
SLP-MIN-VAL	1	SLIDER PROPERTIES
SLP-MAX-VAL	2	
SLP-PAINT-LABELS	1000	
SLP-PAINT-TICKS	1001	
SLP-PAINT-TRACK	1002	
SLP-SNAP-TO-TICKS	1003	
SLP-INVERTED	1004	
SLP-MAJOR-TICK	1005	
SLP-MINOR-TICK	1006	
LBS-UNSORTED	1	LIST-BOX STYLES
LBS-NO-BOX	2	
LBS-BOX	4	
LBS-NOTIFY-DBLCLICK	256	
LBS-NOTIFY-SELCHANGE	512	
LBS-PAGED	1024	
LBS-UPPER	2048	
LBS-NO-SEARCH	8192	
LBP-MASS-UPDATE	3	LIST-BOX PROPERTIES
LBP-INSERTION-INDEX	4	
LBP-DATA-COLUMNS	5	
LBP-DISPLAY-COLUMNS	6	
LBP-QUERY-INDEX	7	
LBP-ALIGNMENT	8	
LBP-SEPARATION	9	
LBP-DIVIDERS	10	
LBP-SORT-ORDER	11	
LBP-ITEM-TO-ADD	4097	
LBP-RESET-LIST	4098	
LBP-ITEM-TO-DELETE	4099	
LBP-SEARCH-TEXT	4100	
LBP-SELECTION-INDEX	4103	
LBP-ITEM-VALUE	4104	
LBP-THUMB-POSITION	4105	
CMS-UNSORTED	1	COMBO-BOX STYLES
CMS-DROP-DOWN	0	
CMS-STATIC-LIST	2	
CMS-DROP-LIST	4	
CMS-BOX	8	
CMS-NO-BOX	16	
CMS-NOTIFY-DBLCLICK	256	
CMS-NOTIFY-SELCHANGE	512	
CMS-UPPER	2048	
CMS-LOWER	4096	
CMP-MASS-UPDATE	3	COMBO-BOX PROPERTIES
CMP-MAX-TEXT	4	
CMP-INSERTION-INDEX4	5	

Constant Name	Value	Notes
CMP-ITEM-TO-ADD	4097	
CMP-RESET-LIST	4098	
CMP-ITEM-TO-DELETE	4099	
FS-RAISED	1	FRAME STYLES
FS-LOWERED	2	
FS-ENGRAVED	4	
FS-RIMMED	8	
FS-HEAVY	16	
FS-VERY-HEAVY	32	
FS-ALTERNATE	64	
FS-FULL-HEIGHT	128	
FP-HIGH-COLOR	1	FRAME PROPERTIES
FP-LOW-COLOR	2	
FP-FILL-COLOR	3	
FP-FILL-PERCENT	4	
FP-FILL-COLOR2	5	
FP-TITLE-POSITION	6	
TS-MULTILINE	1	TABBED-PANE STYLES
TS-BUTTONS	2	
TS-FIXED-WIDTH	4	
TP-BITMAP-HANDLE	1	TABBED-PANE PROPERTIES
TP-BITMAP-WIDTH	2	
TP-BITMAP-NUMBER	3	
TP-TAB-TO-ADD	4097	
TP-RESET-TABS	4098	
TP-TAB-TO-DELETE	4099	
BRS-DOTTED	1	BAR STYLES
BRS-DASHED	2	
BRS-DOTDASH	3	
BRP-WIDTH	1	BAR PROPERTIES
BRP-COLORS	2	
BRP-SHADING	3	
BRP-POSITION-SHIFT	4	
BRP-LEADING-SHIFT	5	
BRP-TRAILING-SHIFT	6	
BTS-TILED	256	BITMAP STYLES
BTS-CENTERED	512	
BTS-SCALED	1024	
BTP-BITMAP-NUMBER	1	BITMAP PROPERTIES
BTP-BITMAP-HANDLE	2	
BTP-BITMAP-START	3	
BTP-BITMAP-END	4	
BTP-BITMAP-TIMER	5	
BTP-TILE-OFFSET	100	
BTP-BORDER-WIDTH	101	
BTP-BORDER-HEIGHT	102	
BTP-IMAGE	103	
GRS-BOXED	1	GRID STYLES
GRS-NO-BOX	2	
GRS-VSCROLL	4	
GRS-HSCROLL	8	
GRS-COLUMN-HEADINGS	16	

Constant Name	Value	Notes
GRS-ROW-HEADINGS	32	
GRS-TILED-HEADINGS	64	
GRS-CENTERED-HEADINGS	128	
GRS-USE-TAB	256	
GRS-ADJUSTABLE-COLUMNS	512	
GRS-PAGED	1024	
GRP-ROW-DIVIDERS	1	GRID PROPERTIES
GRP-VPADDING	2	
GRP-DIVIDER-COLOR	3	
GRP-INSERTION-INDEX	4	
GRP-DATA-COLUMNS	5	
GRP-DISPLAY-COLUMNS	6	
GRP-ALIGNMENT	7	
GRP-SEPARATION	8	
GRP-COLUMN-DIVIDERS	9	
GRP-ROW-COLOR-PATTERN	10	
GRP-Y	11	
GRP-X	12	
GRP-COLUMN-COLOR	13	
GRP-ROW-COLOR	14	
GRP-CELL-COLOR	15	
GRP-COLUMN-FONT	16	
GRP-ROW-FONT	17	
GRP-CELL-FONT	18	
GRP-BITMAP	19	
GRP-BITMAP-NUMBER	20	
GRP-BITMAP-WIDTH	21	
GRP-BITMAP-TRAILING	22	
GRP-NUM-ROWS	23	
GRP-CURSOR-Y	24	
GRP-CURSOR-X	25	
GRP-CURSOR-FRAME-WIDTH	26	
GRP-VIRTUAL-WIDTH	27	
GRP-DATA-TYPES	28	
GRP-CURSOR-COLOR	29	
GRP-HEADING-COLOR	30	
GRP-HEADING-FONT	31	
GRP-HEADING-DIVIDER-COLOR	32	
GRP-START-X	33	
GRP-START-Y	34	
GRP-REGION-COLOR	35	
GRP-MASS-UPDATE	36	
GRP-HIDDEN-DATA	37	
GRP-END-COLOR	38	
GRP-FILE-POS	39	
GRP-NUM-COL-HEADINGS	40	
GRP-RECORD-TO-ADD	4097	
GRP-RESET-GRID	4098	
GRP-CELL-DATA	4099	
GRP-RECORD-TO-DELETE	4100	

Constant Name	Value	Notes
GRP-RECORD-DATA	4101	
GRP-LAST-ROW	4102	
GRP-VSCROLL-POS	4103	
GRP-HSCROLL-POS	4104	
GRP-ACTION	4105	
GRP-SEARCH-TEXT	4106	
GRP-SEARCH-OPTIONS	4107	
GRP-INSERT-ROWS	4108	
TVS-BOXED	1	TREE-VIEW STYLES
TVS-NO-BOX	2	
TVS-BUTTONS	4	
TVS-SHOW-LINES	8	
TVS-LINES-AT-ROOT	16	
TVS-SHOW-SEL-ALWAYS	32	
TVP-PARENT	1	TREE-VIEW PROPERTIES
TVP-PLACEMENT	2	
TVP-ITEM	3	
TVP-BITMAP-HANDLE	4	
TVP-BITMAP-WIDTH3	5	
TVP-ITEM-TO-ADD	4097	
TVP-ITEM-TEXT	4098	
TVP-NEXT-ITEM	4099	
TVP-ITEM-TO-DELETE	4100	
TVP-RESET-LIST	4101	
TVP-ENSURE-VISIBLE	4102	
TVP-EXPAND	4103	
TVP-ITEM-TO-EMPTY	4104	
TVP-BITMAP-NUMBER	4105	
TVP-HIDDEN-DATA	4106	
TVP-HAS-CHILDREN	4107	
WBS-NOTIFY-CHANGE	1	WEB-BROWSER STYLES
WBP-BUSY	1	WEB-BROWSER PROPERTIES
WBP-TYPE	2	
WBP-STATUS-TEXT	3	
WBP-NAVIGATE-URL	4	
WBP-PROGRESS	5	
WBP-MAX-PROGRESS	6	
WBP-GO-BACK	4097	
WBP-GO-FORWARD	4098	
WBP-GO-HOME	4099	
WBP-GO-SEARCH	4100	
WBP-REFRESH	4101	
WBP-STOP	4102	

ELASTIC COBOL -CRTVARS.DEF and CRTVARS.DEF

NO CONSTANTS ARE DEFINED BY THIS COPY FILE.

ELASTIC COBOL -FONTS.DEF and FONTS.DEF

Constant Name	Value	Notes
WFONT-SUPPORTED	1	
WFONT-CHOOSE-FONT	2	
WFONT-GET-FONT	101	
WFONT-GET-CLOSEST-FONT	102	
WFONT-DESCRIBE-FONT	106	
WFONT-FONT-SUPPORT	1	
WFONT-FULL-SUPPORT	2	
WFONTERR-UNSUPPORTED	0	
WFONTERR-CANCELLED	-1	
WFONTERR-FONT-NOT-FOUND	-2	
WFONTERR-INVALID-HANDLE	-3	
WFCHOOSE-FIXED-ONLY	1	
WFCHOOSE-INITIALIZE	2	
WFCHOOSE-EFFECTS-OK	4	

ELASTIC COBOL -KEYPROG.DEF and KEYPROG.DEF

NO CONSTANTS ARE DEFINED BY THIS COPY FILE.

ELASTIC COBOL -OPENSAVE.DEF and OPENSAVE.DEF

Constant Name	Value	Notes
OPENSAVE-SUPPORTED	1	
OPENSAVE-OPEN-BOX	2	
OPENSAVE-SAVE-BOX	3	
OPNSAVERR-UNSUPPORTED	0	
OPNSAVERR-CANCELLED	-1	
OPNSAVERR-NO-MEMORY	-1	
OPNSAVERR-NAME-TOO-LARGE	-3	

ELASTIC COBOL -PALETTE.DEF and PALETTE.DEF

Constant Name	Value	Notes

Constant Name	Value	Notes
WPALETTE-SUPPORTED	1	
WPALETTE-NUM-COLORS	2	
WPALETTE-GET-COLOR	3	
WPALETTE-SET-COLOR	4	
WPALETTE-UPDATE	5	
WPALETTE-CHOOSE-COLOR	6	
WPALETTE-SET-USER-COLOR	7	
WPCHOOSE-USE-DEFAULT	1	
WPUSER-COLOR-3D	1	
WPUSER-COLOR-BACKGROUND	2	
WPAL-NO-SUPPORT	0	
WPAL-PALETTE-SUPPORTED	1	
WPAL-FULL-SUPPORT	2	
WPERR-UNSUPPORTED	0	
WPERR-BAD-ARG	-1	
WPERR-CANCELLED	-2	

ELASTIC COBOL -SHOWMSG.DEF and SHOWMSG.DEF

Constant Name	Value	Notes
ACU-LENGTH	80	
ACU-FULL-LEN	250	

ELASTIC COBOL -STDFONTS.DEF and STDFONTS.DEF

NO CONSTANTS ARE DEFINED FOR THIS COPY FILE.

ELASTIC COBOL -WINHELP.DEF and WINHELP.DEF

Constant Name	Value	Notes
HELP-CONTEXT	1	
HELP-QUIT	2	
HELP-CONTENTS	3	
HELP-HELPONHELP	4	
HELP-SETCONTENTS	5	
HELP-CONTEXTPOPUP	8	
HELP-FORCEFILE	9	
HELP-KEY	257	
HELP-COMMAND	258	

Constant Name	Value	Notes
HELP-PARTIALKEY	261	
HELP-MULTIKEY	513	
HELP-SETWINPOS	515	
HELP-CONTEXTMENU	10	
HELP-FINDER	11	
HELP-WM-HELP	12	
HELP-SETPOPUP-POS	13	

ELASTIC COBOL -WINPRINT.DEF and WINPRINT.DEF

Constant Name	Value	Notes
WINPRINT-SUPPORTED	1	WIN$PRINTER OPCODES
WINPRINT-SETUP	2	
WINPRINT-SET-STD-FONT	3	
WINPRINT-GET-PAGE-LAYOUT	4	
WINPRINT-GET-SETTINGS-SIZE	5	
WINPRINT-GET-SETTINGS	6	
WINPRINT-SET-SETTINGS	7	
WINPRINT-SET-FONT	8	
WINPRINT-SET-LINES-PER-PAGE	9	
WINPRINT-GET-CAPABILITIES	10	
WINPRINT-PRINT-BITMAP	11	
WINPRINT-SET-MARGINS	12	
WINPRINT-GET-NO-PRINTERS	13	
WINPRINT-GET-PRINTER-INFO	14	
WINPRINT-SET-PRINTER	15	
WINPRINT-GET-CURRENT-INFO	16	
WINPRINT-SETUP-OLD	17	
WINPRINT-SET-DATA-COLUMNS	18	
WINPRINT-CLEAR-DATA-COLUMNS	19	
WINPRINT-SET-PAGE-COLUMN	20	
WINPRINT-CLEAR-PAGE-COLUMNS	21	
WINPRINT-GET-PAGE-COLUMN	22	
WPRTERR-UNSUPPORTED	0	RETURN CODES
WPRTERR-BAD-ARG	-1	
WCANCELLED	-2	
WPRTERR-BUFFER-TOO-SMALL	-3	
WPRTERR-NO-MEMORY	-4	
WPRTERR-SPOOLER-OPEN	-5	
WPRTERR-SPOOLER-CLOSED	-6	
WPRTERR-DEVICE-INCAPABLE	-7	
WPRTERR-ENUM-FAIL	-8	
WPRTERR-DRV-LOADFAIL	-9	
WPRTERR-BAD-DRIVER	-10	

Constant Name	Value	Notes
WPRTFONT-DEFAULT	1	STANDARD FONT VALUES
WPRTFONT-COURIER-12	2	
WPRTFONT-COURIER-12-COMP	3	
WPRTFONT-COURIER-10	4	
WPRTFONT-COURIER-10-COMP	5	
WPRTBITMAP-PRINTER-BITMAP	1	BITMAP FLAGS
WPRTBITMAP-SCALE-CELLS	2	
WPRTBITMAP-SCALE-INCHES	4	
WPRTBITMAP-SCALE-CENTIMETERS	8	
WPRTBITMAP-SCALE-PIXELS	16	
WPRTMARGIN-DEFAULT-MARGINS	0	MARGINS FLAGS
WPRTMARGIN-CELLS	1	
WPRTMARGIN-INCHES	2	
WPRTMARGIN-CENTIMETERS	3	
WPRTMARGIN-PIXELS	4	
WPRTSEL-ORIENT-DEFAULT	0	PRINTER SELECTION
WPRTSEL-ORIENT-PORTRAIT	1	
WPRTSEL-ORIENT-LANDSCAPE	2	
WPRTSEL-QUALITY-DEFAULT	0	
WPRTSEL-QUALITY-HIGH	-1	
WPRTSEL-QUALITY-MEDIUM	-2	
WPRTSEL-QUALITY-LOW	-3	
WPRTSEL-QUALITY-DRAFT	-4	
WPRTUNITS-CELLS	0	UNITS
WPRTUNITS-INCHES	1	
WPRTUNITS-CENTIMETERS	2	
WPRTUNITS-PIXELS	3	
WPRTALIGN-NONE	" "	ALIGNMENTS
WPRTALIGN-LEFT	"L"	
WPRTALIGN-RIGHT	"R"	
WPRTALIGN-CENTER	"C"	
WPRTALIGN-DECIMAL	"D"	
WPRTALIGN-DECIMAL-SUPPRESS	"S"	

ELASTIC COBOL -WINVERS.DEF and WINVERS.DEF

NO CONSTANTS ARE DEFINED FOR THIS COPY FILE.

Appendix G - Call Functions

CALL FUNCTIONS (cross-platform)

Elastic COBOL includes support for certain CALL functions for compatibility with other vendors. These CALL functions are implemented in a cross-platform manner independent of third-party libraries.

The supported CALL functions are listed below.

CBL_AND

Perform a bitwise 'and' of the source and destination.

General Format

CALL "CBL_AND" USING source, destination, length GIVING status

Parameters

Source	PIC X(n); source bits
Destination	PIC X(n); destination bits
Length	numeric
Status	numeric-identifier; 0 if successful, 1 if unsuccessful

CBL_COPY_FILE

Copy a physical file from a source-file to a destination-file.

General Format

CALL "CBL_COPY_FILE" USING source-file, destination-file GIVING status

Parameters

Source-file	PIC X(n); Name of file to copy
Destination-file	PIC X(n); Name of file to which to copy source-file to
Status	numeric-identifier; 0 if successful, 1 if unsuccessful.

CBL_CREATE_DIR

Create a physical directory on disk.

General Format

CALL "CBL_CREATE_DIR" USING directory-name GIVING status

Parameters

Directory-name	PIC X(n); Name of directory to create
Status	numeric-identifier; 0 if successful, 1 if unsuccessful.

CBL_DELETE_DIR

Delete a physical directory from disk.

General Format

CALL "CBL_DELETE_DIR" USING directory-name GIVING status

Parameters

Directory-name	PIC X(n); Name of directory to create
Status	numeric-identifier; 0 if successful, 1 if unsuccessful.

CBL_DELETE_FILE

Delete a physical file from disk.

General Format

CALL "CBL_DELETE_FILE" USING file-name GIVING status

Parameters

File-name	PIC X(n); Name of file to delete
Status	numeric-identifier; 0 if successful, 1 if unsuccessful

CBL_EQ

Perform a bitwise 'equality' of the source and destination.

General Format

CALL "CBL_EQ" USING source, destination, length GIVING status

Parameters

Source	PIC X(n); source bits
Destination	PIC X(n); destination bits
Length	numeric
Status	numeric-identifier; 0 if successful, 1 if unsuccessful

CBL_IMP

Perform a bitwise 'implies' of the source and destination.

General Format

CALL "CBL_IMP" USING source, destination, length GIVING status

Parameters

Source	PIC X(n); source bits
Destination	PIC X(n); destination bits
Length	numeric
Status	numeric-identifier; 0 if successful, 1 if unsuccessful

CBL_JOIN_FILENAME

Join a filename together from its constituent parts.

General Format

CALL "CBL_JOIN_FILENAME" USING

split-join-param, join-buffer, path-buffer, basename-buffer, extension-buffer GIVING status

Parameters

Split-join-param Group-item:

```
01 SPLIT-JOIN-PARAM.
    05 PARAM-LENGTH PIC X(2) COMP-X.
    05 SPLITJOIN-FLAG1 PIC X COMP-X.
    05 SPLITJOIN-FLAG2 PIC X COMP-X.
    05 PATH-START PIC X(2) COMP-X.
    05 PATH-LENGTH PIC X(2) COMP-X.
    05 BASENAME-START PIC X(2) COMP-X.
    05 BASENAME-LENGTH PIC X(2) COMP-X.
    05 EXTENSION-START PIC X(2) COMP-X.
    05 EXTENSION-LENGTH PIC X(2) COMP-X.
    05 TOTAL-LENGTH PIC X(2) COMP-X.
    05 SPLIT-BUFFER-LENGTH PIX X(2) COMP-X.
    05 JOIN-BUFFER-LENGTH PIC X(2) COMP-X.
    05 FIRST-PATH-LENGTH PIC X(2) COMP-X.
```

Join-buffer	PIC X(n)
Path-buffer	PIC X(n)
Basename-buffer	PIC X(n)
Extension-buffer	PIC X(n)
Status	numeric-identifier

CBL_NOT

Perform a bitwise 'not' of the destination.

General Format

CALL "CBL_NOT" USING destination, length GIVING status

Parameters

Destination	PIC X(n); destination bits
Length	numeric
Status	numeric-identifier; 0 if successful, 1 if unsuccessful

CBL_OR

Perform a bitwise 'or' of the source and destination.

General Format

CALL "CBL_OR" USING source, destination, length GIVING status

Parameters

Source	PIC X(n); source bits
Destination	PIC X(n); destination bits
Length	numeric
Status	numeric-identifier; 0 if successful, 1 if unsuccessful

CBL_RENAME_FILE

Rename a physical file from an old filename to a new filename.

General Format

CALL "CBL_RENAME_FILE" old-filename, new-filename GIVING status

Parameters

Old-filename	The existing filename.
New-filename	The new filename to which to rename old-filename.
Status	numeric-identifier; 0 if successful, 1 if unsuccessful

CBL_TOLOWER

Convert text to lower-case.

General Format

CALL "CBL_TOLOWER" USING destination, length GIVING status

Parameters

Destination	PIC X(length); the text to which to convert to lower-case.
Length	numeric; the length of text to convert.
Status	numeric-identifier; 0 if successful, 1 is unsuccessful, 2 if incorrect number of args

CBL_TOUPPER

Convert text to upper-case.

General Format

CALL "CBL_TOUPPER" USING destination, length GIVING status

Parameters

Destination	PIC X(length); the text to which to convert to upper-case.
Length	numeric; the length of text to convert.
Status	numeric-identifier; 0 if successful, 1 is unsuccessful, 2 if incorrect number of args

CBL_XOR

Perform a bitwise 'xor' of the source and destination.

General Format

CALL "CBL_XOR" USING source, destination, length GIVING status

Parameters

Source	PIC X(n); source bits
Destination	PIC X(n); destination bits
Length	numeric
Status	numeric-identifier; 0 if successful, 1 if unsuccessful

CBL_YIELD_RUN_UNIT

Yield the run unit. Invokes Thread.yield(). This releases the CPU early and should not be necessary on modern systems.

General Format

CALL "CBL_YIELD_RUN_UNIT"

ASCII2HEX

Converts ASCII to HEX.

General Format

CALL "ASCII2HEX" USING ascii-value, hex-value

Parameters

Ascii-value	PIC X(2)
Hex-value	PIC X(4)

ASCII2OCTAL

Converts ASCII to OCTAL.

General Format

CALL "ASCII2OCTAL" USING ascii-value, octal-value

Parameters

Ascii-value	PIC X(2)
Octal-value	PIC X(8)

HEX2ASCII

Converts HEX to ASCII.

General Format

CALL "HEX2ASCII" USING ascii-value, hex-value

Parameters

Ascii-value	PIC X(2)
Hex-value	PIC X(4)

OCTAL2ASCII

Converts OCTAL to ASCII.

General Format

CALL "OCTAL2ASCII" USING ascii-value, octal-value

Parameters

Ascii-value	PIC X(2)
Octal-value	PIC X(8)

C$ASYNCPOLL

AcuConnect function for interfacing with legacy AcuConnect code.

General Format

CALL "C$ASYNCPOLL" USING handle-of-call, state-of-call[, cobol-parameters]

Parameters

Handle-of-call	Handle; handle of CALL from C$ASYNCRUN
State-of-call	PIC S9; 0 if not completed, 1 if completed.
COBOL-parameters	A list of COBOL parameters for the remote function.

C$ASYNCRUN

AcuConnect function for interfacing with legacy AcuConnect code.

General Format

CALL "C$ASYNCRUN" USING handle-of-call, program-name, cobol-parameters

Parameters

Handle-of-call	Handle; handle of CALL
Program-name	PIC X(n)
COBOL-parameters	A list of COBOL parameters for the remote function.

C$CALLERR

Get reason why the previous CALL failed.

General Format

CALL "C$CALLERR" USING error-code[, error-message]

Parameters

Error-code	PIC X(2)
Error-message	PIC X(n)

C$CHAIN

Java does not supported chaining, so this function merely runs and exits.

General Format

CALL "C$CHAIN" USING program-name

Parameters

Program-name	PIC X(n); program name to call

C$CHDIR

Change current directory. This function only changes the System Property user.dir; it affects only those JVM's which use the System Property (such as Macintosh).

General Format

CALL "C$CHDIR" USING directory-name, status

Parameters

Directory-name	Name of directory to make current directory.
Status	numeric-identifier; 0 if successful, error number if unsuccessful.

C$COPY

Copy a physical file on disk.

General Format

CALL "C$COPY" USING source-file, destination-file[, file-type] GIVING status

Parameters

Source-file	PIC X(n); source filename.
Destination-file	PIC X(n); destination filename
File-type	PIC X; "S", "R", or "I" for sequential, relative or indexed
Status	numeric-identifier; 0 if successful, error number if unsuccessful.

C$DELETE

Delete a physical file from disk.

General Format

CALL "C$DELETE" USING file-name[, file-type] GIVING status

Parameters

File-name	PIC X(n); file-name to delete
File-type	PIC X; "S", "R", or "I" for sequential, relative or indexed
Status	numeric-identifier; 0 if successful, error number if unsuccessful.

C$CENTURY

The C$CENTURY routine returns the current century digits, generally either a 19 or a 20. This function is present for compatibility and should not be used in new code; there are COBOL ACCEPT variations that give the same information through a more standard mechanism.

General Format

CALL "C$CENTURY" USING century-item

Parameters

Century-item must be a PIC X(2) or PIC 9(2) item.

Return

Century-item holds the current century digits (19 for 1900-1999, 20 for 2000-2099).

C$DELAY

The C$DELAY function delays program execution by the given number of seconds. This delay is done in a CPU efficient manner.

General Format

CALL "C$DELAY" USING delay-seconds

Parameters

Delay-seconds must be a PIC 9(n) item, containing seconds to delay.

C$PLAYSOUND

The C$PLAYSOUND function plays an audio file to the local speaker if possible. Not all systems include audio capabilities, so this should only be used for additional effect, never as a primary means of user communication.

General Format

CALL "C$PLAYSOUND" USING audio-filename

Parameters

Audio-filename must be the filename of an audio file supported by the Java environment. In JDK 1.3 and above, this includes .au, .wav, and .mid format files.

C$SHOW

The C$SHOW function sets the state of the main window. This function is present for compatibility only and should be avoided.

General Format

CALL "C$SHOW" USING window-state

Parameters

Window-state may be:

0:	HIDE
2:	SHOW and MINIMIZE
3:	SHOW and MAXIMIZE
5:	SHOW
6:	MINIMIZE
9:	RESTORE
10:	TO FRONT
11:	TO BACK

Other window-state numbers are undefined.

C$SHOWARGS

The C$SHOWARGS function displays information about its parameters, useful mainly for debugging native calls.

General Format

CALL "C$SHOWARGS" USING [any-parameter-list]

Parameters

The parameters may be of any type and length.

C$TITLE

The C$TITLE function sets the title bar text of the main window.

General Format

CALL "C$TITLE" USING title-text

Parameters

Title-text is the text to be displayed in the title bar.

C$BARCODE

The C$BARCODE routine generates a barcode, suitable for printing or displaying on the screen using a barcode driver. The use of this routine requires that a barcode implementation be present in the CLASSPATH, and a Elastic COBOL driver for the barcode implementation.

A driver is included for Dragon Technologies' JBarcodeBean. If no DRIVER is specified, then this included driver is used. Note that the driver for JBarcodeBean is included, not JBarcodeBean itself which must be purchased separately.

Other drivers may not support all parameters listed, or may support additional parameters.

General Format

CALL "C$BARCODE" USING {parameter-name parameter-value}…

RETURNING barcode-object

Parameters

Parameter Name	Parameter Value
DRIVER	Name of class implementing com.heirloomcomputing.ecs.api.GenericBarcode
TYPE	The type of the barcode.
VALUE	The value of the barcode, its content to be rendered.
CHECK_DIGIT_BOOL	'Y' to include a check digit, 'N' to exclude the check digit.
SHOW_TEXT_BOOL	'Y' to show text in image, 'N' to exclude text from image.
NARROWEST_BAR_WIDTH_INT	Narrowest bar width, as an integer.
ANGLE_DEGREES_DOUBLE	Angle at which to render barcode, 0..360.
BARCODE_HEIGHT_INT	Height of barcode, as an integer.

BACKGROUND_COLOR	Background color.
BARCODE_BACKGROUND_COLOR	Background in barcode.
Barcode-object	A Component or Image, depending on driver.

DRIVER

Specify the DRIVER name. This name may be resolved through a program parameter to allow the driver to be specified externally. This name must be the name of a Java class file that implements com.heirloomcomputing.ecs.api.GenericBarcode (see the API documentation for more information). A DRIVER must always be the first parameter unless the default driver is used.

Special values for DRIVER include DEFAULT and DRAGON. DEFAULT currently is DRAGON. DRAGON references the Dragon Technologies' JbarcodeBean at http://www.dragontechnology.com/barcode/

TYPE

Types are determined by the driver, but where present, the following names should be implemented by the driver:

Code128

Code39

Code39_2to1

ExtendedCode39

ExtendedCode39_2to1

Interleaved25

Interleaved25_2to1

Codebar

Codebar_2to1

MSI

VALUE

The value of the barcode, its content to be rendered. The contents allowed for VALUE vary between the different barcode types. Some may only allow numeric data, alphabetic data, alphanumeric data, or any ASCII value. See the documentation for the type of driver for more information.

CHECK_DIGIT_BOOL

If a type supports a check digit, it may be included or excluded from the barcode rendering by passing 'Y' to include or 'N' to exclude. Not all barcode types allow this setting to be changed.

SHOW_TEXT_BOOL

Pass 'Y' to show the text of the VALUE within the rendered image, or 'N' to not show the text.

NARROWEST_BAR_WIDTH_INT

Setting the narrowest bar width allows the rendered image to be scaled readily to different bar width capabilities.

ANGLE_DEGREES_DOUBLE

Set the angle to a number from 0 through 360 to render it at an angle other than normal. This allows the barcode to be printed on its side, for instance.

BARCODE_HEIGHT_INT

Set the height of the barcode using an integer.

BACKGROUND_COLOR

The background color of the barcode image may be set to any standard color name.

BARCODE_BACKGROUND_COLOR

The background color within the barcode itself may be set to any standard color name.

Barcode-object

The barcode-object is the object returned from the C$BARCODE generator. The type of the call return must be OBJECT REFERENCE. It may in turn be passed to either P$COMPONENTOUT for Component objects or P$DRAWBITMAP for Image objects. The driver should document what type of barcode is returned. If barcode-object is NULL or IS an instance of com.heirloomcomputing.ecs.api.Datatype, then it is an error message and may be output using DISPLAY.

The DEFAULT or DRAGON driver returns a Component.

C$GETFREEMEMORY

Obtain the total free memory currently available to the Elastic COBOL program. This may be affected by the execution environment's memory settings as well as the physical memory of the computer and logical memory provided by the operating system.

General Format

```
CALL "C$GETFREEMEMORY" RETURNING free-memory
```

Parameters

Free-memory must be a numeric type large enough to hold the reasonable return values for the system. The value is expressed in bytes.

C$GETTOTALMEMORY

Obtain the total memory currently available to the Elastic COBOL program. This may be affected by the execution environment's memory settings as well as the physical memory of the computer and logical memory provided by the operating system.

General Format

CALL "C$GETTOTALMEMORY" RETURNING total-memory

Parameters

Total-memory must be a numeric type large enough to hold the reasonable return values for the system. The value is expressed in bytes.

C$FILEINFO

Obtain operating system information about a file.

General Format

CALL "C$FILEINFO" USING file-name, file-info GIVING status

Parameters

File-name	PIC X(n); name of file from which information is obtained
File-info	Group item:

```
01 FILEINFO-GROUP.
   05 SIZE-ITEM PIC X(8) COMP-X.
   05 DATE-ITEM PIC X(8) COMP-X.
   05 TIME-ITEM PIC X(8) COMP-X.
```

Status	Numeric-identifier; 0 if successful, 1 if unsuccessful.

C$FULLNAME

Obtain a complete filename from a partial filename.

General Format

CALL "C$FULLNAME" USING file-name, full-name[, file-info] GIVING status-code

Parameters

File-name	PIC X(n); name of file from which information is obtained.
Full-name	PIC X(n); destination for fullname
File-info	See C$FILEINFO
Status	numeric-identifier; 0 if successful, 1 if unsuccessful.

C$JUSTIFY

Justify data to left, center, or right after removing all leading and trailing spaces.

General Format

CALL "C$JUSTIFY" USING data-item[, justify-type]

Parameters

Data-item	Any data item.
Justify-type	PIC X; "L" for left, "R" for right (default), "C" for center

C$MAKEDIR

Create a directory.

General Format

CALL "C$MAKEDIR" USING directory-name GIVING status

Parameters

Directory-name	PIC X(n); directory name to create
Status	numeric-identifier; 0 if successful, 1 if unsuccessful

C$NARG

Return the number of parameters passed to the program. Avoid this function; use FUNCTION ARGUMENT-LENGTH instead as it is context sensitive.

General Format

CALL "C$NARG" USING arguments

Parameters

Arguments	numeric-identifier; the number of arguments passed

C$OPENSAVEBOX

Create an Open or Save As… dialog box.

General Format

CALL "C$OPENSAVEBOX" USING op-code, data GIVING status

Parameters

Op-code	Numeric opcode described below:
Data	Group item:

```
01 OPENSAVEBOX-GROUP-ITEM.
  05 OS-FILENAME PIC X(256).
  05 OS-FLAGS PIC 9(4) COMP-X.
```

```
05 OS-DEFAULT-EXTENSION PIC X(12).
05 OS-TITLE PIC X(80).
05 OS-FILTERS PIC X(512).
05 OS-DEFAULT-FILTER PIC 9(4) COMP-X.
05 OS-DEFAULT-DIRECTORY PIC X(128).
05 OS-BASENAME PIC X(128).
```

Status	Signed-numeric-item; 1 is successful, <=0 is unsuccessful.

Op-code Support

OPENSAVE-SUPPORTED (1)	If the system supports the CALL "C$OPENSAVEBOX" functionality, status is set to 1, otherwise status is set to 0.
OPENSAVE-OPEN-BOX (2)	Create an Open dialog box.
OPENSAVE-SAVE-BOX (3)	Create a Save As… dialog box.

OPENSAVEBOX-GROUP-ITEM

OS-FILENAME	Set to the default filename on input, set to the filename selected by the user on output.
OS-FLAGS	Reserved, set to 0.
OS-DEFAULT-EXTENSION	Set to the default extension, or spaces for none.
OS-TITLE	Set to the title for the dialog box.
OS-FILTERS	Reserved, set to spaces.
OS-DEFAULT-FILTER	Reserved, set to zero.
OS-DEFAULT-DIRECTORY	Set to the default directory.
OS-BASENAME	Set to the basename on output, the filename with no directory information.

C$RUN

Run an external, operating system program.

General Format

CALL "C$RUN" USING command-line GIVING status

Parameters

Command-line	PIC X(n); command-line including program-name and parameters
Status	signed-numeric-identifier; 0 if successful, -1 if unsuccessful.

C$SLEEP

Sleep the CPU for a number of seconds. Always use a sleep method to delay rather than a loop.

General Format

CALL "C$SLEEP" USING seconds

Parameters

Seconds	The number of seconds to sleep (may be fractional).

C$SYSTEM

Run an external, operating system program.

General Format

CALL "C$SYSTEM" USING command-line[, flags] GIVING status

Parameters

Command-line	PIC X(n); command-line to execute including program-name and parameters
Flags	May be added from the following: CSYS-ASYNC (1) CSYS-NO-IO (2) CSYS-MAXIMIZED (4) CSYS-MINIMIZED (8) CSYS-COMPATIBILITY (16) CSYS-HIDDEN (32) CSYS-SHELL (64)
Status	signed-numeric-identifier; program's exit status

Only CSYS-ASYNC (1) is currently supported. All other flag values are reserved for future use.

C$TOLOWER

Convert text to lower-case.

General Format

CALL "C$TOLOWER" USING destination, length GIVING status

Parameters

Destination	PIC X(length); the text to which to convert to lower-case.
Length	numeric; the length of text to convert.
Status	numeric-identifier; 0 if successful, 1 is unsuccessful, 2 if incorrect number of args

C$TOUPPER

Convert text to upper-case.

General Format

```
CALL "C$TOUPPER" USING destination, length GIVING status
```

Parameters

Destination	PIC X(length); the text to which to convert to upper-case.
Length	Numeric; the length of text to convert.
Status	Numeric-identifier; 0 if successful, 1 is unsuccessful, 2 if incorrect number of args

DSRUN

The DSRUN command is available only in conjunction with the DialogSysToJava product

The DSRUN command invokes DialogSysToJava, controlling the graphics display. The DialogSysToJava's runtime library must be available in the CLASSPATH to be active, as well as any files generated by the DialogSysToJava conversion.

The DSRUN command is available by the aliases DSCRUN and DSGRUN as well.

See the DialogSysToJava product documentation for more information.

General Format

```
CALL "DSRUN" USING control-block data-block [event-block]
```

P$CLEARDIALOG

Clear printer dialog values to default values.

General Format

```
CALL "P$CLEARDIALOG"
```

Parameters

None

P$DISPLAYDIALOG

Display the printer dialog to the user. This is automatically done by an OPEN if enable dialog is true; this routine should generally not be called directly.

This is available in Java 1.2+ only. In Java 1.1, the printer dialog is required.

General Format

CALL "P$DISPLAYDIALOG" RETURNING dialog-return

Parameters

Dialog-return is from the following values:

0	OK
1	CANCELED
2	ERROR

P$ENABLEDIALOG

Enable the printer dialog to be displayed to the user automatically upon next open. The default is that the printer dialog is enabled.

This is available in Java 1.2+ only. In Java 1.1, the printer dialog is required.

General Format

CALL "P$ENABLEDIALOG" [USING dialog-enable]

Parameters

Dialog-enable, if specified, may be set to 'Y' for enable or 'N' for disable. When omitted, this is 'Y'.

P$DISABLEDIALOG

Disable the printer dialog from being displayed automatically to the user upon next open.

This is available in Java 1.2+ only. In Java 1.1, the printer dialog is required.

General Format

CALL "P$DISABLEDIALOG" [USING dialog-disable]

Parameters

Dialog-disable, if specified, may be set to 'Y' for disable or 'N' for enable. When omitted, this is 'Y'.

P$GETDIALOG

Get printer dialog attributes.

General Format

CALL "P$GETDIALOG" USING {parameter-name parameter-value}…

Parameters

Parameter-name must be a name supported by the current printer driver. The JDK 1.2+ driver supports:

SUPPORTED	List of supported attributes.
JOB_NAME	Name of job.
USER_NAME	Name of user submitting job.

Parameter-value holds the return value from parameter-name.

P$SETDIALOG

Set printer dialog attributes.

General Format

CALL "P$SETDIALOG" USING {parameter-name parameter-value}…

Parameters

Parameter-name must be a name supported by the current printer driver. The JDK 1.2+ driver supports:

COPIES	Number of copies to print.

Parameter-value holds the value to which to set parameter-name.

P$DRAWBITMAP

Draw a bitmap image on the current printer page.

General Format

CALL "P$DRAWBITMAP" USING
image-resource
[xpos ypos] [position-mode] [position-units]
[size-width size-height] [size-units]

Parameters

The image-resource is a resource name (typically a filename) specifying the image to print on the current page. The image-resource must be of a type supported by Elastic COBOL, currently .gif, .jpg, .bmp, .ico and on JDK 1.3+ .png.

Xpos and ypos are the x and y position on the page to print, either in absolute position-units or relative position-units from the current position.

Position-mode specifies 'A' for absolute or 'R' for relative positioning.

Position-units specifies the units in which the xpos and ypos values are expressed. If omitted, then default units are used.

Size-width and size-height are the width and height of the print in size-units. If omitted, then the natural size for the bitmap is used.

Size-units specifies the units in which size-width and size-height are expressed. If omitted, then position-units is used.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$DRAWBOX

Draw a box or rectangle on the current printer page.

General Format

CALL "P$DRAWBOX" USING [xpos ypos] [position-mode] [position-units] [size-width size-height] [size-units] [shade-boolean] [3-d-boolean]

CALL "P$DRAWBOX" USING [xpos ypos] [position-mode] [position-units] [size-width size-height] [size-units] [shade-boolean] [arc-width arc-height]

Parameters

Xpos and ypos are the x and y position on the page to print, either in absolute position-units or relative position-units from the current position.

Position-mode specifies 'A' for absolute or 'R' for relative positioning.

Position-units specifies the units in which the xpos and ypos values are expressed. If omitted, then default units are used.

Size-width and size-height are the width and height of the print in size-units. If omitted, then the natural size for the bitmap is used.

Size-units specifies the units in which size-width and size-height are expressed. If omitted, then position-units is used.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

Shade-boolean may be 'Y' to shade the box, 'N' to not shade the box.

3-d-boolean may be 'Y' for raised 3-d, or 'N' for lowered 3-d. If not present, the box is not 3-d.

Arc-width and arc-height determines the width and height in pixels of the rounded portion of the rectangle.

P$DRAWOVAL

Draw an oval or circle on the current printer page.

General Format

CALL "P$DRAWOVAL" USING [xpos ypos] [position-mode] [position-units] [size-width size-height] [size-units] [shade-boolean] [start-angle end-angle]

Parameters

Xpos and ypos are the x and y position on the page to print, either in absolute position-units or relative position-units from the current position.

Position-mode specifies 'A' for absolute or 'R' for relative positioning.

Position-units specifies the units in which the xpos and ypos values are expressed. If omitted, then default units are used.

Size-width and size-height are the width and height of the print in size-units. If omitted, then the natural size for the bitmap is used. If width and height are the same, then the result is a circle.

Size-units specifies the units in which size-width and size-height are expressed. If omitted, then position-units is used.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

Shade-boolean may be 'Y' to shade the box, 'N' to not shade the box.

3-d-boolean may be 'Y' for raised 3-d, or 'N' for lowered 3-d. If not present, the box is not 3-d.

Start-angle and end-angle are used to create arcs rather than ovals. This is particularly useful for drawing wedges.

P$DRAWLINE

Draw a line on the current printer page between two co-ordinates.

General Format

CALL "P$DRAWLINE" USING
[x1 y1] [position-mode-1] [position-units-1]
[x2 y2] [position-mode-1] [position-units-2]

Parameters

X1 and y1 are the x and y position on the page to start the line, either in absolute position-units or relative position-units from the current position.

X2 and y2 are the x and y position on the page to end the line, either in absolute position-units or relative position-units from the current position.

Position-mode specifies 'A' for absolute or 'R' for relative positioning.

Position-units specifies the units in which the xpos and ypos values are expressed. If omitted, then default units are used.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$GETPOSITION

Get the current printer drawing position from the printer driver.

General Format

CALL "P$GETPOSITION" USING xpos ypos [units]

Parameters

Set xpos and ypos to current drawing x and y positions, in terms of units.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$LINETO

Draw a line from the current drawing position to the given co-ordinates on the current printer page.

General Format

CALL "P$LINETO" x2 y2 [units]

Parameters

X2 and y2 are the x and y position on the page to end the line, either in absolute position-units or relative position-units from the current position.

Position-mode specifies 'A' for absolute or 'R' for relative positioning.

Position-units specifies the units in which the xpos and ypos values are expressed. If omitted, then default units are used.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$MOVETO

Move the current printer drawing position to the given co-ordinates on the current printer page.

General Format

CALL "P$MOVETO" USING x1 y1 [position-mode] [position-units]

Parameters

X1 and y1 are the x and y position on the page to start the line, either in absolute position-units or relative position-units from the current position.

Position-mode specifies 'A' for absolute or 'R' for relative positioning.

Position-units specifies the units in which the xpos and ypos values are expressed. If omitted, then default units are used.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$SETBOXSHADE

Set the printer shading color.

General Format

CALL "P$SETBOXSHADE" color-name intensity-percent

Parameters

Color-name is a standard color-name.

Intensity-percent is the percentage of the color to use.

P$SETPEN

Set the printer drawing pen's attributes.

General Format

CALL "P$SETPEN" USING style width color-name

Parameters

Style is ignored by the current printer driver.

Width is ignored by the current printer driver.

Color-name is a standard color name.

P$SETPOSITION

Set the current printer drawing position.

General Format

CALL "P$SETPOSITION" USING [xpos ypos] [position-mode] [position-units]

Parameters

Xpos and ypos are the x and y position on the page to print, either in absolute position-units or relative position-units from the current position.

Position-mode specifies 'A' for absolute or 'R' for relative positioning.

Position-units specifies the units in which the xpos and ypos values are expressed. If omitted, then default units are used.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$EJECT

Eject the current page from the printer, feeding in the next page.

General Format

```
CALL "P$EJECT"
```

P$CLOSE

Close the printer driver, automatically ejecting the current page.

This is implicitly done by a CLOSE of a file descriptor referencing the 'printer:' device.

General Format

```
CALL "P$CLOSE"
```

P$OPEN

Open the printer driver, making a connection to the printer. This may automatically show a printer dialog box, depending on settings and version of Java.

This is implicitly done by an OPEN of a file descriptor referencing the 'printer:' device.

General Format

```
CALL "P$OPEN" [USING filename] RETURNING open-success
```

Parameters

Filename is the same filename that is used in the remainder of the printer:filename when opening the printer using the standard COBOL OPEN verb. If it includes a question mark (?) character, then the printer dialog is shown automatically.

Parameters may be specified as parameter-name=parameter-value within the filename, using the slash (/) character to separate parameters.

Parameter values recognized by some printer drivers include:

FONT	Name of the font.
SIZE	Size of the font (height).
COLS	Number of columns.
ROWS	Number of rows.
ALIGNX	Horizontal alignment in pixels. (Obsolete)
ALIGNY	Vertical alignment in pixels. (Obsolete)
MARGINX	Left margin in pixels. (Obsolete)
MARGINY	Top margin in pixels. (Obsolete)
DPI	Dots Per Inch (used to calculate device-units, cannot be used to set DPI)
BOLD	Bold font. (Obsolete)
ITALIC	Italic font. (Obsolete)
PLAIN	Plain font. (Obsolete)
GRAPHICS	Enable graphics escap, set code to parameter-value. (Obsolete)
TEXT	Disable graphics escape
HEIGHT-ADJUST	(Default .97) (Obsolete)
PAGE-WIDTH-ADJUST	(Default .93) (Obsolete)
PAGE-HEIGHT-ADJUST	(Default .91) (Obsolete)
DIMENSION	(Default .825) (Obsolete)

Open-success will be zero (0) for success, or negative (<0) for failure.

P$CLEARFONT

Set the printer font back to the default font.

General Format

```
CALL "P$CLEARFONT"
```

P$GETTEXTEXTENT

Get the dimensional extent of a piece of text, as rendered in the current font.

General Format

```
CALL "P$GETTEXTEXTENT" USING text size-width size-height [units]
```

Parameters

Text is the text from which to determine the extent.

Size-width is the width in units of the given text.

Size-height in the height in units of the given text.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$GETTEXTMETRICS

Get attributes of the current printer font.

General Format

CALL "P$GETTEXTMETRICS" USING {parameter-name parameter-value}…

Parameters

Parameter-name must be one of the following:

HEIGHT	Returns the height of the font.
ASCENT	Returns the ascent of the font.
MAXIMUM_ASCENT	Returns the maximum ascent of the font.
DESCENT	Returns the descent of the font.
MAXIMUM_DESCENT	Returns the maximum descent of the font.
LEADING	Returns the leading of the font.
TYPICAL_CHARACTER_WIDTH	Returns the width of the character zero (0).
MAXIMUM_CHARACTER_WIDTH	Returns the width of the widest character.

Parameter-value is filled in with the value of its parameter-name.

P$GETTEXTPOSITION

Get the current printer text drawing position.

General Format

CALL "P$GETTEXTPOSITION" xpos ypos [alignment] [units]

Parameters

Xpos and ypos are filled with the X and Y drawing position.

Alignment is either 'T' for Top or 'B' for Bottom.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$SETDEFAULTALIGNMENT

Set the default printer alignment for text.

General Format

CALL "P$SETDEFAULTALIGNMENT" USING alignment

Parameters

Alignment is either 'T' for Top or 'B' for Bottom, referring to the top or bottom of the font.

P$SETFONT

Set the current printer font.

General Format

CALL "P$SETFONT" USING {parameter-name parameter-value}…

Parameters

Parameter-name must be one of the following:

HEIGHT	Set the font height.
ITALIC	Set the font to be italic if parameter-value is 'Y', or not italic if 'N'.
BOLD	Set the font to be bold if parameter-value is 'Y', or not bold if 'N'.
UNDERLINE	Set the font to be underlined if parameter-value is 'Y', or not underlined if 'N'.
STRIKE_OUT	Set the font to strike-out if parameter-value is 'Y', or not strike-out if 'N'.
NAME	Set the font name to parameter-value.

P$SETLINEEXTENDMODE

Set the vertical spacing to use when outputting a carriage-return (without linefeed) to the printer. This defaults to zero (0).

General Format

CALL "P$SETLINEEXTENDMODE" USING space-amount [units]

Parameters

Space-amount is the amount of space in units to advance during a carriage-return.

P$SETTABSTOPS

Set the printer tab positions.

General Format

CALL "P$SETTABSTOPS" USING tab-stop-increment [units]

Parameters

A tab is placed at every tab-stop-increment on the page, where tab-stop-increment is expressed in units.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$SETTEXTCOLOR

Set the current printer text color.

General Format

CALL "P$SETTEXTCOLOR" USING color-name

Parameters

Color-name is a standard color name.

P$SETPAINTMODE

Set the current printer drawing mode.

General Format

CALL "P$SETPAINTMODE" [USING xor-color-name]

Parameters

If there are no parameters, then paint mode (normal) is used.

If xor-color-name is specified, then the current color is exclusive or'ed with the given standard color name to create the actual painting color.

P$SETTEXTPOSITION

Set the current printer text position.

General Format

CALL "P$SETTEXTPOSITION" USING xpos ypos [alignment] [position-mode] [position-units]

Parameters

Xpos and ypos are the x and y position on the page to print, either in absolute position-units or relative position-units from the current position.

Alignment is 'T' for Top, or 'B' for Bottom.

Position-mode specifies 'A' for absolute or 'R' for relative positioning.

Position-units specifies the units in which the xpos and ypos values are expressed. If omitted, then default units are used.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$TEXTOUT

Output text to the printer.

General Format

CALL "P$TEXTOUT" USING text [xpos ypos] [position-mode] [position-units]

[box-boolean] [shade-boolean]

Parameters

Xpos and ypos are the x and y position on the page to print, either in absolute position-units or relative position-units from the current position.

Position-mode specifies 'A' for absolute or 'R' for relative positioning.

Position-units specifies the units in which the xpos and ypos values are expressed. If omitted, then default units are used.

Box-boolean is 'Y' to place the text in a box, 'N' to not place the text in a box.

Shade-boolean is 'Y' to shade the text box, 'N' to not shade the text box.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$COMPONENTOUT

Output a graphical Component to the printer. This is generally used for outputting barcodes generated from C$BARCODE.

General Format

CALL "P$COMPONENTOUT" USING graphics-object xpos ypos [position-mode] [position-units]

[box-boolean] [shade-boolean] [size-width] [size-height] [size-units]

Parameters

The graphics-object must be an OBJECT REFERENCE, referencing a Component or Image. This is generally used for output from the C$BARCODE function.

Xpos and ypos are the x and y position on the page to print, either in absolute position-units or relative position-units from the current position.

Position-mode specifies 'A' for absolute or 'R' for relative positioning.

Position-units specifies the units in which the xpos and ypos values are expressed. If omitted, then default units are used.

Box-boolean is 'Y' to place the text in a box, 'N' to not place the text in a box.

Shade-boolean is 'Y' to shade the text box, 'N' to not shade the text box.

Size-width and size-height are the width and height of the print in size-units. If omitted, then the natural size for the bitmap is used.

Size-units specifies the units in which size-width and size-height are expressed. If omitted, then position-units is used.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$SETDEFAULTMODE

Set the default printer mode.

General Format

CALL "P$SETDEFAULTMODE" USING mode

Parameters

Mode may be 'A' for absolute or 'R' for relative. This determines the default mode when not specified in other printing functions.

P$SETDEFAULTUNITS

Set the default unit of printer measurement.

General Format

CALL "P$SETDEFAULTUNITS" USING units

Parameters

Units is the default units to be used in other printing functions when omitted.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$SETLEFTMARGIN

Set the printer left margin for this page.

General Format

```
CALL "P$SETLEFTMARGIN" USING size-width [units]
```

Parameters

Set the left margin to size-width, in terms of units.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$SETTOPMARGIN

Set the printer top margin for succeeding pages.

General Format

```
CALL "C$SETTOPMARGIN" size-height [units]
```

Parameters

Set the top margin to size-height, in terms of units.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$SETBOTTOMMARGIN

Set the printer bottom margin.

General Format

```
CALL "P$SETBOTTOMMARGIN" size-height [units]
```

Parameters

Set the bottom margin to size-height, in terms of units.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.

'D'	for device-units.
'P'	for pixels.

P$ENABLEESCAPESEQUENCES

Enable escape sequences in text output.

Basic printer escape sequences are supported by the printer driver when enabled.

General Format

CALL "P$ENABLEESCAPESEQUENCES"

P$DISABLEESCAPESEQUENCES

Disable escape sequences in text output.

General Format

CALL "P$DISABLEESCAPESEQUENCES"

P$GETDEVICECAPABILITIES

Get printer device capabilities.

General Format

CALL "P$GETDEVICECAPABILITIES" USING {parameter-name parameter-value}…

Parameters

TECHNOLOGY	Returns the driver in use.
SUPPORTED	Returns list of supported properties.
HORIZONTAL_AREA	Returns width in millimeters.
VERTICAL_AREA	Returns height in millimeters.
HORIZONTAL_RESOLUTION	Returns width in dots.
VERTICAL_RESOLUTION	Returns height in dots.
HORIZONTAL_INCHES	Returns width in inches.
VERTICAL_INCHES	Returns height in inches.
PHYSICAL_WIDTH	Returns width in device-units.
PHYSICAL_HEIGHT	Returns height in device-units.
LOGICAL_PIXELS_X	Returns pixels-per-inch horizontal supported.
LOGICAL_PIXELS_Y	Returns pixels-per-inch vertical

supported.

P$GETHANDLE

Get the handle number of the current printer.

General Format

CALL "P$GETHANDLE" USING handle

Parameters

Fill handle with the current printer number.

P$SETHANDLE

Set the current printer to the given printer handle number.

General Format

CALL "P$SETHANDLE" USING handle

Parameters

Set the current printer to the printer number described by handle.

P$CONNECTORTHICKNESS

Set the thickness of connectors on the printed page.

General Format

CALL "P$CONNECTORTHICKNESS" USING connector-width [units]

Parameters

Connector-width is the width of the connector pieces in terms of units.

Units may be any of the following:

'I'	for inches
'M'	for metric centimeters.
'C'	for character positions.
'D'	for device-units.
'P'	for pixels.

P$CONNECTOR

Place a connector on the current printer page. Connectors are suitable for creating lines forms, where line drawing art may have been used previously.

Before a page is ejected, all connectors on a page are connected wherever possible. A connector may be connected to its left, right, up or down. A connector may only make _one_ such connection per direction. No connections are made diagonally. No connector is shown if it has nothing to connect to.

General Format

CALL "P$CONNECTOR" USING xpos ypos [connector-dir] [units] [connector-type]

Parameters

Connector-dir is one or more of the following:

L	Left
R	Right
U	Up
D	Down
H	Horizontal (Left & Right)
V	Vertical (Up & Down)
A	All (Left, Right, Up & Down)
-	Horizontal (Left & Right)
|	Vertical (Up & Down)
+	All (Left, Right, Up & Down)

Connector-type is one of the following:

S	Single
D	Double
R	Round

P$COMMAND

The P$COMMAND passes an arbitrary command string to the COBOL printer driver. This is used for supporting unusual commands or commands that would not be available in most printer drivers.

The entire command is sent as a single text string.

General Format

CALL "P$COMMAND" USING command-string

The JDK 1.2 printer driver supports the following commands:

SCALE sx sy

SHEAR shx shy

ROTATE theta

ROTATE theta x y

TRANSFORM m00 m10 m01 m11 m02 m12

Scale is the Affine Transform:

```
[       sx      0       0       ]
[       0       sy      0       ]
[       0       0       1       ]
```

Shear is the Affine Transform:

```
[       1       shx     0       ]
[       shy     1       0       ]
[       0       0       1       ]
```

Rotate is the Affine Transform:

```
[       cos(theta)      -sin(theta)     0       ]
[       sin(theta)      cos(theta)              0       ]
[       0               0                       1       ]
```

Transform is the Affine Transform:

```
[       m00     m01     m02     ]
[       m10     m11     m12     ]
[       0       0       1       ]
```

W$BITMAP

Operate on bitmaps, graphical images, in any supported format, including BMP, GIF and JPG. All Elastic COBOL components that support bitmaps also support a BITMAP-VALUE clause allowing the bitmap to be directly specified without requiring use of this function. Use the BITMAP-VALUE clause rather than this function.

General Format

CALL "W$BITMAP" USING WBITMAP-DISPLAY, name, row, column, flags GIVING bitmap-handle

CALL "W$BITMAP" USING WBITMAP-DESTROY, bitmap-destroy-handle

CALL "W$BITMAP" USING WBITMAP-LOAD, name, bitmap-handle GIVING bitmap-handle

Parameters

Op-code	numeric, from one of the following: WBITMAP_DISPLAY (1) WBITMAP_DESTROY (2) WBITMAP_LOAD (3)
Name	PIC X(n); name of bitmap to display
Row	numeric; row to place upper-left corner
Column	numeric; column to place upper-left corner
Flags	numeric; if set to WBITMAP-NO-FILL (1), inhibits background filling.
Bitmap-destroy-handle	handle; handle previously returned by W$BITMAP to destroy
Bitmap-handle	handle; bitmap handle returned referring to the created bitmap

W$FONT

Operate on fonts. All Elastic COBOL components that supports fonts also support a FONT-VALUE clause allowing the font to be directly specified without requiring use of this function. Use the FONT-VALUE clause rather than this function.

General Format

CALL "W$FONT" USING op-code, font-handle, font-data GIVING status

Parameters

Op-code	numeric, from one of the following: WFONT-SUPPORTED (1) WFONT-GET-FONT (101) WFONT-GET-CLOSEST-FONT (102) WFONT-DESCRIBE-FONT (106) WFONT-CHOOSE-FONT (2)
Font-handle	handle; identifier to hold or reference font handle.
Font-data	Group item:

```
01 FONT-DATA.
    05 FONT-FACE-DATA.
        10 FONT-DEVICE HANDLE.
            88 DEVICE-CONSOLE VALUE 0.
            88 DEVICE-WIN-PRINTER VALUE 1.
        10 FONT-NAME PIC X(33).
        10 FONT-CHARACTER-SET PIC X COMP-X.
            88 CHARACTER-SET-IRRELEVANT VALUE 0.
            88 CHARACTER-SET-DEFAULT VALUE 1.
            88 CHARACTER-SET-WIN-OEM VALUE 2.
            88 CHARACTER-SET-WIN-SYMBOL VALUE 3.
            88 CHARACTER-SET-WIN-SHIFTJIS VALUE 4.
        10 FONT-SIZE PIC X COMP-X.
        10 FONT-BOLD-STATE PIC X COMP-X.
```

```
           88 FONT-BOLD VALUE 1, FALSE 0.
       10 FONT-ITALIC-STATE PIC X COMP-X.
           88 FONT-ITALIC VALUE 1, FALSE 0.
       10 FONT-UNDERLINE-STATE PIC X COMP-X.
           88 FONT-UNDERLINE VALUE 1, FALSE 0.
       10 FONT-STRIKEOUT-STATE PIC X COMP-X.
           88 FONT-STRIKEOUT VALUE 1, FALSE 0.
       10 FONT-PITCH-STATE PIC X COMP-X.
           88 FONT-FIXED-PITCH VALUE 1, FALSE 0.
       10 FONT-FAMILY PIC X COMP-X.
           88 FAMILY-IRRELEVANT VALUE 0.
           88 FAMILY-MODERN VALUE 1.
           88 FAMILY-ROMAN VALUE 2
           88 FAMILY-SWISS VALUE 3.
           88 FAMILY-SCRIPT VALUE 4.
           88 FAMILY-DECORATIVE VALUE 5.
   05 FONT-CHOOSE-DATA.
       10 FONT-CHOOSE-FLAGS PIC X COMP-X.
       10 FONT-CHOOSE-MINIMUM-SIZE PIC X COMP-X.
       10 FONT-CHOOSE-MAXIMUM-SIZE PIC X COMP-X.
       10 FONT-CHOOSE-RED PIC X COMP-X.
       10 FONT-CHOOSE-GREEN PIC X COMP-X.
       10 FONT-CHOOSE-BLUE PIC X COMP-X.
       10 FONT-CHOOSE-COLOR-NUMBER PIC X COMP-X.
```

status	signed-numeric-item; 1 for successful, 0 for unsuccessful.

WFONT-SUPPORTED (1)	Check if the runtime system supports W$FONT; return 0 if not, 1 if supported except for choose font, 2 if supported including choose-font.
WFONT-GET-FONT (101)	Get the font most closely matching font-data.
WFONT-GET-CLOSEST-FONT (102)	Get the font most closely matching font-data.
WFONT-DESCRIBE-FONT (106)	Fill in font-data using the font in font-handle.
WFONT-CHOOSE-FONT (2)	Present user with a dialog box allowing selection of a font.

W$MENU

Operate on menus. Elastic COBOL supports a MENU component directly without requiring use of this function. Use the MENU component directly rather than this function.

General Format

CALL "W$MENU" USING op-code, parameters GIVING status

Parameters

Op-code — numeric, from one of the following:

WMENU-NEW (1)
WMENU-DESTROY (2)
WMENU-ADD (3)
WMENU-CHANGE (4)
WMENU-DELETE (5)
WMENU-CHECK (6)
WMENU-UNCHECK (7)
WMENU-ENABLE (8)
WMENU-DISABLE (9)
WMENU-SHOW (10)
WMENU-GET-MENU (11)
WMENU-INPUT (12)
WMENU-BLOCK (13)
WMENU-UNBLOCK (14)
WMENU-GET-BLOCK (15)
WMENU-SET-BLOCK (16)
WMENU-RELEASE (17)
WMENU-GET-CONFIGURATION (18)
WMENU-SET-CONFIGURATION (19)
WMENU-REFRESH (20)
WMENU-DESTROY-DELAYED (21)
WMENU-GET-DELAYED-FLAG (22)
WMENU-SET-DELAYED-FLAG (23)
WMENU-NEW-POPUP (26)
WMENU-POPUP (27)

Status — PIC S9(9); return result from W$MENU

WMENU-NEW (1)	Construct a new menu bar.
WMENU-DESTROY (2)	Destroy an existing menu.
WMENU-ADD (3)	Add one menu to another menu; takes additional Parameters Menu-handle, position, flags, text, id, submenu.
WMENU-CHANGE (4)	Change a menu; takes additional Parameters Menu-handle, position, flags, text, id, submenu.
WMENU-DELETE (5)	Delete a menu; takes additional parameters: Menu-handle, id.
WMENU-CHECK (6)	Checkmark a menu; takes additional parameters: Menu-handle, id.
WMENU-UNCHECK (7)	Remove checkmark from menu; takes additional parameters: Menu-handle, id.
WMENU-ENABLE (8)	Enable a menu; takes additional parameters: Menu-handle , id.
WMENU-DISABLE (9)	Disable a menu; takes additional parameters: Menu-handle, id.
WMENU-SHOW (10)	Display a menu; takes additional parameters: Menu-handle.

WMENU-GET-MENU (11)	Returns handle of currently displayed menu.
WMENU-BLOCK (13)	Suppress menu input.
WMENU-UNBLOCK (14)	Stop suppressing menu input.
WMENU-GET-BLOCK (15)	Get blocking count, useful before setting custom blocking count.
WMENU-SET-BLOCK (16)	Set blocking count. This is used for restoring a previously gotten blocking count.
WMENU-RELEASE (17)	Logically remove menu from screen.
WMENU-GET-CONFIGURATION (18)	Get menu handler's configuration. Reserved for future use.
WMENU-SET-CONFIGURATION (19)	Set menu handler's configuration. Reserved for future use.
WMENU-REFRESH (20)	Visually refresh a menu.
WMENU-DESTROY-DELAYED (21)	Destroy if not shown, or destroy upon next show.
WMENU-NEW-POPUP (26)	Construct a new pop-up menu.
WMENU-POPUP (27)	Display a popup-menu and wait for user response.

W$MESSAGEBOX

Create a message box. Avoid this function call; use DISPLAY MESSAGE BOX instead.

General Format

CALL "W$MESSAGEBOX" USING text, title, mode GIVING status

Parameters

Text — PIC X(n); text to display.

Title — PIC X(n); title of message box.

Mode — numeric; added from one of each group:

0	OK
1	YES-NO
2	OK-CANCEL
3	YES-NO-CANCEL
256	WARNING
512	ERROR
4096	DEFAULT-2
8192	DEFAULT-3

Status — numeric; result of user selection

W$PALETTE

Operate on the color palette.

General Format

CALL "W$PALETTE" USING op-code, palette-data GIVING palette-result

Parameters

Op-code numeric

Palette-data Group item:

```
01 PALETTE-GROUP.
   05 PAL-COLOR-ID PIC X COMP-X.
   05 PAL-FLAGS REDEFINES PAL-COLOR-ID PIC X COMP-X.
   05 PAL-RED PIC X COMP-X.
   05 PAL-USER-COLOR-ID REDEFINES PAL-RED PIC X COMP-X.
   05 PAL-GREEN PIC X COMP-X.
    05 PAL-BLUE PIC X COMP-X.
```

Palette-result signed-numeric-identifier; 1 is success, 0 is failure.

Op-codes

WPALETTE-SUPPORTED (1)	Returns: WPAL-NO-SUPPORT (0) WPAL-PALETTE-SUPPORTED (1) WPAL-FULL-SUPPORT (2)
WPALETTE-NUM-COLORS (2)	Returns 2, 16, 256, or 32767. True-color machines return 32767.
WPALETTE-GET-COLOR (3)	Get color RGB corresponding to the COBOL color number (1-16) in PAL-COLOR-ID.
WPALETTE-SET-COLOR (4)	Set color RGB corresponding to the COBOL color number (1-16) in PAL-COLOR-ID.
WPALETTE-UPDATE (5)	Force color changes to be updated immediately (ignored).
WPALETTE-CHOOSE-COLOR (6)	Allow user to select a color from a palette.
WPALETTE-SET-USER-COLOR (7)	Set a system color (ignored).

WIN$PLAYSOUND

Play an audio clip. The audio clip may be in any format supported by the JDK. In JDK 1.3, this includes .WAV, .AU, .MID, .RMI and many more. This will function on any platform that will play audio.

General Format

CALL "WIN$PLAYSOUND" USING sound-name, sound-flags GIVING sound-status

Parameters

Sound-name	PIC X(n); name of sound resource to play
Sound-flags	numeric; added together from the following table SND-SYNC (0) SND-ASYNC (1) SND-LOOP (8) SND-NOSTOP (16)
Sound-status	signed-numeric-identifier; -1 is N/A, 0 is failure, 1 is success. Currently, sound-flags is ignored.

WIN$VERSION

Determine the version of Windows. Avoid this function as it is not very meaningful on non-Windows platforms.

General Format

CALL "WIN$VERSION" USING winversion

Parameters

Winversion Group item:

```
01 WINVERSION-GROUP-ITEM.
    05 WINDOWS-MAJOR-VERSION PIC X COMP-X.
    05 WINDOWS-MINOR-VERSION PIC X COMP-X.
    05 WNIDOWS-PLATFORM.
        88 WINDOWS-3-1   VALUE 1.
        88 WINDOWS-95    VALUE 2.
        88 WINDOWS-9X    VALUE 2.
        88 WINDOWS-NT    VALUE 3.
    05 WINDOWS-WORDSIZE PIC X COMP-X.
        88 WORDSIZE-16   VALUE 1.
        88 WORDSIZE-32   VALUE 2.
```

LIB$GET_SYMBOL

Get a symbol from the System Properties (global) or program configuration (local).

General Format

CALL "LIB$GET_SYMBOL" USING sym-name, sym-value[, sym-size[, sym-location]]

Parameters

Sym-name	PIC X(n); name of symbol
Sym-value	PIC X(n); returned value of symbol if found

Sym-size	numeric-identifier; number of characters returned
Sym-location	numeric-identifier; where symbol was found, 1 if local, 2 if global

LIB$SET_SYMBOL

Set a symbol in the System Properties (global) or program configuration (local).

General Format

CALL "LIB$SET_SYMBOL" USING sym-name, sym-value, sym-location

Parameters

Sym-name	PIC X(n); name of symbol
Sym-value	PIC X(n); value of symbol
Sym-location	PIC X or 9; 1 if local, 2 if global

M$ALLOC

Allocate memory from the heap to use in the COBOL program.

General Format

CALL "M$ALLOC" USING memory-size, memory-address

Parameters

Memory-size	numeric; number of bytes to allocate
Memory-address	Pointer; memory area to put data

M$FREE

Free memory previously allocated from the heap for use in the COBOL program. This function is unnecessary as Elastic COBOL is running in a garbage-collection environment, where any memory no longer referenced is automatically freed.

General format

CALL "M$FREE" USING memory-address

Parameters

Memory-address	Pointer; memory area to put data

M$GET

Get data from memory.

General format

CALL "M$GET" USING memory-address, data-item[, data-size[, data-offset]]

Parameters

Memory-address	Pointer; memory area to put data
Data-item	any data item; data will be retrieved from here for storage
Data-size	numeric; number of bytes to copy
Data-offset	numeric; offset into memory-address to copy

M$PUT

Put data into memory.

General format

CALL "M$PUT" USING memory-address, data-item[, data-size[, data-offset]]

Parameters

Memory-address	Pointer; memory area to put data
Data-item	any data item; data will be retrieved from here for storage
Data-size	numeric; number of bytes to copy
Data-offset	numeric; offset into memory-address to copy

RENAME

Rename a physical file from an old filename to a new filename.

General format

CALL "RENAME" old-filename, new-filename[, reserved-status [,file-type]] GIVING status

Parameters

Old-filename	PIC X(n); The existing filename.
New-filename	PIC X(n); The new filename to which to rename old-filename.
Reserved-status	numeric-identifier; 0 if successful, 1 if unsuccessful; currently not set
File-type	PIC X; "S" if sequential, "R" if relative, "I" if indexed; currently not used
Status	numeric-identifier; 0 if successful, 1 if unsuccessful.

SYSTEM

General format

CALL "SYSTEM" command-line GIVING status

Parameters

Command-line PIC X(n); program and parameters to run.

Status numeric-identifier; exit value of program.

CALL FUNCTIONS (third-party support)

MQ-Series requires the MQ runtime for Java available from IBM.

MQBACK	hconn, compcode, reason
MQBEGIN	hconn, compcode, reason
MQCLOSE	hconn, hobj, options, compcode, reason
MQCMIT	hconn, compcode, reason
MQCONN	name, hconn, compcode, reason
MQCONNX	name, hconn, compcode, reason
MQDISC	hconn, compcode, reason
MQGET	hconn, hobj, msgdesc, getmsgopts, bufferlength, buffer, datalength, compcode, reason
MQINQ	hconn, hobj, selectorcount, selectorstable, intattrcount, intattrstable, charattrlength, charattrs, compcode, reason
MQOPEN	hconn, objdesc, options, hobj, compcode, reason
MQPUT	hconn, hobj, msgdesc, putmsgopts, bufferlength, buffer, compcode, reason
MQPUT1	hconn, objdesc, msgdesc, putmsgopts, bufferlength, buffer, compcode, reason
MQSET	hconn, hobj, selectorcount, selectorstable, intattrcount, intattrstable, charattrlength, charattrs, compcode, reason

CALL FUNCTIONS (native-code)

Some platforms with direct native calling (such as Windows X86 and Linux X86) may call any shared library with simple parameters. API functions available on these systems are not listed here as the support is dynamic based on system.

Appendix H - HP 3000 MPE/iX Intrinsics

A summary of the available functions is included here for reference; for more information, see the HP documentation.

These functions are not available on any platform other than HP 3000 MPE/iX.

For these functions to work correctly, the program must be compiled using the -mpe switch, or by setting the datatype to HP MPE in the IDE. This forces all datatypes to be HP COBOL-II compatible where applicable, allowing the data to be passed correctly.

TurboIMAGE Intrinsics

DBBEGIN	base|baseidlist, text, mode, status, textlen
DBCLOSE	base, dset, mode, status
DBCONTROL	base, qualifier, mode, status
DBDELETE	base, dset, mode, status
DBEND	base|baseidlist|transid, text, mode, status, textlen
DBERROR	status, buffer, length
DBEXPLAIN	status
DBFIND	base, dset, mode, status, item, argument
DBGET	base, dset, mode, status, list, buffer, argument
DBINFO	base, qualifier, mode, status, buffer
DBLOCK	base, qualifier, mode, status
DBMEMO	base, text, mode, status, textlen
DBOPEN	base, password, mode, status
DBPUT	base, dset, mode, status, list, buffer
DBUNLOCK	base, dset, mode, status
DBUPDATE	base, dset, mode, status, list, buffer
DBXBEGIN	base, text, mode, status, textlen
DBXEND	base, text, mode, status, textlen
DBXUNDO	base, text, mode, status, textlen

VPLUS Intrinsics

VCHANGEFIELD	comarea, specbuffer, numentries
VCLOSEBATCH	comarea
VCLOSEFORMF	comarea
VCLOSETERM	comarea
VERRMSG	comarea, buffer, buflen, actuallen
VFIELDEDITS	comarea
VFINISHFORM	comarea
VGETBUFFER	comarea, buffer, buflen

VGETFIELD	comarea, fieldnum, fieldbuf, buflen, actuallen, nextfldnum
VGETFIELDINFO	comarea, infobuf, infobuflen
VGETFILEINFO	comarea, infobuf, infobuflen
VGETFORMINFO	comarea, infobuf, infobuflen
VGETKEYLABELS	comarea, formorglob, numoflabels, labels
VGETLANG	comarea, langnum
VGETNEXTFORM	comarea
VGETINT	comarea, fieldnum, variable
VGETDINT	comarea, fieldnum, variable
VGETREAL	comarea, fieldnum, variable
VGETLONG	comarea, fieldnum, variable
VGETPACKED	comarea, fieldnum, variable, numdigits, decplaces
VGETZONED	comarea, fieldnum, variable, numdigits, decplaces
VGETYYMMDD	comarea, fieldnum, variable
VINITFORM	comarea
VLOADFORMS	comarea, numofforms, formsloaded, forms
VOPENBATCH	comarea, batchfile
VOPENFORMF	comarea, formfile
VOPENTERM	comarea, termfile
VPLACECURSOR	comarea, fieldnum
VPOSTBATCH	comarea
VPRINTFORM	comarea, printcntl, pagecntl
VPRINTSCREEN	comarea, readsize
VPUTBUFFER	comarea, buffer, buflen
VPUTFIELD	comarea, fieldnum, fieldbuf, buflen, actuallen, nextfldnum
VPUTINT	comarea, fieldnum, variable
VPUTDINT	comarea, fieldnum, variable
VPUTREAL	comarea, fieldnum, variable
VPUTLONG	comarea, fieldnum, variable
VPUTPACKED	comarea, fieldnum, variable, numdigits, decplaces
VPUTZONED	comarea, fieldnum, variable, numdigits, decplaces
VPUTYYMMDD	comarea, fieldnum, variable
VPUTWINDOW	comarea, message, length
VREADBATCH	comarea
VREADFIELDS	comarea
VSETERROR	comarea, fieldnum, message, msglen
VSETKEYLABEL	comarea, formorglob, keynum, label
VSETKEYLABELS	comarea, formorglob, numoflabels, labels
VSETLANG	comarea, langnum, errorcode
VSHOWFORM	comarea
VUNLOADFORM	comarea, whichform
VWRITEBATCH	comarea

KSAM Intrinsics

CKCLOSE	filetable, status
CKDELETE	filetable, status
CKERROR	status, result
CKLOCK	filetable, status, lockcond
CKOPEN	filetable, status
CKOPENSHR	filetable, status
CKREAD	filetable, status, record, recordsize
CKREADBYKEY	filetable, status, record, key, keyloc, recordsize
CKREWRITE	filetable, status, record, recordsize
CKSTART	filetable, status, relop, key, keyloc, keylength
CKUNLOCK	filetable, status
CKWRITE	filetable, status, record, recordsize

Appendix I - Date and Time Formats

Elastic COBOL supports FORMAT OF DATE|TIME|TIMESTAMP for direct expression of date and time functionality within the traditional PIC X/9 framework. The FORMAT OF specifies an explicit or implicit format of the date/time.

The implicit specifications are:

Name	Default Format
DATE	@Y-%m-%d
TIME	%H:%M:%S
TIMESTAMP	@Y-%m-%d-%H.%M.%S.@Sm

The format clause uses literal characters and an escape mechanism, exhibited using special punctuation characters, to describe how the date and time are formatted. These explicit date/time formats are useful for directly managing date/time in a variety of circumstances, including extended intrinsic function support and SQL support.

For each escape, the escape character may be included within the literal text by doubling the escape character. The characters '`', '~', '#' and '&' are reserved for future use as escape characters.

Certain escape sequences lead to variable number of bytes, especially those where localization may place a part in determining the final representation.

The values generated from the date/time may be numeric or alphanumeric, dependent upon the escape.

Escape	Bytes	Description
%	-	IBM AS/400 formatting style 1
%%	1	% character
%d	2	Day of Month as Integer [01,31]
%D	8	Same as %m/%d/%y
%H	2	Hour as Integer [00,23]
%I	2	Hour as Integer [01,12]
%j	3	Julian Day of Year as Integer [001,366]
%m	2	Month as Integer [01,12]
%M	2	Minute as Integer [00,59]
%p	2+	Local text equivalent of AM or PM.
%r	8+	Same as %I:%M:%S %p
%R	5	Same as %H:%M
%S	2	Seconds as Integer [00,61]
%y	2	Year without Century as Integer [00,99]
%Y	4	Year with Century as Integer [0000,9999]

Escape	Bytes	Description
@	-	IBM AS/400 formatting style 2
@@	1	@ character
@C	1	Century as integer (19xx is 0)
@p	2+	Text equivalent of AM or PM.
@y	4	Year with Century as Integer [0000,9999]
@Y	4	Year with Century as Integer [0000,9999]
@Sh	2	Hundredths of Second as Integer [00,99]
@Sm	6	Millionths of Second as Integer [000000,999999]
@So	3	Thousandths of Second as Integer [000,999]
@St	1	Tenths of Second as Integer [0,9]

Escape	Bytes	Description
!		Java external unformatted; all numeric formats
!!	1	! character
#e	-	Era
#Y	-	Year with Century
#y	-	Year without Century
#c	-	Century (19 for 19xx)
#C	-	Century as Byte (0 for 19xx)
#N	-	Millenium (1 for 1xxx)
#m	-	Month [0,11]
#u	-	Week of Year
#W	-	Week of Month
#d	-	Date [0,31]
#j	-	Julian Date of Year [0,366]
#w	-	Weekday [0,6]
#U	-	Week in Month
#A	-	AM (0) or PM (1)
#I	-	12-Hour Hour
#H	-	24-Hour Hour
#S	-	Second
#s	-	Milliseconds
#z	-	Timezone
#Z	-	Daylight Savings Timezone
#n	-	Nanoseconds

Escape	Bytes	Description
^		Java external formatted; all numeric formats
^^	1	^ character
&e	1	Era
&Y	4+	Year with Century
&y	2+	Year without Century
&c	2	Century (19 for 19xx)
&C	1	Century as Byte (0 for 19xx)
&N	1	Millenium (1 for 1xxx)
&m	2	Month [0,11]
&u	2	Week of Year
&W	1	Week of Month
&d	2	Date [0,31]
&j	3	Julian Date of Year [0,366]
&w	1	Weekday [0,6]
&U	1	Week in Month
&A	1	AM (0) or PM (1)
&I	2	12-Hour Hour
&H	2	24-Hour Hour
&S	2	Second
&s	2	Milliseconds
&z	8	Timezone
&Z	8	Daylight Savings Timezone
&n	9	Nanoseconds

Escape	Bytes	Description
$		Name format long; name formats where applicable
$$	1	$ character
$e	3	Era (BCE or CE)
$Y	4	Year (numeric)
$c	2	Century (numeric)
$C	1	Century Byte (numeric)
$N	1	Millenium (numeric)
$m	-	Month (January - December)
$u	2	Week of Year (numeric)
$W	1	Week of Month (numeric)
$d	2	Date of Month (numeric)
$j	3	Julian Date of Year (numeric)
$w	9	Weekday (Sunday - Saturday)
$U	1	Week in Month (numeric)
$A	2	AM or PM
$I	2	12-Hour Hour (numeric)
$H	2	24-Hour Hour (numeric)
$M	2	Minute (numeric)
$S	2	Second (numeric)
$s	3	Millisecond (numeric)
$z	5	Time Zone (+|- HH:MM)
$Z	5	Daylight Saving Time Zone (+|- HH:MM)
$n	9	Nanoseconds (numeric)

Escape	Bytes	Description
*		Name format short; name formats where applicable
**	1	* character
*e	2	Era (BC or AD)
*Y	4	Year (numeric)
*c	2	Century (numeric)
*C	1	Century Byte (numeric)
*N	1	Millenium (numeric)
*m	3	Month (Jan-Dec)
*u	2	Week of Year (numeric)
*W	1	Week of Month (numeric)
*d	2	Date of Month (numeric)
*j	3	Julian Date of Year (numeric)
*w	3	Weekday (Sun- Sat)
*U	1	Week in Month (numeric)
*A	2	AM or PM
*I	2	12-Hour Hour (numeric)
*H	2	24-Hour Hour (numeric)
*M	2	Minute (numeric)
*S	2	Second (numeric)
*s	3	Millisecond (numeric)
*z	5	Time Zone (+|- HH:MM)
*Z	5	Daylight Saving Time Zone (+|- HH:MM)
*n	9	Nanoseconds (numeric)

Appendix J - Standard Color Names

A Color Name may be specified in one of three ways. It may be given by a symbolic name from the chart below, expressed as decimal RGB, or as hex RGB.

Decimal RGB is expressed as R,G,B where R is red 0..255, G is green 0..255, B is blue 0..255.

Hex RGB is expressed as #RRGGBB, where R is red 00-FF, G is green 00-FF, B is blue 00-FF.

Color names may also be set by using a Unix style rgb.txt file. Elastic COBOL will attempt to load 'rgb.txt' from the current directory, or from the filename given by the program property named 'rgb.txt'. The Unix style rgb.txt file consists of a line sequential file with lines of the form 'R G B name', where R, G, and B are red, green, blue in decimal from 0..255.

These color names are referred to as the standard color names and are usable where an unlimited number of colors are allowed. Certain items such as the screen section may support only a limited color selection, and such colors will not be from this list.

Color names predefined by Elastic COBOL include:

DARKGRAY	DARK GRAY
GRAY	LIGHT GRAY
LIGHTGRAY	BLUE
ORANGE	GREEN
PINK	CYAN
BLACK	RED
DARK BLUE	MAGENTA
DARK GREEN	YELLOW
DARK CYAN	WHITE
DARK RED	GRAY
DARK MAGENTA	MAROON
BROWN	PURPLE
OLIVE	NAVY
TEAL	SILVER
FUCHSIA	LIME
AQUA	GREY
DARK-GREY	DARKGREY
LIGHTGREY	LIGHT-GREY
ACTIVE-CAPTION	ACTIVE-CAPTION-BORDER
ACTIVE-CAPTION-TEXT	CONTROL
CONTROL-DARK-SHADOW	CONTROL-HIGHLIGHT
CONTROL-LIGHT-HIGHLIGHT	CONTROL-SHADOW
CONTROL-TEXT	DESKTOP
INACTIVE-CAPTION	INACTIVE-CAPTION-BORDER
INACTIVE-CAPTION-TEXT	INFO
INFO-TEXT	MENU
MENU-TEXT	SCROLLBAR

TEXT-HIGHLIGHT	TEXT-HIGHLIGHT-TEXT
TEXT-INACTIVE-TEXT	TEXT-TEXT
WINDOW	WINDOW-BORDER
WINDOW-TEXT	

Index