INTERNATIONAL STANDARD

ISO/IEC 15852

First edition
1999-03-01

Information technology — Programming languages — M Windowing API

Technologies de l'information — Langages de programmation — Interface de programme d'application (API) du langage M, utilisant des fenêtres

Reference number
ISO/IEC 15852:1999(E)

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

Contents

		Page
1	General	1
1.1	Scope	1
1.2	Relation to Environment	1
1.3	Windows, Elements and Choices	1
1.4	Displays	2
1.5	Events	2
1.6	Attributes	2
1.7	Drawing	2
1.8	Parents and Children	2
2	Normative Reference	3
3	Attributes	4
3.1	Manipulating Attributes	4
3.2	Attribute Definitions	5
3.3	Attribute Value Domains	6
3.3.1	Length of Character Strings	6
3.3.2	Colors	6
3.3.3	Characters and the Empty String	6
3.3.4	True/False Values	7
3.3.5	Implementation and Application Attributes	7
3.3.6	External Resources	7
3.3.7	Fonts	8
3.3.8	Identifiers	9
3.3.9	Position, Size and Units	10
3.3.10	Display Characteristics	11
3.3.11	Drawing Characteristics	12
3.3.12	Orientation	13
3.3.13	Key and Pointer Button Codes	13
3.3.14	Insertion Point and Selection Range	13
3.3.15	Title Position	13
3.3.16	Choice Layout	14
3.3.17	Modality	14
3.3.18	Scrolling	14
3.3.19	Focus Location	15
4	Displays	15
4.1	Display Attributes	15
4.1.1	BCOLOR	16
4.1.2	CLIPBOARD	16

© ISO/IEC 1999

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

4.1.3 COLOR 16
4.1.4 COLORTYPE 16
4.1.5 FCOLOR 16
4.1.6 FOCUS 17
4.1.7 KEYBOARD 17
4.1.8 PEN 17
4.1.9 PLATFORM 17
4.1.10 PTR 18
4.1.11 SIZE 18
4.1.12 SPECTRUM 18
4.1.13 TYPEFACE 18
4.1.14 UNITS 19

5 Windows 19
5.1 Window Types 19
5.1.1 MTERM Windows 20
5.2 Window Components 20
5.3 Window Repainting 20
5.4 Window Attributes 20
5.4.1 Applicability of Window Attributes 21
5.4.2 ACTIVE 23
5.4.3 BCOLOR 23
5.4.4 COLOR 23
5.4.5 DEFBUTTON 23
5.4.6 DISPLAY 24
5.4.7 EVENT 24
5.4.8 FCOLOR 24
5.4.9 FFACE 24
5.4.10 FSIZE 24
5.4.11 FSTYLE 25
5.4.12 ICON 25
5.4.13 ICONIFY 25
5.4.14 ID 25
5.4.15 ITITLE 25
5.4.16 MENUBAR 26
5.4.17 MIN 26
5.4.18 MODAL 26
5.4.19 NEXTG 26
5.4.20 PARENT 27
5.4.21 POS 27
5.4.22 RESIZE 28
5.4.23 SCROLL 28
5.4.24 SIZE 28
5.4.25 SIZEMIN 29
5.4.26 SIZEWIN 29
5.4.27 TIED 29
5.4.28 TITLE 30
5.4.29 TYPE 30
5.4.30 UNITS 30
5.4.31 VISIBLE 30

6 Elements 31
6.1 Gadgets 31
6.1.1 Check Box 31
6.1.2 Document 32
6.1.3 Generic Box 32
6.1.4 Group Frame 32
6.1.5 Label 32

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

6.1.6 List Box 32
6.1.7 List Button 32
6.1.8 List Entry Box 32
6.1.9 Long List Box 33
6.1.10 Push Button 33
6.1.11 Radio Button Set 33
6.1.12 Scroll 33
6.1.13 Symbol 33
6.1.14 Text 33
6.2 Menus 33
6.3 Timers 34
6.4 Element Attributes 34
6.4.1 ACTIVE 38
6.4.2 BCOLOR 38
6.4.3 CANCEL 38
6.4.4 CANCHANGE 38
6.4.5 CHANGED 38
6.4.6 CHARMAX 39
6.4.7 CHOICE 39
6.4.8 DRAW 39
6.4.9 DRAWTYPE 40
6.4.10 EVENT 40
6.4.11 FCOLOR 40
6.4.12 FFACE 40
6.4.13 FRAMED 40
6.4.14 FSIZE 41
6.4.15 FSTYLE 41
6.4.16 ID 41
6.4.17 INSELECT 41
6.4.18 INTERVAL 42
6.4.19 NEXTG 42
6.4.20 POS 43
6.4.21 RESOURCE 43
6.4.22 ROWCOL 44
6.4.23 SCROLL 44
6.4.24 SCROLLBY 45
6.4.25 SCROLLDIR 45
6.4.26 SCROLLPOS 45
6.4.27 SCROLLRANGE 46
6.4.28 SELECTMAX 46
6.4.29 SELECTVAL 46
6.4.30 SIZE 47
6.4.31 TBCOLOR 48
6.4.32 TFCOLOR 48
6.4.33 TFFACE 48
6.4.34 TFSIZE 48
6.4.35 TFSTYLE 49
6.4.36 TITLE 49
6.4.37 TOPSHOW 49
6.4.38 TPOS 50
6.4.39 TYPE 50
6.4.40 UNITS 50
6.4.41 VALUE 50
6.4.42 VISIBLE 52
6.5 Choice Attributes 52
6.5.1 Applicability of Choice Attributes 53
6.5.2 ACCELERATOR 53
6.5.3 ACTIVE 53

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.5.4 AID 54
6.5.5 EVENT 54
6.5.6 MARKER 54
6.5.7 SEPARATOR 54
6.5.8 SUBMENU 55

7 Drawing 55
7.1 Draw Command Numbers 55
7.2 Draw Commands 56
7.2.1 ARC 56
7.2.2 BCOLOR 57
7.2.3 BITMAP 57
7.2.4 BOX 57
7.2.5 DMODE 57
7.2.6 DRAWTEXT 58
7.2.7 FCOLOR 58
7.2.8 FFACE 58
7.2.9 FILLPAT 58
7.2.10 FSIZE 59
7.2.11 FSTYLE 59
7.2.12 LINE 59
7.2.13 OVAL 59
7.2.14 PENSIZE 59
7.2.15 PICTURE 60
7.2.16 PIE 60
7.2.17 POINT 60

8 Events 60
8.1 Event Types 60
8.1.1 Window State Events 61
8.1.2 Pointer Events 61
8.1.3 Keyboard Events 62
8.1.4 Focus Events 62
8.1.5 Select Events 63
8.1.6 Long List Box Events 64
8.1.7 Help Events 64
8.1.8 Timer Events 64
8.1.9 Applicability of Event Types 64
8.2 Event Specification Attributes 68
8.2.1 ENABLE 68
8.2.2 FILTERIN 68
8.3 Processing Events 69
8.4 Event Commands 69
8.4.1 ESTART 69
8.4.2 ESTOP 70
8.4.3 ETRIGGER 70
8.5 Event Information 71
8.5.1 CHOICE 71
8.5.2 CLASS 72
8.5.3 ELEMENT 72
8.5.4 KEY 72
8.5.5 NEXTFOCUS 72
8.5.6 PRIORFOCUS 73
8.5.7 OK 73
8.5.8 PBUTTON 73
8.5.9 PPOS 73
8.5.10 PSTATE 74
8.5.11 SEQUENCE 74

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

8.5.12 TYPE 74
8.5.13 WINDOW 74
8.5.14 Applicability of Event Information Attributes 75

9 Additional Functions and Special Variables 78
9.1 Functions 78
9.1.1 $WFONT 78
9.1.2 $WTFIT 78
9.1.3 $WTWIDTH 79
9.2 Special Variables 79

10 Error Handling 79
10.1 Error Information 79
10.2 Errors 80

11 Conformance 81
11.1 Implementations 81
11.1.1 Conforming Implementation 81
11.1.2 MDC Conforming Implementation 83
11.1.3 MDC Strictly Conforming Implementation 83
11.1.4 MDC Partial Implementation 83
11.1.5 Multiple Levels of Conformance 83

Index 84

Tables

Window Attributes 22
Element Attributes 36
Choice Attributes 53
Events 65
Event Information 76
Implementation Definition 81
Implementation-Specific Features 81
Portability Limits 82
Unimplemented Features 82

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 15852 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 22, *Programming languages, their environments and system software interfaces*.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Information technology — Programming languages — M Windowing API

1 General

1.1 Scope

The M Windowing API (MWAPI) extends the M programming technology with the addition of capabilities for developing and operating graphical user interface (GUI) applications.

For the purposes of this International Standard, an application is defined as a collection of one or more M routines using MWAPI capabilities and a user is a person utilizing such an application.

1.2 Relation to Environment

The MWAPI may, but is not required to, operate in conjunction with an underlying windowing platform. The MWAPI does not provide a detailed specification of look and feel for applications. Rather, if an underlying windowing platform is present, it is intended that the MWAPI adhere to the platform's look and feel to the extent practicable. If no underlying windowing platform is present, the MWAPI implementation determines look and feel characteristics and carries out actions that would otherwise be carried out by the underlying windowing platform.

1.3 Windows, Elements and Choices

The primary mechanism for communicating with a user is a window. Windows provide a general mechanism for displaying text and graphics, typically enclosed by a border.

Associated with windows, elements perform specialized tasks. Elements include gadgets, menus, and timers.

Gadgets are used for specialized input and output. For instance, there are gadgets designed for text input and gadgets that enable users to select from a list of items.

Menus are used to enable users to select actions to be performed.

Timers provide a means for notification of an application when a specified period of time has elapsed.

Menus and certain gadgets contain a list of choices. For instance, a list box has a list of items that a user can select.

An entity is a window, element, or choice.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

1.4 Displays

Windows are shown to the user via a display, a logical surface on which windows are placed. Each M process has its own display(s); displays are not shared, although physical devices may be shared.

There is not necessarily a one-to-one correspondence between a display and the hardware device on which it appears. Multiple displays may appear on the same device and may overlap. A single display may span multiple devices.

Each display has associated input devices, typically a keyboard with alphanumeric, function and other keys, and a pointer such as a mouse.

1.5 Events

Applications are made aware of actions taken by users, such as pressing a key or selecting a menu item, or of other occurrences via events. Applications specify the events about which they wish to be notified and the code to execute when an event occurs.

1.6 Attributes

The characteristics of windows, elements, and choices are specified through attributes in the structured system variable ^$WINDOW. The characteristics of displays are specified through attributes in the structured system variable ^$DISPLAY. The characteristics of events that occur are specified through attributes in the structured system variable ^$EVENT.

1.7 Drawing

Within one type of gadget, the generic box, the MWAPI provides facilities for drawing geometric figures, text, and other graphical objects.

1.8 Parents and Children

Each window and element has a single parent. A window's parent is either the display on which it appears or another window. An element's parent is a window. If P is the parent of C, then C is a child of P.

A is an ancestor of C and C is a descendant of A if

(i) A is the parent of C, or

(ii) there exist one or more windows (B_1 ... B_n) such that A is the parent of B_1, B_n is the parent of C, and, for x equal to B_1 through B_{n-1}, B_x is the parent of B_{x+1}.

A window or element inherits default values for certain attributes from its parent. A window or element cannot be created before its parent is created. A window or element can be created by the same M command that creates its parent.

P is a parent menu of S and S is a submenu of P if S is displayed when a choice in P is activated.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

2 Normative Reference

The following standard contains provisions which, through reference in this text, constitute provisions of this International Standard. At the time of publication, the edition indicated was valid. All standards are subject to revision, and parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent edition of the standard indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

> ANSI/MDC X11.1-1995, *American National Standard for Information Systems — Programming Languages — M.*

The following terms used in this text have the meaning defined in ANSI/MDC X11.1-1995:

actuallist
command
digit
doargument
eol
expr
expratom
function
glvn
gvn
ident
intexpr
intlit
killargument
L
lvn
mnemonicspace
name
namevalue
numexpr
numlit
openargument
postcond
process
routine
setdestination
SP
ssvn
structured system variable
subscript
svn
timeout
tvexpr
useargument
V
xargument

In addition, the syntax metalanguage used in this International Standard is the same as that used in ANSI/MDC X11.1.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

3 Attributes

Attributes are specified as nodes within the structured system variables (ssvns) ^$WINDOW, ^$DISPLAY and ^$EVENT. Attributes are identified via keywords that appear as subscripts within these ssvns. These keywords are defined as dattribute, wattribute, elattribute, cattribute, especattribute, and einfoattribute values, and are referred to collectively as attribute names. All attribute names not specified by this International Standard are reserved.

From the perspective of any M process, ^$DISPLAY, ^$WINDOW and ^$EVENT contain information about only those displays, windows and events that are accessible to the process.

3.1 Manipulating Attributes

When an M process begins execution, ^$WINDOW and ^$EVENT are not defined.

Values may be assigned to attributes by referencing ^$WINDOW, ^$DISPLAY or ^$EVENT as a glvn on the left hand side of the equal sign in an M MERGE command or as a setdestination in an M SET command.

When attribute values are assigned with an M MERGE command, all visible effects occur at the successful conclusion of the command. Note that this does not require that the MERGE command modify an ssvn in an atomic fashion. Rather, it places the visible effects at the conclusion of the command.

If an application attempts to assign a value to a reserved attribute name, an error condition occurs with $ECODE containing ",M?1," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

An entity is created at the conclusion of the first M command that assigns a value to one or more of its attributes. If a value is not explicitly assigned to an attribute when an entity is created, a default value may be provided by the MWAPI or the underlying windowing platform. If the default value is dependent upon the values of one or more attributes of another existing entity, the attribute values of that other entity at the time of the new entity's creation are used. Subsequent changes to the existing entity's attribute values do not affect previously established default values.

> Note: If there are multiple attributes for which values must be assigned when an entity is created, the entity can only be created using the M MERGE command. All gadgets currently defined by this International Standard have this characteristic.

Certain attributes of an existing display, window, element, choice or event information structure can be modified by assigning new values to them. Attribute values are also modified by the MWAPI in response to user actions or other occurrences.

If a value domain is specified for an attribute, all values outside the specified domain are reserved. Unless specifically stated otherwise, if an application attempts to assign to an attribute a value that is outside the specified domain of that attribute, an error condition occurs with $ECODE containing ",M?2," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

The value of an attribute, whether explicitly assigned by the application, or present by default, or set by the MWAPI, can be obtained by referencing ^$WINDOW, ^$DISPLAY or ^$EVENT.

In any context in which an attribute is modifiable, using the M KILL command to remove the value of an attribute causes the attribute to have the default value that it would have if the entity were created at that instant, if a default is specified, or to become undefined, if no default is specified.

In any context in which an attribute is not modifiable, an attempt by an application to assign a value to the attribute or to remove the value of the attribute is ignored.

--`,,```,,,,````-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

An MTERM window that was created by the execution of an OPEN command is destroyed by using its wname as the device designation in an M CLOSE command. Any other window, an element, or a choice is destroyed by removing its wname, ename, or item, respectively, from ^$WINDOW with the M KILL command. Destroying an entity causes it to become undefined and causes all of the attributes of the entity to be removed from ^$WINDOW. When a window is destroyed, all descendant windows and elements are destroyed. When an element is destroyed, all descendant choices are destroyed

Execution of an M KILL command which has a killargument equal to "^$WINDOW" or "^$W" causes all windows defined in ^$WINDOW to be destroyed and causes the ^$WINDOW ssvn to become undefined. When a process executes an M HALT command, all of the windows that belong exclusively to it are destroyed and ^$EVENT becomes undefined.

3.2 Attribute Definitions

Within this document, each attribute is defined with a table of the following format:

Definition	Specifies the meaning of the attribute.
Domain	Specifies the range of possible values for the attribute.
Access	Specifies the ways in which the attribute can be accessed by an application:
	Create — Indicates that the application can assign a value for the attribute when the entity is created.
	Modify — For a wattribute, elattribute, cattribute, or especattribute, Modify indicates that the application can assign a value to the attribute after the entity is created. For a dattribute or einfoattribute, Modify indicates that the application can assign a value to the attribute.
	Reference — Indicates that the application can obtain the current value of the attribute.
Default	For a wattribute, elattribute, cattribute, or especattribute, Default specifies the value of the attribute if no value is assigned when the entity is created. For a dattribute or einfoattribute, Default specifies the value of the attribute prior to any assignment of a value to it by the application. In addition to specific default values, the following general default value types are defined.
	None — Indicates that no default value is defined and the attribute node is undefined.
	Platform — Indicates that the default is established by the underlying windowing platform, if there is one, and by the implementation of the MWAPI otherwise.
	Not Applicable — Indicates that no default applies because the attribute can only be referenced.

--`,,```,,,,````-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

3.3 Attribute Value Domains

3.3.1 Length of Character Strings

When the MWAPI is present, the portability requirement in Section 2 of the MUMPS International Standard is modified such that any result, whether intermediate or final, of expression evaluation whose length does not exceed the portability limit for longchars, and which would otherwise be permitted, is permitted. This does not permit assignment of a value to an ssvn node whose domain is not longchars, to an lvn, or to a gvn where such assignment would not be permitted by the MUMPS International Standard. This also does not permit use of a value as a subscript, as an actuallist parameter, or as an xargument where such use would not be permitted by the MUMPS International Standard.

3.3.2 Colors

Color specifications have three components: a red value, a green value, and a blue value.

color ::= expr V rcolor , gcolor , bcolor

rcolor ::= intlit

gcolor ::= intlit

bcolor ::= intlit

rcolor specifies the amount of red, gcolor specifies the amount of green, and bcolor specifies the amount of blue. rcolor, gcolor, and bcolor values are restricted to a range of 0, meaning no color, through 65535, meaning full color.

A color value of "0,0,0" signifies black; a color value of "65535,65535,65535" indicates white; and a color value with identical rcolor, gcolor, and bcolor values signifies a shade of grey.

Appropriate scaling from the 0 to 65535 range to the color domain of the underlying windowing platform or hardware is performed automatically by the MWAPI.

3.3.3 Characters and the Empty String

chars denotes a sequence of zero or more characters.

noncommachars denotes a sequence of zero or more characters, excluding the comma character.

longchars denotes a sequence of zero or more characters.

emptystring denotes an M empty string.

tab denotes a tab character.

linend denotes an M eol.

mnemonichar denotes an ampersand character.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

3.3.4 True/False Values

truefalse ::= expr

Any truefalse value assigned by an application to an attribute is interpreted as TRUE or FALSE in accordance with the M definition of tvexpr. If the interpretation is TRUE, the attribute's value is set to 1 (one). If the interpretation is FALSE, the attribute's value is set to 0 (zero).

Any truefalse value assigned by the MWAPI has a value of 1 (one) for TRUE and 0 (zero) for FALSE.

3.3.5 Implementation and Application Attributes

implementationattributes, implementationvalues, and implementationdrawcommands are reserved for implementors.

implementationattribute ::= Z unspecified

implementationvalue ::= Z unspecified

implementationdrawcommand ::= Z unspecified

applicationattributes are reserved for application use.

applicationattribute ::= Y chars

An applicationattribute must be a valid M subscript.

The meanings of implementationattributes, implementationvalues, applicationattributes, and implementationdrawcommands are not specified.

3.3.6 External Resources

External resources provided by the implementor, by the underlying windowing platform, or by another source, are identified by extresource.

extresource ::= expr V	F,fileid R,resourceid implementationvalue

fileid and resourceid are platform-dependent file and resource identifiers, respectively. The formats of fileids and resourceids are unspecified.

In addition to external sources for symbols, a number of symbol types are specified by the MWAPI.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
                 | extresource                          |
                 |                                      |
symbol ::=       | expr V   | M,ERROR                   |
                 |          | M,INFO                    |
                 |          | M,QUEST                   |
                 |          | M,WARN                    |
```

symbol values of "M,ERROR", "M,INFO", "M,QUEST", and "M,WARN" specify symbols that identify error, information, question, and warning situations, respectively. If the underlying windowing platform provides symbols for these situations, they are used. Otherwise, symbols are provided by the MWAPI. extresource enables use of other symbols not specified by the MWAPI.

```
                 | extresource                          |
                 |                                      |
marker ::=       | expr V   | M,BULLET                  |
                 |          | M,CHECK                   |
                 |          | M,DIAMOND                 |
```

marker values of "M,BULLET", "M,CHECK", and "M,DIAMOND" specify menu choice markers shaped like bullets, check marks, and diamonds, respectively. If the underlying windowing platform provides markers for these situations, they are used. Otherwise, markers are provided by the MWAPI. extresource enables use of other markers not specified by the MWAPI.

3.3.7 Fonts

Type fonts specify the appearance characteristics of text. A font specification consists of three components: a face, a size, and a style.

```
fontface ::= expr V      | M,DEFAULT      |
                         | M,FIXED        |
                         | M,SANS         |
                         | M,SERIF        |
                         | alpha chars    |
```

The following fontface values are defined:

M,DEFAULT Default type face of the underlying windowing platform.

M,FIXED A non-proportional type face.

M,SANS A proportional type face from a sans-serif family.

M,SERIF A proportional type face from a serif family.

fontface values not beginning with "M," may be used by implementors to identify additional type faces.

fontsize ::= expr V numlit

Positive numeric values for fontsize specify the size of displayed text measured in points. A point is 1/72 inches.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

fontstyle ::= expr V | NORMAL
fstyle [,fstyle] ...

fstyle ::= | BOLD
ITALIC
ULINE
implementationvalue

If fontstyle includes the fstyle BOLD, the text is displayed in bold face. If fontstyle includes the fstyle ITALIC, the text is displayed italicized. If fontstyle includes the fstyle ULINE, the text is displayed underlined.

If a font requested by an application cannot be provided, an alternative font with different fontface, fontsize, and/or fontstyle values may be substituted by the underlying windowing platform or the MWAPI implementation. Any references to attributes identifying fonts (including attributes to which values beginning with "M," have been assigned) yield information about the actual font.

3.3.8 Identifiers

A widen is a sequence of letters and/or numbers optionally preceded by a percent sign.

widen ::= | % alphanum | [alphanum] ...

alphanum ::= | ident
digit

Windows are identified by wnames.

wname ::= widen

Elements are identified by enames.

ename ::= | gname
mname
tname

Gadgets are identified by gnames.

gname ::= widen

Menus are identified by mnames.

mname ::= widen

Timers are identified by tnames.

tname ::= widen

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Choices are identified by items.

item ::= chars

For any attribute which has as its domain wname or ename, if an application attempts to assign (via a command) a value for the attribute which does not identify a window or element which exists at the conclusion of that command, an error condition occurs with $ECODE containing ",M?3," and $EREF containing a value that indicates the ssvn reference to which the application attempted to assign the invalid wname or ename. Subsequent destruction of the window or element does not cause an error condition to occur.

3.3.9 Position, Size and Units

A UNITS dattribute, wattribute, or elattribute specifies the unit of measure for the size or position of a display, window, or element, respectively.

unitspec ::=
- CHAR [, chars]
- PIXEL
- POINT
- REL [, hscale , vscale]
- implementationvalue

A unitspec value of CHAR specifies that position or size is measured relative to the size of characters in a basis font which is determined by context. A unit in the vertical dimension is equal to the line spacing (including external leading) in that font. If chars is specified and is not equal to emptystring, a unit in the horizontal dimension is equal to the average character width in that font for the characters in chars. If chars is not specified or if chars is equal to emptystring, a unit in the horizontal dimension is equal to the maximum character width in that font.

A unitspec value of PIXEL specifies that position or size is measured in pixels. A unit is equal to the size of a pixel.

A unitspec value of POINT specifies that position or size is measured in points. A unit is equal to the size of a point, 1/72 inches.

A unitspec value of REL specifies a relative scale. If hscale and vscale are not given, they default to 100. REL units are defined relative to a basis size of another object. If the other object is a display, the basis size is the size of the display. If the other object is a window, the basis size is the size of the window viewport when that other window is created.

If hbasis and vbasis are the basis size in PIXEL units, and hrel and vrel are a size or position specification in REL units, hrel and vrel are converted to PIXEL units (hpix and vpix) by the following calculations:

hpix = hrel / hscale * hbasis

vpix = vrel / vscale * vbasis

hscale ::= numlit

vscale ::= numlit

A position consists of a horizontal location (hpos), a vertical location (vpos), and an optional units of measure designation (unitspec).

pos ::= expr V hpos , vpos [, unitspec]

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

hpos ::= numlit

vpos ::= numlit

The position of an entity is always specified as the location of the origin of that entity with respect to the origin of a display, window or gadget. The origin of a display or a window frame or a window viewport or a gadget is its upper left hand corner. When a window is created, the origin of its application area is the upper left hand corner of the window viewport. Subsequent scrolling moves the application area origin. The origin is position (0,0) and positions are measured in a positive direction to the right and down.

Note: In the future, support may be added for different coordinate systems to facilitate support of languages with orientations other than left-to-right and top-to-bottom.

A size consists of a horizontal size (hsize), a vertical size (vsize), and an optional units of measure designation (unitspec).

size ::= expr V	hsize [, [vsize] [, unitspec]] , vsize [, unitspec]

hsize ::= numlit

vsize ::= numlit

Unless indicated otherwise, if an application attempts to assign to an attribute an hsize or vsize value which is less than or equal to zero, an error occurs with $ECODE containing ",M?2," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

If the unitspec is omitted when a pos or size value is assigned to an attribute, unitspec defaults to the then current value of the entity's UNITS attribute.

Any reference to ^$DISPLAY or ^$WINDOW for a pos or size value yields a value that includes hpos, vpos, and unitspec pieces and that is expressed in the unit of measure specified (when the reference is made) by the entity's UNITS attribute.

If a size or position value assigned by an application cannot be provided, a size or position is determined by the underlying windowing platform or the MWAPI implementation. Any reference to attributes identifying a size or position reflects the actual size or position.

3.3.10 Display Characteristics

displaycolor specifies the capability of the display to show colors or shades of grey.

displaycolor ::= expr V	COLOR GREY MONO implementationvalue

A displaycolor value equal to "COLOR" indicates a display that can show colors. A displaycolor value equal to "GREY" indicates a display that can show multiple shades of grey. A displaycolor value equal to "MONO" indicates a display that can show only a single background and foreground color.

platformversion identifies the name and version designator of the underlying windowing platform.

platformversion ::= expr V [platformname , versionid]

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

platformname ::=
| MAC
| MSWIN
| PM
| X/MTF
| X/OPNLK
| implementationvalue |

A platformname value of "MAC" indicates an Apple Macintosh system. A platformname value of "MSWIN" indicates a Microsoft Windows system. A platformname value of "PM" indicates a Presentation Manager system. A platformname value of "X/MTF" indicates an XWindow/Motif system. A platformname value of "X/OPNLK" indicates an XWindow/Open Look system.

versionid ::= chars

3.3.11 Drawing Characteristics

drawtype specifies the type of drawcommands that can be specified.

drawtype ::= expr V
| MDRAW
| implementationvalue |

fillmode specifies the relationship between components of a drawing and the draw background color.

fillmode ::= expr V
| OPAQUE
| TRANSPARENT
| implementationvalue |

fillpattern specifies the pattern used to fill an area.

fillpattern ::= expr V
| BDIAGONAL
| CROSSHATCH
| DCROSSHATCH
| FDIAGONAL
| HORIZONTAL
| NONE
| SOLID
| VERTICAL
| implementationvalue |

dmode specifies the effect of drawing by certain drawcommands on existing pixels in the drawing area.

dmode ::= expr V
| SOLID
| XOR
| implementationvalue |

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

3.3.12 Orientation

An orientation value equal to "H" indicates that an entity has horizontal orientation. An orientation value equal to "V" indicates that an entity has vertical orientation.

orientation ::= expr V | H
V
implementationvalue |

3.3.13 Key and Pointer Button Codes

Key sequences, consisting of one or more keys, are identified by key codes (keycodes).

keycode ::= noncommachars

Pointer buttons are identified by pointer button codes (pbuttoncodes).

pbuttoncode ::= noncommachars

The correspondence between actual keyboard keys and key codes and between actual pointer buttons and pointer button codes is established by a mapping mechanism. For each key code, the mapping mechanism specifies the corresponding key sequence and whether the key code is a data key code or a function key code. For each pointer button code, the mapping mechanism specifies the corresponding pointer button.

While this International Standard does not specify how this mapping is carried out, a mapping mechanism must be available.

3.3.14 Insertion Point and Selection Range

insel ::= expr V insert , seloff

insert ::= intlit

seloff ::= [-] intlit

insert indicates a position within a text value. A value equal to 0 (zero) indicates a position before the first character. A value of N, where N>0, indicates a position after the Nth character and, if the $N+1^{th}$ character is present, before the $N+1^{th}$ character.

seloff indicates an offset relative to insert.

3.3.15 Title Position

titlepos specifies where a gadget title appears.

titlepos ::= expr V | LEFT
RIGHT
TOP
implementationattribute |

If titlepos is "LEFT", the upper right hand corner of the title is immediately adjacent to the upper left hand corner of the gadget. If titlepos is "RIGHT", the upper left hand corner of the title is immediately adjacent to the upper right hand corner of the gadget. If titlepos is "TOP", the lower left hand corner of the title is immediately adjacent to the upper left hand corner of the gadget.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

3.3.16 Choice Layout

rowcolspec specifies how choices are positioned within a gadget.

rowcolspec ::= expr V rowcol [, rowcolsize [, filldir]]

rowcol ::= | COL
ROW |

rowcolsize ::= intlit

filldir ::= | H
V |

The meaning of rowcolspec is specified in Section 0

3.3.17 Modality

modaltype specifies the extent to which a window with the MODAL wattribute defined disables other windows.

modaltype ::= expr V | ANCESTORS
APPLICATION
PARENT
implementationvalue |

The meaning of modaltype is specified in Section 0.

3.3.18 Scrolling

scrollrange specifies the value range that a scroll or long list box gadget displays.

scrollrange ::= expr V | scrollmin
[scrollmin] , scrollmax |

scrollmin ::= numlit

scrollmax ::= numlit

scrollby specifies the amount by which the VALUE elattribute increases or decreases when the user presses a scroll gadget's movement controls.

scrollby ::= expr V | scrollsmall
[scrollsmall] , scrollbig |

scrollsmall ::= numlit

scrollbig ::= numlit

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

3.3.19 Focus Location

focusloc identifies a window or gadget which has, had, or is about to get focus.

focusloc ::= expr V | wname
wname , gname
emptystring

If focusloc identifies a window, it has the form:

wname

If focusloc identifies a gadget, it has the form:

wname , gname

If focusloc does not identify a window or gadget defined in ^$WINDOW, it has the form:

emptystring

4 Displays

The ssvn ^$DISPLAY contains information about the displays accessible to a process, in the form

^$DI[SPLAY] (display , dattribute)

display is the identifier for a display.

display ::= expr

The format of display values and the means for associating displays with actual hardware devices are not specified except that the display value must be a valid subscript.

4.1 Display Attributes

Display attributes, identified by dattributes, define the characteristics of displays.

dattribute ::= expr V | BCOLOR
CLIPBOARD
COLOR
COLORTYPE
FCOLOR
FOCUS
KEYBOARD
PEN
PLATFORM
PTR
SIZE
SPECTRUM
TYPEFACE
UNITS
applicationattribute
implementationattribute

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

4.1.1 BCOLOR dattribute

Definition BCOLOR indicates the default value for the BCOLOR wattribute for windows that are children of the display.

Domain color

Access Modify, Reference

Default Platform

4.1.2 CLIPBOARD dattribute

Definition CLIPBOARD specifies the data associated with the clipboard provided by the underlying windowing platform, if there is one, and by the MWAPI implementation, otherwise.

Domain longchars

Access Modify, Reference

Default Not Applicable

4.1.3 COLOR dattribute

Definition COLOR indicates the default value for the COLOR wattribute for windows that are children of the display.

Domain color

Access Modify, Reference

Default Platform

4.1.4 COLORTYPE dattribute

Definition COLORTYPE indicates the color capabilities of the display.

Domain displaycolor

Access Reference

Default Not Applicable

4.1.5 FCOLOR dattribute

Definition FCOLOR indicates the default value for the FCOLOR wattribute for windows that are children of the display.

Domain color

Access Modify, Reference

Default Platform

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

4.1.6 FOCUS dattribute

Definition FOCUS identifies the window and gadget, if any, which has focus; that is, to which keyboard and pointer input associated with this display are sent. If no window created by the application has focus, or if the window or gadget which has focus is destroyed, the value is platform dependent.

If an application assigns a value to FOCUS which identifies a window or gadget defined in ^$WINDOW, that window or gadget receives focus but a focus event does not occur.

If an application attempts to set focus to a window or gadget that is not currently visible and active, or to set focus to a gadget that cannot get focus in the underlying windowing platform, an error condition occurs with $ECODE containing ",M?4," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

Domain focusloc

Access Modify, Reference

Default Platform

4.1.7 KEYBOARD dattribute

Definition If KEYBOARD is defined, the display has a keyboard. If KEYBOARD is not defined, the display does not have a keyboard.

Domain emptystring

Access Reference

Default Not applicable

4.1.8 PEN dattribute

Definition If PEN is defined, the display has a pen. If PEN is not defined, the display does not have a pen.

Domain emptystring

Access Reference

Default Not applicable

4.1.9 PLATFORM dattribute

Definition PLATFORM identifies the name and version identifier of the underlying windowing platform.

The versionid indicates the version of the underlying windowing platform.

If there is no underlying windowing platform, platformversion is equal to implementationvalue.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Domain platformversion

Access Reference

Default Not applicable

4.1.10 PTR dattribute

Definition If PTR is defined, the display has a pointer, such as a mouse. If PTR is not defined, the display does not have a pointer.

Domain emptystring

Access Reference

Default Not applicable

4.1.11 SIZE dattribute

Definition SIZE specifies the size of the display expressed in the unit of measure defined by the display's UNITS dattribute.

Domain size

Access Reference

Default Not applicable

4.1.12 SPECTRUM dattribute

Definition If the value of the display's COLORTYPE dattribute is equal to "COLOR", SPECTRUM specifies the number of distinct color values (that is, the number of combinations of rcolor, gcolor and bcolor values that produce distinct colors) supported by the display. If the value of the display's COLORTYPE dattribute is equal to "GREY", SPECTRUM specifies the number of distinct grey-scale values supported by the display. If the value of the display's COLORTYPE dattribute is equal to "MONO", SPECTRUM is equal to 2.

Domain intexpr

Access Reference

Default Not applicable

4.1.13 TYPEFACE dattribute

Definition TYPEFACE provides, via descendant nodes, information about the type faces available for the display. Each type face is identified by an entry of the form

^$DI[SPLAY](display , tfacekeyword, fontface , fontsize)

where

tfacekeyword ::= expr V TYPEFACE

and where fontface and fontsize subscripts indicate available type face and size combinations. A fontsize value of 0 (zero) indicates a scalable type face.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

No value is present at the level of the TYPEFACE dattribute itself. For all descendant nodes, the value is an emptystring.

Domain emptystring

Access Reference

Default Not applicable

4.1.14 UNITS dattribute

Definition UNITS indicates the measurement units for the display. If the unit of measure is CHAR, the basis font is the font specified by an FFACE value of "M,DEFAULT", an FSIZE value of 12, and an FSTYLE value of "NORMAL".

Domain unitspec

Access Modify, Reference

Default "PIXEL"

5 Windows

The ssvn ^$WINDOW contains information about the windows accessible to a process, in the form

^$W[INDOW] (wname , wattribute)

wname is the identifier for a window.

5.1 Window Types

The MWAPI supports the following window types.

Application windows are the basic input/output mechanism of the MWAPI. Application windows can contain elements.

MTERM windows are terminal emulation windows that accept M input/output commands. The creation of elements within MTERM windows is reserved.

A window's type is indicated by a windowtype value.

windowtype ::= expr V	APPLICATION MTERM implementationvalue	

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

5.1.1 MTERM Windows

An MTERM window is opened by specifying the "MTERM" mnemonicspace in the OPEN command. If the device (i.e. the first expr in the openargument) specified in the OPEN command does not exist as a wname in ^$WINDOW, a window is created with wname equal to the device name and TYPE wattribute equal to "MTERM". If the device specified in the OPEN command exists as a wname in ^$WINDOW and that window's TYPE wattribute is equal to "MTERM", any attributes specified for that window are used. If the device specified in the OPEN command exists as a wname in ^$WINDOW and that window's TYPE wattribute is not equal to "MTERM", an error condition occurs with $ECODE containing ",M?5,".

Opening an MTERM window enables it to be seen by the user provided that it is not obscured by another window or positioned outside the visible area of its parent.

If an M CLOSE command designates an MTERM window that was created by the execution of an OPEN command, the window is destroyed. If an M CLOSE command designates an MTERM window that was created by the assignment of values to attributes in ^$WINDOW, the CLOSE command does not cause the window to be destroyed. If an application attempts to destroy an open MTERM window with an M KILL command, an error condition occurs with $ECODE containing ",M?6," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

5.2 Window Components

A window consists of:

An application area, a virtual surface on which gadgets, menus or other windows are placed. The application area has no explicit size.

A window viewport is an area in which the user can view some or all of the application area. If the application area is larger than the window viewport, it may be possible to scroll and/or resize the viewport so that a different portion of the application area is visible.

A window frame may surround the window viewport. The frame may contain the window title, may contain a menu bar, and may also contain controls that enable the user to resize the window, to iconify the window, and to display a system menu supplied by the underlying windowing platform.

When a window is created, the origin of its application area and its window viewport coincide and both have a position of (0,0).

5.3 Window Repainting

When a previously invisible window or element becomes visible or when all or part of a previously obscured window or element becomes unobscured, any necessary repainting is carried out automatically by the MWAPI implementation or by the underlying windowing platform, if any.

5.4 Window Attributes

Window attributes, identified by wattributes, define the characteristics of windows.

wattribute ::= expr V	ACTIVE BCOLOR COLOR	

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

DEFBUTTON
DISPLAY
EVENT
FCOLOR
FFACE
FSIZE
FSTYLE
ICON
ICONIFY
ID
ITITLE
MENUBAR
MIN
MODAL
NEXTG
PARENT
POS
RESIZE
SCROLL
SIZE
SIZEMIN
SIZEWIN
TIED
TITLE
TYPE
UNITS
VISIBLE
applicationattribute
implementationattribute

5.4.1 Applicability of Window Attributes

The applicability of wattributes to window types is specified by Table 1. A "Yes" entry indicates that the use of the attribute with the window type is defined. An empty entry indicates that the use of the attribute with the window type is reserved.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Table 1: Window Attributes		
Attribute	Application Window	M Term Window
ACTIVE	Yes	
BCOLOR	Yes	
COLOR	Yes	
DEFBUTTON	Yes	
DISPLAY	Yes	Yes
EVENT	Yes	
FCOLOR	Yes	
FFACE	Yes	
FSIZE	Yes	
FSTYLE	Yes	
ICON	Yes	
ICONIFY	Yes	
ID	Yes	
ITITLE	Yes	
MENUBAR	Yes	
MIN	Yes	
MODAL	Yes	Yes
NEXTG	Yes	
PARENT	Yes	
POS	Yes	Yes
RESIZE	Yes	Yes
SCROLL	Yes	
SIZE	Yes	Yes
SIZEMIN	Yes	
SIZEWIN	Yes	
TIED	Yes	
TITLE	Yes	Yes
TYPE	Yes	Yes
UNITS	Yes	Yes
VISIBLE	Yes	

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

5.4.2 ACTIVE wattribute

Definition — If ACTIVE is FALSE, the window and its descendant windows and elements are disabled. If a window or element is disabled, the user cannot interact with it and events do not occur for it.

If ACTIVE is TRUE for a window and all of its ancestor windows, if any, the window is not disabled.

Domain — truefalse

Access — Create, Modify, Reference

Default — TRUE

5.4.3 BCOLOR wattribute

Definition — BCOLOR specifies the default background color for gadgets subsequently created within the window. Modifying BCOLOR does not affect gadgets that already exist.

Domain — color

Access — Create, Modify, Reference

Default — Value of parent's BCOLOR dattribute or wattribute

5.4.4 COLOR wattribute

Definition — COLOR specifies the color of the window's application area.

Domain — color

Access — Create, Modify, Reference

Default — Value of parent's COLOR dattribute or wattribute

5.4.5 DEFBUTTON wattribute

Definition — DEFBUTTON specifies the default push button for the window. If the value of DEFBUTTON specifies a gadget which is a child of the window and which has a TYPE attribute value of "BUTTON", the user can "push" the specified button, that is cause it to generate a SELECT event, by pressing a key on the keyboard. The specific key to be pressed is defined by the underlying windowing platform.

If an application attempts to assign a value to DEFBUTTON which (a) does not specify a gadget which is a child of the window or (b) does not identify a gadget with a TYPE elattribute value equal to "BUTTON", an error condition occurs with $ECODE containing ",M?7," and $EREF containing a value that indicates the ssvn reference for which the invalid DEFBUTTON value was assigned.

The effect is not specified if DEFBUTTON identifies a gadget that has been destroyed after the value identifying it was assigned to DEFBUTTON.

Domain — gname

Access — Create, Modify, Reference

Default — Platform

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

5.4.6 DISPLAY wattribute

Definition DISPLAY specifies the display on which the window appears.

Domain display

Access Create, Reference

Default Value of $PDISPLAY

5.4.7 EVENT wattribute

Definition EVENT specifies, via descendant nodes identified by etype subscripts, the window's callback routine(s) for the event specified by etype. The format of a window event specification is defined in Section 0. No value is present at the level of the EVENT wattribute itself.

Domain L doargument

Access Create, Modify, Reference

Default None

5.4.8 FCOLOR wattribute

Definition FCOLOR specifies the default foreground color for gadgets subsequently created within the window. Modifying FCOLOR does not affect gadgets that already exist.

Domain color

Access Create, Modify, Reference

Default Value of parent's FCOLOR dattribute or wattribute

5.4.9 FFACE wattribute

Definition FFACE specifies the default type face that is used for subsequently created gadgets in the window.

Domain fontface

Access Create, Modify, Reference

Default "M,DEFAULT"

5.4.10 FSIZE wattribute

Definition FSIZE specifies the default type size that is used for subsequently created gadgets in the window.

Domain fontsize

Access Create, Modify, Reference

Default 12

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

5.4.11 FSTYLE wattribute

Definition FSTYLE specifies the default type style that is used for subsequently created gadgets in the window.

Domain fontstyle

Access Create, Modify, Reference

Default "NORMAL"

5.4.12 ICON wattribute

Definition ICON identifies the icon to display if the window is iconified.

Domain extresource

Access Create, Modify, Reference

Default Platform

5.4.13 ICONIFY wattribute

Definition If ICONIFY is TRUE, the window contains a means (e.g. a button on the window frame) for the user to iconify the window. If ICONIFY is FALSE and if the underlying windowing platform so permits, the window does not contain a means for the user to iconify the window.

Domain truefalse

Access Create, Reference

Default Platform

5.4.14 ID wattribute

Definition ID is an internal identifier for the window that assists in accessing this window in the underlying windowing platform.

Domain Unspecified

Access Reference

Default Not applicable

5.4.15 ITITLE wattribute

Definition ITITLE specifies text to appear with the icon when the window is iconified. If the value of ITITLE is equal to emptystring, then no text appears.

Domain chars

Access Create, Modify, Reference

Default Value of the window's TITLE wattribute

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

5.4.16 MENUBAR wattribute

Definition MENUBAR identifies the menu to be displayed in the window's menu bar. If the MENUBAR attribute is not specified when a window is created, no menu bar is created for the window. If the underlying windowing platform prevents a menu bar from being added after window creation, attempts to modify the MENUBAR attribute are ignored. If the value specified for MENUBAR is not the mname of a menu which is a child of the window, or if the value is equal to emptystring, an empty menu bar is created.

Domain mname or emptystring

Access Create, Modify, Reference

Default None

5.4.17 MIN wattribute

Definition If MIN is TRUE, the window is in the iconic state. If MIN is FALSE, the window is in the open (non-iconic) state. If the application assigns a value of TRUE to MIN, the window is made iconic.

Domain truefalse

Access Create, Modify, Reference

Default FALSE

5.4.18 MODAL wattribute

Definition A window for which the MODAL wattribute is defined is referred to as a modal window.

When a modal window is visible, other windows may be disabled. A MODAL wattribute value equal to "PARENT" indicates that the modal window disables only its parent window. A MODAL wattribute value equal to "ANCESTORS" indicates that the modal window disables all of its ancestor windows. A MODAL wattribute value equal to "APPLICATION" indicates that the modal window disables all active windows defined in ^$WINDOW except itself.

If an application attempts to create a child window of a modal window which is not also a modal window, an error condition occurs with $ECODE containing ",M?9," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

Domain modaltype

Access For an MTERM window: Reference

For other window types: Create, Reference

Default For an MTERM window: "APPLICATION"

For other window types: None

5.4.19 NEXTG wattribute

Definition NEXTG indicates the action that occurs if this window receives focus but no gadget within the window receives focus.
If an application attempts to assign a value to NEXTG that does not identify a gadget that has been defined for the window, an error condition occurs with $ECODE containing ",M?4," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

If the NEXTG wattribute for window W identifies a gadget G that cannot get focus, that gadget's NEXTG elattribute determines where focus goes. This process is repeated for successive Gs until (a) gadget G's NEXTG elattribute identifies a previously identified gadget G or gadget G's NEXTG elattribute is not specified, in which case focus remains at window W; or (b) gadget G's NEXT elattribute identifies a gadget G_n which can get focus is reached, in which case focus goes to gadget G_n.
The effect is not specified if NEXTG identifies a gadget whose ACTIVE elattribute is FALSE or if NEXTG identifies a gadget that is obscured or if NEXTG identifies a gadget that has been destroyed after the value identifying it was assigned to NEXTG.

Domain gname

Access Create, Modify, Reference

Default None

5.4.20 PARENT wattribute

Definition PARENT identifies the window that is this window's parent. If a window has a parent window, it (a) becomes invisible when its parent becomes invisible or is iconified; (b) if its VISIBLE attribute is TRUE, becomes visible when its parent becomes visible and is not iconified; and (c) is destroyed if its parent is destroyed.

If a window's parent is a display, the PARENT wattribute is not defined.

The effect of assigning a value for the PARENT wattribute which causes a single window to be both an ancestor and a descendant of another window is not specified.

Domain wname

Access Create, Reference

Default None

5.4.21 POS wattribute

Definition POS specifies the position of the origin of the window frame, expressed in the unit of measure defined by the unitspec component of the pos value, if present, or by the window's UNITS wattribute otherwise.

If the window's TIED wattribute is TRUE, the position is relative to the origin of the application area of the window's parent. If the unit of measure is CHAR, the basis font is the font specified by the parent's FFACE, FSIZE and FSTYLE wattributes. If the unit of measure is REL, the basis size is the size of the parent's application area when the parent is created.

If the window's TIED wattribute is FALSE, the position is relative to the origin of the display. If the unit of measure is CHAR, the basis font is the font specified by an FFACE value of "M,DEFAULT", an FSIZE value of 12, and an FSTYLE value of "NORMAL". If the unit of measure is REL, the basis size is the size of the display.

Domain pos

Access Create, Modify, Reference

Default Platform

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

5.4.22 RESIZE wattribute

Definition If RESIZE is TRUE, the user can alter the size of the window. If RESIZE is FALSE, the user cannot alter the size of the window.

Domain truefalse

Access Create, Reference

Default Platform

5.4.23 SCROLL wattribute

Definition If SCROLL is TRUE and a portion of the application area lies outside the window viewport, a horizontal and/or a vertical scroll bar is present, enabling the user to scroll the application area. Otherwise, scroll bars for scrolling the application area are not present, if the underlying windowing platform permits them to be omitted.

Domain truefalse

Access Create, Reference

Default Platform

5.4.24 SIZE wattribute

Definition SIZE specifies the width and height of the window viewport, expressed in the unit of measure defined by the unitspec component of the size value, if present, and by the window's UNITS wattribute otherwise.

If the unit of measure is CHAR, the basis font is the font specified by the window's FFACE, FSIZE and FSTYLE wattributes.

If the window's TIED wattribute is TRUE and the unit of measure is REL, the basis size is the size of the parent's window viewport when the parent is created. If the window's TIED wattribute is FALSE and the unit of measure is REL, the basis size is the size of the display.

When an application assigns a value to the SIZE wattribute of an existing window, if hsize or vsize is not specified then the horizontal or vertical size of the window, respectively, is not altered.

Domain size

Access Create, Modify, Reference

Default If hsize, vsize, or both are not specified when a window is created, defaults for the unspecified values are determined as described below.

When an application window is created, the MWAPI attempts to make it large enough to encompass all gadgets, defined at that time, whose POS elattributes do not contain negative values for hpos or vpos. Subsequent creation, destruction or movement of gadgets does not cause the SIZE value to change.

When an MTERM window is created, an implementation may determine a default size based on openarguments and usearguments specified for the window. If the default size is not determined in this way, the default size is the equivalent of "80,24,CHAR".

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

5.4.25 SIZEMIN wattribute

Definition SIZEMIN specifies the minimum size to which the user can shrink the window viewport, expressed in the unit of measure defined by the unitspec component of the size value, if present, and by the window's UNITS wattribute otherwise. Any attempt by the user to make the window viewport smaller than SIZEMIN is ignored.

Domain size

Access Create, Modify, Reference

Default Platform

5.4.26 SIZEWIN wattribute

Definition SIZEWIN specifies the overall size of the window (including the window frame, if any) expressed in the unit of measure defined by the window's UNITS wattribute. If the underlying windowing platforms does not provide this size exactly, a best estimate is provided.

If an application attempts to assign, with a single M MERGE command, values for both the SIZE and SIZEWIN wattributes, it is not specified which of these wattributes determines the window size.

Domain size

Access Create, Modify, Reference

Default Not applicable. If a value is not assigned to SIZEWIN, its value is determined implicitly by the value of SIZE.

5.4.27 TIED wattribute

Definition If TIED is TRUE (a) if this window and its parent overlap, this window is always on top of its parent; and (b) this window maintains its position, relative to its parent's origin, when its parent moves.

If an application attempts to assign a value to TIED which is equal to TRUE and the window's parent is a display, an error condition occurs with $ECODE containing ",M?2," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

Domain truefalse

Access Create, Reference

Default FALSE

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

5.4.28 TITLE wattribute

Definition TITLE specifies the window title text. If the TITLE attribute is not specified when a window is created, there may be no title area in the window border. If the underlying windowing platform prevents a title from being added or removed after window creation, attempts to modify the TITLE attribute are ignored. If the value specified for TITLE is an empty string, an empty title area is created.

The presence or absence of a TITLE value may be ignored if the underlying windowing platform does not permit or requires a title to be displayed for a window as a result of the values assigned to other wattributes.

Domain chars

Access Create, Modify, Reference

Default For an application window, none. For an MTERM window, the default is the value of the wname that identifies the window.

5.4.29 TYPE wattribute

Definition TYPE specifies the window type.

Domain windowtype

Access Create, Reference

Default For windows implicitly created by an M OPEN command , "MTERM".

For other windows, "APPLICATION".

5.4.30 UNITS wattribute

Definition UNITS indicates the unit of measure used to specify the size and position of the window.

Domain unitspec

Access Create, Modify, Reference

Default For an application window:

Value of parent's UNITS dattribute or wattribute

For an MTERM window:

"CHAR", with a basis font defined by an FFACE value of "M,FIXED", an FSIZE value of "12", and an FSTYLE value of "NORMAL".

5.4.31 VISIBLE wattribute

Definition If VISIBLE is FALSE, a window cannot be seen by the user. If VISIBLE is TRUE, a window can be seen by the user provided that it is not obscured and it is not a child of an invisible window.

--`,,```,,,,````-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Domain	truefalse
Access	Create, Modify, Reference
Default	TRUE

6 Elements

Elements perform specialized tasks within windows. Elements include gadgets, menus, and timers.

6.1 Gadgets

The ssvn ^$WINDOW contains information about the gadgets accessible to a process, in the form

^$W[INDOW] (wname , gkeyword , gname , elattribute)

gname is the identifier for a gadget and wname is the identifier for the window that is the parent of that gadget.

gkeyword ::= expr V G

The MWAPI defines the following gadget types, specified via gtype value for the TYPE elattribute.

gtype ::= expr V	BUTTON CHECK DOCUMENT FRAME GENERIC LABEL LIST LISTBUTTON LISTENTRY LONGLIST RADIO SCROLL SYMBOL TEXT implementationvalue

6.1.1 Check Box

A check box gadget enables the user or the application to turn an indicator on or off. Typically, a check box appears as text accompanied by an area that can be selected or deselected. A check box has a TYPE elattribute value equal to "CHECK".

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.1.2 Document

A document gadget enables the user to view, enter, and modify one or more lines of text. Text editing capabilities are provided by the underlying windowing platform or the MWAPI. If the text cannot all be simultaneously displayed in the gadget, horizontal and vertical scrolling mechanisms can be specified. A document has a TYPE elattribute value equal to "DOCUMENT".

6.1.3 Generic Box

A generic box is an area in which an application can draw text, geometric figures, and other graphical objects and can receive events. A generic box has a TYPE elattribute value equal to "GENERIC".

6.1.4 Group Frame

A group frame is a rectangular outline used to visually group gadgets on a window. A group frame can also be a vertical or horizontal line. A group frame cannot get focus and events do not occur for it. A group frame has a TYPE elattribute value equal to "FRAME".

6.1.5 Label

A label is static text that appears within a window. A label cannot get focus and events do not occur for it. A label has a TYPE elattribute value equal to "LABEL".

6.1.6 List Box

A list box displays a list of items and enables the user to select one or more of them. If the list box is not large enough to show all items at once, a scrolling mechanism enables the user to move through the list. Such navigation through the list is handled automatically by the underlying windowing platform and the MWAPI. A list box has a TYPE elattribute value equal to "LIST".

6.1.7 List Button

A list button is a combination of a push button and a list. Normally, the list is invisible. When the push button is pushed, the list becomes visible. When the user selects an item from the list, the list becomes invisible. The currently selected value is displayed with the push button. A list button has a TYPE elattribute value equal to "LISTBUTTON".

6.1.8 List Entry Box

A list entry box is a combination of a text entry area (similar to a text gadget) and a iist area (similar to a list box gadget.) It enables a user to select an item from the list area, which is then displayed in the text entry area. The user can enter or modify text in the text entry area. The resulting value need not be present in the list area. A list entry box has a TYPE elattribute value equal to "LISTENTRY".

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.1.9 Long List Box

Like a list box, a long list box displays a list of items and enables the user to select one or more of them. Unlike a list box, however, navigation through the list is handled by the application. The long list box is intended for use with lists that are very large, enabling an application to optimize its construction of the list in response to user actions. A long list box has a TYPE elattribute value equal to "LONGLIST".

6.1.10 Push Button

A push button is a gadget that the user can "press", typically to cause an event to occur. A push button has a TYPE elattribute value equal to "BUTTON".

6.1.11 Radio Button Set

A radio button set is a collection of related items that the user can select. At any time, at most one item can be selected. Selecting an item causes any previously selected item to be deselected. A radio button set has a TYPE elattribute value equal to "RADIO".

6.1.12 Scroll

A scroll gadget visually represents a value on a numeric scale through some sort of position indicator. The user can modify the value by moving the position indicator. A scroll gadget does not directly cause anything to scroll, although applications may employ scroll gadgets to do so. A scroll gadget can be horizontally or vertically oriented. A scroll gadget has a TYPE elattribute value equal to "SCROLL".

6.1.13 Symbol

A symbol is an image to be displayed within a window. Symbol gadgets cannot get focus and events do not occur for them. A symbol has a TYPE elattribute value equal to "SYMBOL".

6.1.14 Text

A text gadget enables the user to view, enter, and modify a single line of text. Text editing capabilities are provided by the underlying windowing platform or the MWAPI. If the text cannot all be simultaneously displayed in the gadget, horizontal scrolling mechanisms are provided. A text gadget has a TYPE elattribute value equal to "TEXT".

6.2 Menus

A menu is a list of choices from which a user makes a selection, typically to cause an action to occur immediately. The MWAPI defines two types of menus: a menu bar which typically displays choices horizontally and appears at the top of a window; and a pop-up menu which typically displays choices vertically. A pop-up menu can appear descendant from a menu bar or somewhere else in the window, completely independent of the menu bar.

The ssvn ^$WINDOW contains information about the menus accessible to a process, in the form

^$W[INDOW] (wname , mkeyword , mname , elattribute)

mname is the identifier for a menu and wname is the identifier for the window that is the parent of that menu.

mkeyword ::= expr V M

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.3 Timers

A timer generates an event after a specified time interval elapses. It has no visible representation.

The ssvn ^$WINDOW contains information about the timers accessible to a process, in the form

^$W[INDOW] (wname , tkeyword , tname , elattribute)

tname is the identifier for a timer and wname is the identifier for the window that is the parent of that timer.

tkeyword ::= expr V T

Provided that event processing is activated and that the ACTIVE wattribute has a value of TRUE for all ancestor windows and that the timer's INTERVAL elattribute has a valid value and that timer events are enabled for a timer, an interval timer starts (a) whenever a value is assigned to one or more of the timer's elattributes or (b) whenever a value of TRUE is assigned to the ACTIVE wattribute of an ancestor window or (c) the application executes an explicit or implicit M QUIT command from a callback routine processing a timer event for the timer.

An interval timer is cancelled (a) whenever a value is assigned to or removed from one or more of the timer's elattributes or (b) the timer is destroyed or (c) whenever a value of FALSE is assigned to the ACTIVE wattribute of an ancestor window.

If, after a timer starts, the time period specified by the timer's INTERVAL elattribute elapses without the interval timer being cancelled, the interval timer expires and a timer event occurs for the timer.

6.4 Element Attributes

Element attributes, identified by elattributes, define the characteristics of elements.

elattribute ::= expr V	ACTIVE BCOLOR CANCEL CANCHANGE CHANGED CHARMAX CHOICE DRAW DRAWTYPE EVENT FCOLOR FFACE FRAMED FSIZE FSTYLE ID INSELECT INTERVAL NEXTG POS RESOURCE ROWCOL SCROLL SCROLLBY SCROLLDIR

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

SCROLLPOS
SCROLLRANGE
SELECTMAX
SELECTVAL
SIZE
TBCOLOR
TFCOLOR
TFFACE
TFSIZE
TFSTYLE
TITLE
TOPSHOW
TPOS
TYPE
UNITS
VALUE
VISIBLE
applicationattribute
implementationattribute

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Table 2: Element Attributes

	Check Box	Docu-ment	Generic Box	Group Frame	Label	List Box	List Button	List Entry Box
ACTIVE	Yes	Yes	Yes			Yes	Yes	Yes
BCOLOR	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
CANCEL	Yes	Yes	Yes			Yes	Yes	Yes
CANCHANGE		Yes				Yes	Yes	Yes
CHANGED	Yes	Yes				Yes	Yes	Yes
CHARMAX		Yes						Yes
CHOICE						Yes	Yes	Yes
DRAW			Yes					
DRAWTYPE			Yes					
EVENT	Yes	Yes	Yes			Yes	Yes	Yes
FCOLOR	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
FFACE		Yes	Yes			Yes	Yes	Yes
FRAMED		Yes	Yes		Yes			
FSIZE		Yes	Yes			Yes	Yes	Yes
FSTYLE		Yes	Yes			Yes	Yes	Yes
ID	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
INSELECT		Yes						Yes
INTERVAL								
NEXTG	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
POS	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
RESOURCE								
ROWCOL								
SCROLL		Yes						
SCROLLBY								
SCROLLDIR								
SCROLLPOS								
SCROLLRANGE								
SELECTMAX						Yes		
SELECTVAL		Yes						Yes
SIZE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
TBCOLOR		Yes				Yes	Yes	Yes
TFCOLOR		Yes				Yes	Yes	Yes
TFFACE	Yes	Yes		Yes	Yes	Yes	Yes	Yes
TFSIZE	Yes	Yes		Yes	Yes	Yes	Yes	Yes
TFSTYLE	Yes	Yes		Yes	Yes	Yes	Yes	Yes
TITLE	Yes	Yes		Yes	Yes	Yes	Yes	Yes
TOPSHOW						Yes		Yes
TPOS		Yes				Yes	Yes	Yes
TYPE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
UNITS	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
VALUE	Yes	Yes				Yes	Yes	Yes
VISIBLE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Table 2: Element Attributes (Continued)

	Long List Box	Menu	Push Button	Radio Button Set	Scroll	Symbol	Text	Timer
ACTIVE	Yes	Yes	Yes	Yes	Yes		Yes	Yes
BCOLOR	Yes			Yes	Yes		Yes	
CANCEL	Yes		Yes	Yes			Yes	
CANCHANGE	Yes						Yes	
CHANGED	Yes			Yes	Yes		Yes	
CHARMAX							Yes	
CHOICE	Yes	Yes		Yes				
DRAW								
DRAWTYPE								
EVENT	Yes		Yes	Yes	Yes		Yes	Yes
FCOLOR	Yes			Yes	Yes		Yes	
FFACE	Yes			Yes			Yes	
FRAMED				Yes			Yes	
FSIZE	Yes			Yes			Yes	
FSTYLE	Yes			Yes			Yes	
ID	Yes	Yes	Yes	Yes	Yes		Yes	
INSELECT							Yes	
INTERVAL								Yes
NEXTG	Yes		Yes	Yes	Yes		Yes	
POS	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
RESOURCE			Yes			Yes		
ROWCOL				Yes				
SCROLL								
SCROLLBY					Yes			
SCROLLDIR					Yes			
SCROLLPOS	Yes							
SCROLLRANGE	Yes				Yes			
SELECTMAX	Yes							
SELECTVAL							Yes	
SIZE	Yes		Yes	Yes	Yes		Yes	
TBCOLOR	Yes			Yes			Yes	
TFCOLOR	Yes			Yes			Yes	
TFFACE	Yes		Yes	Yes			Yes	
TFSIZE	Yes		Yes	Yes			Yes	
TFSTYLE	Yes		Yes	Yes			Yes	
TITLE	Yes		Yes	Yes			Yes	
TOPSHOW	Yes							
TPOS	Yes			Yes			Yes	
TYPE	Yes		Yes	Yes	Yes	Yes	Yes	
UNITS	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
VALUE	Yes			Yes	Yes		Yes	
VISIBLE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.4.1 ACTIVE elattribute

Definition If ACTIVE is FALSE, the element is disabled; the user cannot interact with the element and events do not occur for it.

If ACTIVE is TRUE, the element is enabled.

Domain truefalse

Access Create, Modify, Reference

Default TRUE

6.4.2 BCOLOR elattribute

Definition BCOLOR specifies the background color for the gadget.

Domain color

Access Create, Modify, Reference

Default Value of parent's BCOLOR wattribute

6.4.3 CANCEL elattribute

Definition If CANCEL is TRUE for an element then, when the element receives focus, any change or unfocus events that would have occurred for the gadget that previously had focus do not occur.

Domain truefalse

Access Create, Modify, Reference

Default FALSE

6.4.4 CANCHANGE elattribute

Definition If CANCHANGE is TRUE, user actions can cause the VALUE attribute for the gadget to change. If CANCHANGE is FALSE, user actions cannot cause the VALUE attribute for the gadget to change, although some user interaction with the gadget may be possible.

Domain truefalse

Access Create, Modify, Reference

Default TRUE

6.4.5 CHANGED elattribute

Definition CHANGED is set to TRUE by the MWAPI implementation whenever the user causes the VALUE elattribute for the gadget to change, even if the net result of a series of modifications leaves the VALUE elattribute effectively unchanged.

Domain truefalse

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Access Create, Modify, Reference

Default FALSE

6.4.6 CHARMAX elattribute

Definition CHARMAX specifies the maximum number of characters the text value of a gadget can have as a result of user entry or modification of text. Embedded linend characters are included in the count. The maximum character limit may not be enforced for modifications of the text value by the application. A value of 0 (zero) indicates that there is no explicit limit on the text value length.

Domain intexpr

Access Create, Reference

Default 0 (zero)

6.4.7 CHOICE elattribute

Definition CHOICE specifies, via descendant nodes identified by item subscripts, the text to be displayed for choices.

For a gadget, the value of an entry of the form

^$W[INDOW](wname , gkeyword , gname , ckeyword , item)

specifies the text that appears for the choice identified by item for the gadget identified by gname.

ckeyword ::= expr V CHOICE

For a menu, the value of an entry of the form

^$W[INDOW](wname , mkeyword , mname , ckeyword , item)

specifies the text that appears for the choice identified by item for the menu identified by mname.

No value is present at the level of the CHOICE elattribute itself.

Domain chars

Access Create, Modify, Reference

Default item

6.4.8 DRAW elattribute

Definition DRAW specifies the number of drawcommands that are descendants of the DRAW elattribute of the gadget. An application cannot assign a value to DRAW. An application can use the M KILL command to remove the DRAW elattribute and its descendant drawcommands. This removes the drawing specified by the drawcommands and the DRAW elattribute is set to 0 (zero).

The format of drawcommands which are descendant to DRAW is specified in Section 0.

Domain intlit

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Access Modify, Reference

Default 0 (zero)

6.4.9 DRAWTYPE elattribute

Definition DRAWTYPE specifies the type of drawcommands that can be specified in the DRAW attribute of the gadget.

Domain drawtype

Access Create, Reference

Default MDRAW

6.4.10 EVENT elattribute

Definition EVENT specifies, via descendant nodes identified by etype subscripts, the element's callback routine(s) for the event specified by etype. The format of an element event specification is defined in Section 0. No value is present at the level of the EVENT elattribute itself.

Domain L doargument

Access Create, Modify, Reference

Default None

6.4.11 FCOLOR elattribute

Definition FCOLOR specifies the foreground color for the gadget. For gadgets containing text, FCOLOR specifies the color of the text.

Domain color

Access Create, Modify, Reference

Default Value of parent's FCOLOR wattribute

6.4.12 FFACE elattribute

Definition FFACE specifies the type face used for all text that appears within the gadget except the title.

Domain fontface

Access Create, Reference

Default Value of parent window's FFACE wattribute

6.4.13 FRAMED elattribute

Definition If FRAMED is TRUE, a frame appears at the perimeter of the gadget.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Domain truefalse

Access Create, Reference

Default For label gadgets, FALSE. For other gadget types, TRUE.

6.4.14 FSIZE elattribute

Definition FSIZE specifies the type size used for all text that appears within the gadget except the title.

Domain fontsize

Access Create, Reference

Default Value of parent window's FSIZE wattribute

6.4.15 FSTYLE elattribute

Definition FSTYLE specifies the type style used for all text that appears within the gadget except the title.

Domain fontstyle

Access Create, Reference

Default Value of parent window's FSTYLE wattribute

6.4.16 ID elattribute

Definition ID is an internal identifier for the element that assists in accessing this element in the underlying windowing platform.

Domain Unspecified

Access Reference

Default Not applicable

6.4.17 INSELECT elattribute

Definition INSELECT indicates the insertion point within a text value and a count of the number of characters selected. New characters inserted into the text value are added immediately following the position indicated by the insert value.

If seloff is 0 (zero), no characters are selected. If seloff is greater than 0 (zero), the seloff characters immediately following the position identified by insert (that is, characters insert+1 through insert+seloff) are selected. If seloff is less than 0 (zero), the seloff characters immediately preceding the position identified by insert (that is, characters insert+1+seloff through insert) are selected.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

When an application attempts to assign a value to the INSELECT elattribute, the following series of actions are performed. If the value of insert is less than 0 (zero), it is set to 0 (zero). If the value of insert exceeds the length of the text value, insert is set to indicate the last character of the text value. If the value of seloff would select characters before the beginning of or after the end of the data value, seloff is set, respectively, to indicate that the selection extends to the first or final character in the value.

If there is no insertion point within the text value of a gadget, its INSERT elattribute is not defined. If no VALUE elattribute is defined, INSELECT is not defined and is not modifiable. If an application assigns values to a gadget's INSELECT and VALUE elattributes with a single M MERGE command, the result is as if VALUE were defined first.

Domain insel

Access Create, Modify, Reference

Default Platform

6.4.18 INTERVAL elattribute

Definition INTERVAL specifies the time, in seconds, that a timer runs following initiation before generating a timer event.

If an application attempts to assign a value to INTERVAL which is not greater than 0 (zero), an error condition occurs with $ECODE containing ",M?2," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

Domain numexpr

Access Create, Modify, Reference

Default None

6.4.19 NEXTG elattribute

Definition NEXTG indicates the action that occurs if, while this gadget has focus, the user presses the key sequence that requests movement to the next gadget. The key sequence that requests movement to the next gadget is determined by the underlying windowing platform or the MWAPI implementation.

If an application attempts to assign a value to NEXTG which does not identify a gadget that has been defined for the same window, an error condition occurs with $ECODE containing ",M?4," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

If the NEXTG elattribute for gadget G_1 identifies a gadget G that cannot get focus, that gadget's NEXTG elattribute determines where focus goes. This process is repeated for successive Gs until (a) gadget G's NEXTG elattribute identifies gadget G_1 or a previous gadget G, or gadget G's NEXTG elattribute is not specified, in which case focus remains at gadget G_1; or (b) gadget G's NEXT elattribute identifies a gadget G_n which can get focus is reached, in which case focus goes to gadget G_n.

The effect is not specified if NEXTG identifies a gadget whose ACTIVE or VISIBLE elattribute is FALSE or if NEXTG identifies a gadget that is obscured or if NEXTG identifies a gadget that has been destroyed after the value identifying it was assigned to NEXTG.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Domain gname

Access Create, Modify, Reference

Default None

6.4.20 POS elattribute

Definition POS specifies the position of the origin of the element with respect to the origin of its parent's application area, expressed in the unit of measure defined by the unitspec component of the pos value, if present, and by the element's UNITS elattribute otherwise. For a document, list box, list button, list entry box, long list box, radio button set or text gadget, the position excludes the gadget's title, if any. For a group frame, the position may exclude the gadget's title, if any.

If the unit of measure is CHAR, the basis font is the font specified by the parent's FFACE, FSIZE and FSTYLE wattributes. If the unit of measure is REL, the basis size is the size of the parent's window viewport when the parent is created.

For gadgets, if a value is not assigned to the POS elattribute when a gadget is created, an error condition occurs with $ECODE containing ",M?7," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

For menus displayed as menu bars and for menus displayed as submenus, the value of the POS elattribute is ignored. For other menus, POS is optional.

Domain pos

Access Create, Modify, Reference

Default For menus not displayed as menu bars or as submenus, POS defaults to the position of the pointer at the time the menu becomes visible.

For other elements, none.

6.4.21 RESOURCE elattribute

Definition RESOURCE specifies an image to display in the gadget.

If, when a symbol gadget is created, an application does not assign a value to the RESOURCE elattribute, an error condition occurs with $ECODE containing ",M?7," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

If an application attempts to assign a value to the RESOURCE elattribute which is not valid, an error condition occurs with $ECODE containing ",M?2," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

The effect is not specified if the image identified by RESOURCE becomes invalid or unavailable after the value identifying it has been assigned to the RESOURCE elattribute.

Domain For a symbol gadget, symbol. For other gadgets, extresource.

Access Create, Reference

Default None

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.4.22 ROWCOL elattribute

Definition ROWCOL specifies how choices are arranged within rows and columns in a radio button set gadget. If a gadget contains N choices, they are displayed in a matrix of R rows by C columns. If rowcol is equal to "ROW", R is equal to rowcolsize and C is equal to the M numeric interpretation of

N\R+(N#R>0)

If rowcol is equal to "COL", C is equal to rowcolsize and R is equal to the M numeric interpretation of

N\C+(N#C>0)

If filldir is equal to "H", then rows are filled first. That is, all cells in row n are filled before any cells in row n+1 are filled. If filldir is equal to "V", then columns are filled first. That is, all cells in column n are filled before any cells in column n+1 are filled. It is possible to have empty cells, rows and columns.

If an application attempts to assign a rowcolsize value that is less than 1 (one), an error occurs with $ECODE containing ",M?2," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

All columns have the same width. If the SIZE elattribute is specified for the gadget, the column width is approximately equal to hsize/C. If the SIZE elattribute is not specified for the gadget, the column width is approximately equal to the widest choice specified for the gadget.

Domain rowcolspec

Access Create, Reference

Default If ROWCOL is not specified it defaults to "COL,1,V".

If intlit is not specified, it defaults to 1 (one).

If filldir is not specified, it defaults to "V".

6.4.23 SCROLL elattribute

Definition SCROLL specifies scrolling behavior for a gadget's data.

If the value of the hsize component of SCROLL is greater than 0 (zero), horizontal scrolling can occur and the horizontal size of the data is specified by hsize. If the value of the hsize component of SCROLL is equal to 0 (zero), horizontal scrolling can occur but no explicit horizontal size is specified for the data. If hsize is omitted, horizontal scrolling cannot occur.

If the value of the vsize component of SCROLL is greater than 0 (zero), vertical scrolling can occur and the vertical size of the data is specified by vsize. If the value of the vsize component of SCROLL is equal to 0 (zero), vertical scrolling can occur but no explicit vertical size is specified for the data. If vsize is omitted, vertical scrolling cannot occur.
The unit of measure is determined in the same manner as is specified for the SIZE elattribute.

A scroll bar may or may not be present when it is not required.

If an application attempts to assign a SCROLL value that contains a negative value for hsize or vsize, an error occurs with $ECODE containing ",M?2," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Domain size

Access Create, Reference

Default FALSE

6.4.24 SCROLLBY elattribute

Definition SCROLLBY specifies the amount by which the VALUE elattribute increases or decreases when the user presses a scroll gadget's movement control. If the user presses a small movement control, the scroll gadget's VALUE elattribute is increased by the amount specified by the scrollsmall component of the SCROLLBY value. If the user presses a large movement control, the scroll gadget's VALUE elattribute is increased by the amount specified by the scrollbig component of the SCROLLBY value.

If an application attempts to assign a scrollby value that contains a negative value for scrollsmall or scrollbig, an error occurs with $ECODE containing ",M?2," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

Domain scrollby

Access Create, Modify, Reference

Default For scrollsmall, the default is 1 (one).

For scrollbig, the default is equal to the M numeric interpretation of

X-Y/5

Where X and Y are equal to, respectively, the scrollmax and scrollmin components of the value of the gadget's SCROLLRANGE elattribute.

6.4.25 SCROLLDIR elattribute

Definition SCROLLDIR specifies the orientation of a scroll gadget. If the value of SCROLLDIR is "H", the scroll gadget is displayed horizontally. If the value of SCROLLDIR is "V", the scroll gadget is displayed vertically.

Domain orientation

Access Create, Reference

Default "H"

6.4.26 SCROLLPOS elattribute

Definition SCROLLPOS specifies the position of the position indicator in a long list box, relative to the scale established by the SCROLLRANGE elattribute. If SCROLLPOS is not defined, no position indicator is present in the long list box.

Domain numexpr

Access Create, Modify, Reference

Default None

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.4.27 SCROLLRANGE elattribute

Definition SCROLLRANGE specifies the value range that a scroll gadget or long list box displays. The position of a scroll gadget's value indicator or of a long list box's position indicator is determined by the scroll value compared to the range specified by the SCROLLRANGE elattribute. For a scroll gadget, the scroll value is equal to the numeric value of the VALUE elattribute. For a long list box gadget, the scroll value is equal to the numeric value of the SCROLLPOS elattribute.

If an application attempts to assign a value to a scroll gadget's VALUE elattribute that is greater than scrollmax, the VALUE attribute is set to scrollmax. If an application attempts to assign a value to a scroll gadget's VALUE elattribute that is less than scrollmin, the VALUE attribute is set to scrollmin. If an application attempts to assign to a gadget's SCROLLRANGE elattribute a value with scrollmax less than scrollmin, the effect is not specified.

Domain scrollrange

Access Create, Modify, Reference

Default The default for scrollmax is 100. The default for scrollmin is 0 (zero).

6.4.28 SELECTMAX elattribute

Definition For certain gadgets with choices, SELECTMAX specifies the maximum number of choices that can be concurrently selected. The user is prevented from selecting more than the specified number of choices. A value of 0 (zero) indicates that no limit is imposed.

Domain intexpr

Access Create, Reference

Default 1 (one)

6.4.29 SELECTVAL elattribute

Definition SELECTVAL contains the selected data associated with or represented by the gadget.

If the gadget's INSELECT elattribute is not defined, its SELECTVAL elattribute is also not defined and is not modifiable.

If an application assigns a value, X, to SELECTVAL, the value of X replaces the selected data in the gadget's data value. If an application removes the SELECTVAL with the M KILL command, the selected data is removed from the gadget's data value. Any changes made to a gadget's data value via SELECTVAL are reflected in the gadget's VALUE elattribute and may also cause modification of the gadget's INSELECT elattribute.

Domain For a document gadget, longchars.

For a list entry box or text gadget, chars.

Access Modify, Reference

Default Not Applicable

--`,,```,,,,````-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.4.30 SIZE elattribute

Definition

SIZE specifies the size of the gadget, expressed in the unit of measure defined by the unitspec component of the size value, if present, and by the gadget's UNITS elattribute otherwise.

If the unit of measure is not CHAR and a gadget is rectangular, SIZE specifies the overall size of the gadget. If the unit of measure is not CHAR and a gadget is not rectangular, SIZE specifies the size of the smallest rectangle that can bound the gadget.

If the unit of measure is CHAR, the basis font is the font specified by the gadget's FFACE, FSIZE and FSTYLE elattributes. For a check box, document, label, list box, list button, list entry box, long list box, push button, radio button set, or text gadget, if the unit of measure is CHAR, the gadget is large enough to accommodate vsize lines of hsize characters each using the basis font; the overall size of the gadget may be larger. For other gadget types, a unit of measure of CHAR specifies the overall size of the gadget.

If the unit of measure is REL, the basis size is the size of the parent's window viewport when the parent is created.

For a group frame, hsize and vsize values greater than 0 (zero) produce a rectangle; an hsize value of 0 (zero) produces a vertical line; a vsize value of 0 (zero) produces a horizontal line. If an application attempts to assign to the SIZE elattribute of a group frame a size value for which hsize and vsize are both 0 (zero), an error condition occurs with $ECODE containing ",M?2," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

For a generic box or group frame gadget, if a value is not assigned to the SIZE elattribute when the gadget is created, an error condition occurs with $ECODE containing ",M?7", and $EREF containing a value that indicates the ssvn reference for which the error occurred.

Whenever a size value specifies the overall size of a gadget, the size includes the gadget's border, if any. For a check box, label or push button gadget, the overall size of a gadget includes the gadget's title, if any. For a document, list box, list button, list entry box, long list box, radio button set or text gadget, the overall size of a gadget excludes the gadget's title, if any. For a group frame, the overall size of a gadget may exclude the gadget's title, if any.

For a list button, size includes the size of the button and the list. The actual height of the list button gadget may be whatever is required to contain the choices and some platforms may not allow the list to be scrolled.

Assigning a value to the SIZE elattribute for a gadget does not imply any automatic scaling of the gadget's contents.

When an application assigns a value to the SIZE elattribute of an existing gadget, if hsize or vsize is not specified then the horizontal or vertical size of the gadget, respectively, is not altered.

Domain

size

Access

Create, Modify, Reference

Default

For a check box, document, symbol, label, list box, list button, list entry box, long list box, push button, radio button set, scroll , or text element, if hsize, vsize, or both are not specified when a gadget is created, the default for each unspecified value is platform.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.4.31 TBCOLOR elattribute

Definition TBCOLOR specifies the background color for the gadget's title. If the underlying windowing platform does not enable the title to have a different background color from the remainder of the gadget, the value of the TBCOLOR elattribute is ignored.

Domain color

Access Create, Modify, Reference

Default Value of gadget's BCOLOR elattribute

6.4.32 TFCOLOR elattribute

Definition TFCOLOR specifies the color for the gadget's title text. If the underlying windowing platform does not enable the title text to have a different color from the remainder of the gadget's text, the value of the TFCOLOR elattribute is ignored.

Domain color

Access Create, Modify, Reference

Default Value of gadget's FCOLOR elattribute

6.4.33 TFFACE elattribute

Definition TFFACE specifies the type face used for the gadget's title text and for menu choices.

Domain fontface

Access Create, Reference

Default Value of parent window's FFACE wattribute

6.4.34 TFSIZE elattribute

Definition TFSIZE specifies the type size used for the gadget's title text and for menu choices.

Domain fontsize

Access Create, Reference

Default Value of parent window's FSIZE wattribute

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.4.35 TFSTYLE elattribute

Definition TFSTYLE specifies the type style used for the gadget's title text and for menu choices.

Domain fontstyle

Access Create, Reference

Default Value of parent window's FSTYLE wattribute

6.4.36 TITLE elattribute

Definition TITLE specifies text that appears within or adjacent to a gadget.

If the value of the TITLE elattribute contains a mnemonichar and the gadget is of a type that can get focus, the character following the mnemonichar is a mnemonic character that may enable the user, via the keyboard, to give focus to the gadget and the mnemonichar is not displayed. If two mnemonichars are immediately adjacent within a TITLE value, no mnemonic character is defined and a single mnemonichar appears in the displayed title text. If the value of the TITLE elattribute contains a mnemonichar and the gadget is not of a type that can get focus, the mnemonichar is ignored. The character mnemonichar cannot itself be used as a mnemonic character.

For a check box gadget, TITLE specifies the text that appears adjacent to the selection area.

For a group frame gadget, TITLE specifies text that appears within or overlaying the frame.

For a label gadget, TITLE specifies the label text.

For a document, list box, list button, list entry box, long list box, radio button set, or text gadget, TITLE specifies text that appears adjacent to the gadget.

For a push button, TITLE specifies the text that appears on the button.

Domain chars

Access Create, Reference

Default For a label gadget, TITLE defaults to an emptystring. For other gadget types, TITLE defaults to none.

6.4.37 TOPSHOW elattribute

Definition TOPSHOW specifies which choice is displayed at the top of the visible area of certain gadget types that display choices. If the user scrolls the choices, the value of TOPSHOW changes.

If the application assigns a value to TOPSHOW, the list of choices may scroll.

If the value assigned to TOPSHOW exactly matches an item subscript specified for the gadget, the choice identified by that item subscript is displayed at the top of the visible area.

If the application assigns a value to TOPSHOW which does not exactly match an item subscript specified for the gadget, the choice identified by the first item subscript that follows (in the collation sequence of ^$WINDOW) the specified TOPSHOW value is displayed at the top of the visible area. TOPSHOW is then set to that item value.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

If the application assigns a value to TOPSHOW which does not exactly match an item subscript specified for the gadget and which is not followed (in the collation sequence of ^$WINDOW) by any item subscript specified for the gadget, the list of choices scrolls (if necessary) so that the last choice in the list is displayed in the visible area. TOPSHOW is then set to the item displayed at the top of the visible area.

Domain item or emptystring

Access Create, Modify, Reference

Default item subscript of first choice in the list, if any; emptystring otherwise.

6.4.38 TPOS elattribute

Definition TPOS specifies the position of a title relative to the gadget it is associated with.

Domain titlepos

Access Create, Reference

Default "TOP"

6.4.39 TYPE elattribute

Definition TYPE specifies the type of the gadget. If an application does not assign a value for the TYPE elattribute when attempting to create a gadget, an error condition occurs with $ECODE containing ",M?7," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

Domain gtype

Access Create, Reference

Default Not applicable

6.4.40 UNITS elattribute

Definition UNITS specifies the unit of measure for the element.

Domain unitspec

Access Create, Modify, Reference

Default Value of parent's UNITS wattribute

6.4.41 VALUE elattribute

Definition VALUE specifies the data associated with or represented by the gadget.

For a check box gadget, if the user selects the gadget, VALUE is set to 1 (one). If the user

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

deselects the gadget, VALUE is set to 0 (zero). If a value that is assigned by the application evaluates to TRUE, the gadget is selected. If a value that is assigned by the application evaluates to FALSE, the gadget is deselected.

For a document, list entry box or text gadget, VALUE specifies the text value. Within a document text value, each linend indicates that, when the text value is displayed, the character following the linend should begin a new line. Otherwise, the formatting mechanisms used for displaying a document text value are determined by the underlying windowing platform, if there is one, and by the MWAPI otherwise.

For a list box or long list box gadget, VALUE specifies the number of choices selected. The choices selected are indicated by descendant nodes, of the form

^$W[INDOW] (wname , gkeyword , gname , vkeyword , item)

where

vkeyword ::= expr V VALUE

and item identifies a selected choice. The value of these nodes is the emptystring. If an application attempts to create a descendant node of VALUE which has an item that does not identify a defined choice for the gadget, an error condition occurs with $ECODE containing ",M?3," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

For a list box or long list box gadget, an application can assign values to descendant nodes of VALUE, but it cannot assign a value to the VALUE node itself. However, an application can use an M KILL command of the form

^$W[INDOW] (wname , gkeyword , gname , vkeyword)

to deselect all current selections in a list box or long list box.

For a radio button set or list button gadget, VALUE specifies the item value of the selected choice. If an application attempts to assign to VALUE an item that does not identify a defined choice for the gadget, an error condition occurs with $ECODE containing ",M?3," and $EREF containing a value that indicates the ssvn reference for which the error occurred. If no choice is selected, the value of VALUE is an emptystring.

For a scroll gadget, VALUE specifies the value represented by the value indicator.

Domain For a check box gadget, truefalse.

For a document gadget, longchars.

For a list entry box or text gadget, chars.

For a list box or long list box gadget, VALUE has a domain of intexpr and descendant nodes have a domain of emptystring.

For a radio button set or list button gadget, item or emptystring.

For a scroll gadget, numexpr.

Access Create, Modify, Reference

Default For a check box gadget, 0 (zero).

For a document, list button, list entry box, radio button set, or text gadget, emptystring.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

For a list box or long list box gadget, 0 (zero) with no descendant nodes.

For a scroll gadget, the value of the scrollmin component of the SCROLLRANGE elattribute.

6.4.42 VISIBLE elattribute

Definition — If VISIBLE is FALSE, a gadget or menu cannot be seen by the user. If VISIBLE is TRUE, a gadget or menu can be seen by the user provided that its ancestors' VISIBLE attributes are TRUE, it is not obscured by another window, and it is not positioned outside the visible area of its parent.

For menus displayed as menu bars and for menus displayed as submenus, the value of the VISIBLE elattribute is ignored. A menu displayed as a menu bar is visible when its parent window is visible. A menu, S, displayed as a submenu becomes visible automatically when a menu choice whose SUBMENU cattribute identifies menu S is activated. A menu, S, displayed as a submenu becomes invisible when (a) a select event occurs for a menu choice of S for which the SUBMENU gattribute is not specified or (b) a menu displayed as a submenu of S becomes invisible.

Domain — truefalse

Access — Create, Modify, Reference

Default — TRUE

6.5 Choice Attributes

Choice attributes, identified by cattributes, define the characteristics of choices. For gadgets, choice specifications have the form

^$W[INDOW] (wname , gkeyword , gname , ckeyword , item , cattribute)

ckeyword ::= expr V CHOICE

For menus, choice specifications have the form

^$W[INDOW] (wname , mkeyword , mname , ckeyword , item , cattribute)

cattribute ::= expr V

ACCELERATOR
ACTIVE
AID
EVENT
MARKER
SEPARATOR
SUBMENU
applicationattribute
implementationattribute

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.5.1 Applicability of Choice Attributes

The applicability of cattributes to element types is specified by Table 3. A "Yes" entry indicates that the use of the attribute with the element type is defined. An empty entry indicates that the use of the attribute with the element type is reserved.

Table 3: Choice Attributes

	List Box	List Button	List Entry Box	Long List Box	Menu	Radio Button Set
ACCELERATOR					Yes	
ACTIVE	Yes	Yes	Yes	Yes	Yes	Yes
AID	Yes	Yes	Yes	Yes	Yes	Yes
EVENT					Yes	
MARKER					Yes	
SEPARATOR					Yes	
SUBMENU					Yes	

6.5.2 ACCELERATOR cattribute

Definition ACCELERATOR specifies an accelerator key sequence for a choice in a menu. Accelerator keys provide a means of selecting menu items via a keyboard.

Domain keycode

Access Create, Reference

Default None

6.5.3 ACTIVE cattribute

Definition If ACTIVE is FALSE, the choice is disabled. The user cannot select or deselect the choice, and it may be displayed greyed or in some other fashion to indicate that it is disabled.

If ACTIVE is TRUE, the choice is enabled.

Domain truefalse

Access Create, Modify, Reference

Default TRUE

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.5.4 AID cattribute

Definition The AID cattribute is not used by the MWAPI. It is provided so that an application can associate an application-dependent value with a menu, list or radio button choice.

Domain chars

Access Create, Modify, Reference

Default None

6.5.5 EVENT cattribute

Definition EVENT specifies, via descendant nodes identified by etype subscripts, the choice's callback routine(s) for the event specified by etype. The format of a choice event specification is defined in Section 0. No value is present at the level of the EVENT cattribute itself.

Domain L doargument

Access Create, Modify, Reference

Default None

6.5.6 MARKER cattribute

Definition MARKER specifies a marker image that appears to the left of the choice text for a menu item.

Domain marker

Access Create, Modify, Reference

Default None

6.5.7 SEPARATOR cattribute

Definition In a pop-up menu, the SEPARATOR cattribute causes a visual separator (typically a single horizontal line) to appear below the choice text of the choice for which it is specified.

In a menu bar, the first instance of the SEPARATOR cattribute causes subsequent menu choices to appear right justified within the menu bar, if appropriate for the underlying windowing platform. Any instances of the SEPARATOR cattribute after the first within a menu bar are ignored.

The only value that can be assigned to the SEPARATOR cattribute is an emptystring.

Domain emptystring

Access Create, Reference

Default None

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.5.8 SUBMENU cattribute

Definition — SUBMENU identifies the mname of the pop-up menu to display if a menu choice is activated. If the mname assigned to the SUBMENU cattribute does not identify a menu defined for the same window or if the mname assigned to the SUBMENU is equal to the mname of the menu itself, an error condition occurs with $ECODE containing ",M?3," and $EREF containing a value that indicates the ssvn reference for which the error occurred. If the SUBMENU cattribute is not specified for a menu choice, no menu is displayed by the MWAPI if the user selects the menu choice.

Domain — mname

Access — Create, Modify, Reference

Default — None

7 Drawing

A drawing is specified by assigning drawcommand values to immediately descendant nodes of the DRAW attribute. These nodes have the form:

^$W[INDOW] (wname , gkeyword , gname , dkeyword , drawnum)

gname is the identifier for a gadget and wname is the identifier for the window that is the parent of that gadget.

dkeyword ::= expr V DRAW

drawnum ::= numexpr

7.1 Draw Command Numbers

drawcommands are stored as immediate descendants of the DRAW elattribute using drawnum subscripts. The drawing that appears is that which would result from sequentially performing all drawcommands for all drawnum subscripts, beginning with the lowest valued drawnum.

If an application attempts to assign a drawnum subscript which has a negative value, an error occurs with $ECODE containing ",M?2," and $EREF containing a value that indicates the ssvn reference for which the error occurred.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

7.2 Draw Commands

drawcommand specifies the draw commands that are available for a gadget if the gadget's DRAWTYPE elattribute has a value equal to "MDRAW".

drawcommand ::=
- drawarc
- drawbcolor
- drawbitmap
- drawbox
- drawfcolor
- drawfface
- drawfillpat
- drawfsize
- drawfstyle
- drawline
- drawmode
- drawoval
- drawpensize
- drawpicture
- drawpie
- drawpoint
- drawtext
- implementationdrawcommand

When used in a drawcommand, the unit of measure for the following position or size specifications

hpos
vpos
radius
hpos1
vpos1
hpos2
vpos2

is determined by the value of the gadget's UNITS elattribute. The unit of measure determines how size and position values are converted to pixels.

When used in a drawcommand, the unit of measure for the following angle specifications

startangle
endangle

is degrees with a 0 (zero) value representing a position on the positive horizontal axis. Degrees increase in a counterclockwise direction.

7.2.1 ARC Draw Command

drawarc ::= expr V ARC , hpos , vpos , radius , startangle , endangle

radius ::= numlit

startangle ::= numlit

endangle ::= numlit

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

The drawarc drawcommand draws a curved line using the current draw foreground color, draw mode, and pen size. The curved line is the portion of the circumference of a circle, with its center at position hpos,vpos and a radius of radius, that lies between the angles startangle and endangle.

7.2.2 BCOLOR Draw Command

drawbcolor ::= expr V BCOLOR , color , fillmode

The drawbcolor drawcommand specifies the current draw background color. If fillmode is equal to "OPAQUE", previous components of the drawing cannot be seen through the draw background color. If fillmode is equal to "TRANSPARENT", previous components of the drawing can be seen through the draw background color.

The specified color is used by subsequent drawcommands until another drawbcolor drawcommand is specified. Prior to the first drawbcolor drawcommand for a drawing, the current draw background color is determined by the element's BCOLOR elattribute.

7.2.3 BITMAP Draw Command

drawbitmap ::= expr V BITMAP , hpos , vpos , extresource

The drawbitmap drawcommand causes the bitmap image identified by extresource to be displayed with its origin at position hpos,vpos.

7.2.4 BOX Draw Command

drawbox ::= expr V BOX , hpos1 , vpos1 , hpos2 , vpos2

hpos1 ::= numlit

vpos1 ::= numlit

hpos2 ::= numlit

vpos2 ::= numlit

The drawbox drawcommand draws the perimeter of a rectangle using the current draw foreground color, draw mode, and pen size. One corner of the rectangle is at position hpos1,vpos1. The diagonally opposite corner of the rectangle is at position hpos2,vpos2. The interior of the rectangle is filled using the current draw background color and fill pattern.

7.2.5 DMODE Draw Command

drawmode ::= expr V DMODE , dmode

The drawing mode determines the effect of certain drawcommands on pixels in the drawing area. If the value of dmode is equal to "SOLID", drawcommands cause pixels to be changed to the current draw foreground color or draw background color. If the value of dmode is equal to "XOR", the new color of each pixel is determined by performing a Boolean exclusive or (XOR) between the pixel in the drawing area and the current draw foreground color or draw background color.

The drawmode drawcommand specifies the current drawing mode. The specified mode is used by subsequent drawcommands until another drawmode drawcommand is specified. Prior to the first drawmode drawcommand for a drawing, the current drawing mode is SOLID.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

7.2.6 DRAWTEXT Draw Command

drawtext ::= expr V DRAWTEXT , hpos , vpos , chars

The drawtext drawcommand displays the characters in chars using the current font face, font size, font style, and draw foreground color. The background of the character cells is filled using the current draw background color. The position hpos,vpos specifies the location of the upper left corner of the character cell for the first character in chars.

7.2.7 FCOLOR Draw Command

drawpencolor ::= expr V FCOLOR , color

The drawfcolor drawcommand specifies the current drawing foreground color. The specified color is used by subsequent drawcommands until another drawfcolor drawcommand is specified. Prior to the first drawfcolor drawcommand for a drawing, the current draw foreground color is determined by the element's FCOLOR elattribute.

7.2.8 FFACE Draw Command

drawfface ::= expr V FFACE , fontface

The drawfface drawcommand specifies the current font face. The specified font face is used by subsequent drawcommands until another drawfface drawcommand is specified. Prior to the first drawfface drawcommand for a drawing, the current font face is specified by the gadget's FFACE elattribute.

7.2.9 FILLPAT Draw Command

drawfillpattern ::= expr V FILLPAT , fillpattern

The drawfillpattern drawcommand specifies the current fill pattern, as follows:

fillpattern Value	Pattern
BDIAGONAL	Diagonal lines running from lower right to upper left.
CROSSHATCH	Vertical and horizontal lines.
DCROSSHATCH	Diagonal lines running from lower right to upper left and diagonal lines running from lower left to upper right.
FDIAGONAL	Diagonal lines running from lower left to upper right.
HORIZONTAL	Horizontal lines.
NONE	Filling is turned off.
SOLID	The area is filled with the current draw foreground color.
VERTICAL	Vertical lines.

If the value of drawfillpattern is an implementationvalue, an implementation-specific fill pattern is used.

The fill pattern is drawn using the current draw foreground color and draw mode, and fills the area. Regions of the area not drawn using the current drawing foreground color are drawn using the current background drawing color and draw mode.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

The specified pattern is used by subsequent drawcommands until another drawfillpattern drawcommand is specified. Prior to the first drawfillpattern drawcommand for a drawing, the current fill pattern is determined by the underlying windowing platform.

7.2.10 FSIZE Draw Command

drawfsize ::= expr V FSIZE , fontsize

The drawfsize drawcommand specifies the current font size. The specified font size is used by subsequent drawcommands until another drawfsize drawcommand is specified. Prior to the first drawfsize drawcommand for a drawing, the current font size is specified by the gadget's FSIZE elattribute.

7.2.11 FSTYLE Draw Command

drawfstyle ::= expr V FSTYLE , fontstyle

The drawfstyle drawcommand specifies the current font style. The specified font style is used by subsequent drawcommands until another drawfstyle drawcommand is specified. Prior to the first drawfstyle drawcommand for a drawing, the current font size is specified by the gadget's FSTYLE elattribute.

7.2.12 LINE Draw Command

drawline ::= expr V LINE , hpos1 , vpos1 , hpos2 , vpos2

The drawline drawcommand draws a line from the position hpos1,vpos1 to, but not including, the position hpos2,vpos2 using the current draw foreground color, draw mode, and pen size.

7.2.13 OVAL Draw Command

drawoval ::= expr V OVAL , hpos1 , vpos1 , hpos2 , vpos2

The drawoval drawcommand draws an ellipse using the current draw foreground color, draw mode, and pen size. The ellipse is drawn tangent to an invisible rectangle with one corner at hpos1,vpos1 and the diagonally opposite corner at hpos2,vpos2. The ellipse is filled using the current draw background color and fill pattern.

7.2.14 PENSIZE Draw Command

drawpensize ::= expr V PENSIZE , pensize

pensize ::= intlit

The drawpensize drawcommand specifies the current pen size. pensize indicates the diameter, measured in pixels, of a round pen. If pensize is equal to 0 (zero), no apparent drawing is done with the pen.

For the drawarc, drawpoint, and drawline drawcommands, the pen is centered on the line or point. For drawbox, drawoval, and drawpie drawcommands, the outer edge of the pen is tangent to the inner edge of the perimeter of the figure.

The specified size is used by subsequent drawcommands until another drawpensize drawcommand is specified. Prior to the first drawpensize drawcommand for a drawing, the current pen size is determined by the underlying windowing platform.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

7.2.15 PICTURE Draw Command

drawpicture ::= expr V PICTURE , hpos , vpos , extresource

The drawpicture drawcommand causes the picture identified by extresource to be displayed with its origin at position hpos,vpos.

7.2.16 PIE Draw Command

drawpie ::= expr V PIE , hpos , vpos , radius , startangle , endangle

The drawpie drawcommand draws a curved line and two straight lines using the current draw foreground color, draw mode, and pen size and fills the enclosed area using the current draw background color and fill pattern. The curved line is the portion of the circumference of a circle, with its center at position hpos,vpos and a radius of radius, that lies between the angles startangle and endangle. The straight lines connect the ends of the arc with the center of the circle.

7.2.17 POINT Draw Command

drawpoint ::= expr V POINT , hpos , vpos

The drawpoint drawcommand draws a point at the location hpos,vpos using the current draw foreground color, draw mode, and pen size.

8 Events

Events occur as a result of user actions or other happenings.

8.1 Event Types

The MWAPI defines the following event types, specified via etype values.

etype ::= expr V	CHANGE CLICK CLOSE DBLCLICK DESELECT FKEYDOWN FKEYUP FOCUS GOBOTTOM GODOWN GODOWNBIG GOTOP GOUP GOUPBIG

--`,,```,,,,````-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

HELP KEYDOWN KEYUP MAX MIN MOVE PDOWN PDRAG PMOVE PUP RESIZE RESTORE SELECT TIMER UNFOCUS implementationattribute

8.1.1 Window State Events

Window state events (i.e. close, minimize, restore, maximize, resize, and move events) occur when certain characteristics of a window change.

A close event, indicated by an etype value equal to "CLOSE", occurs for a window in response to a request to destroy the window.

A minimize event, indicated by an etype value equal to "MIN", occurs for a window when it changes from the open state to the iconic state.

A restore event, indicated by an etype value equal to "RESTORE", occurs for a window when it changes from the iconic state to the open state.

A maximize event, indicated by an etype value equal to "MAX", occurs for a window when it has been expanded to its maximum size.

A resize event, indicated by an etype value equal to "RESIZE", occurs for a window when its size changes.

A move event, indicated by an etype value equal to "MOVE", occurs for a window when its position changes.

8.1.2 Pointer Events

Pointer events (i.e. click, double click, pointer down, pointer up, pointer move, and pointer drag events) occur when the user takes certain actions with the pointer and the pointer is located within a window created via the MWAPI. If the pointer is within a gadget for which a pointer event can occur, then the event occurs for that gadget. Otherwise, if the pointer is within a window for which a pointer event can occur, then the pointer event occurs for that window.

A pointer down event, indicated by an etype value equal to "PDOWN", occurs when the user presses a pointer button.

A pointer up event, indicated by an etype value equal to "PUP", occurs when the user releases a pointer button.

A pointer move event, indicated by an etype value equal to "PMOVE", occurs when the user moves the pointer with no pointer buttons pressed.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

For a scroll gadget, a pointer drag event, indicated by an etype value equal to "PDRAG" occurs when the user drags the gadget's position indicator. For windows and other gadget types, a pointer drag event occurs when the user moves the pointer with at least one pointer button pressed. Pointer drag events occur for the window or gadget in which the pointer was located when the drag was initiated, even if the pointer is subsequently dragged outside of the window or gadget in which the drag began.

A click event, indicated by an etype value equal to "CLICK", occurs when the user presses and releases a pointer button without an intervening change in the pointer position. For a button click action by the user, the following sequence of events occurs: pointer down, pointer up, and click.

A double click event, indicated by an etype value equal to "DBLCLICK", occurs when (a) the user twice presses and releases a pointer button without any intervening changes in the pointer position and (b) the second button press occurs within a specified time interval of the first button press. The time interval is not specified by the MWAPI and may be controllable by the user. For a button double click action by the user, the following sequence of events occurs: pointer down, pointer up, click, pointer down, double click, and pointer up.
A pointer event does not occur if the FILTERIN especattribute is present for the event and the pbuttoncode associated with the event is not one of the pbuttoncodes present in the value of that FILTERIN especattribute.

8.1.3 Keyboard Events

Keyboard events (i.e. data key down, data key up, function key down, and function key up events) occur when the user presses or releases one or more keyboard keys.

If focus is on a gadget for which a keyboard event can occur, then the keyboard event occurs for that gadget. Otherwise, if focus is on a window for which a keyboard event can occur, then the keyboard event occurs for that window.

No behavior is defined for key sequences which are not mapped to key codes.

A data key down event, indicated by an etype value equal to "KEYDOWN", occurs when the user presses a key sequence for a data key code.

A data key up event, indicated by an etype value equal to "KEYUP", occurs when the user releases a key sequence for a data key code.

A function key down event, indicated by an etype value equal to "FKEYDOWN", occurs when the user presses a key sequence for a function key code.

A function key up event, indicated by an etype value equal to "FKEYUP", occurs when the user releases a key sequence for a function key code.

A keyboard event does not occur if (a) the key code represents an accelerator or has a special meaning for the underlying windowing platform or (b) the FILTERIN especattribute is present for the event and the keycode associated with the event is not one of the keycodes present in the value of that FILTERIN especattribute.

For gadgets that accept text input, function key codes are never included in the value of the gadget's VALUE elattribute. If data key up or data key down events are enabled for a gadget, then when a keyboard event occurs the gadget's VALUE elattribute is not modified by the underlying windowing platform or the MWAPI. The application can modify the VALUE elattribute, by assigning a value explicitly.

8.1.4 Focus Events

Focus events are focus, unfocus and change events.
A focus event, indicated by an etype value equal to "FOCUS", occurs when a window or gadget has received focus.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

For a window, an unfocus event, indicated by an etype value equal to "UNFOCUS", occurs when a window is about to lose focus to another window defined in ^$WINDOW. If a window W_1 which is defined in ^$WINDOW has focus, then focus changes to one or more windows not defined in ^$WINDOW, and then focus changes to a window W_2 which is defined in ^$WINDOW, an unfocus event occurs for W_1 if W_1 is a different window than W_2 and no unfocus event occurs for W_1 if W_1 is a the same window as W_2.

For a gadget, an unfocus event occurs when the gadget is about to lose focus to another gadget defined for the same window. If a gadget G in window W has focus, then focus changes to one or more other windows, and then focus changes to window W, an unfocus event occurs for gadget G when focus changes to window W but not to gadget G.

A change event, indicated by an etype value equal to "CHANGE", occurs for a gadget instead of an unfocus event if (a) change events are enabled for the gadget and (b) the value of the gadget's CHANGED elattribute is equal to TRUE. If both change and unfocus events are enabled for a gadget and a change event occurs, then an unfocus event does not occur.

If, during call back processing for a change or unfocus event, an application removes the OK einfoattribute from ^$EVENT, the request to change focus is denied. If the action which produced the change or unfocus event also would produce a subsequent focus event, that focus event does not occur. In addition, any deferred events except timer events are discarded and, to the extent possible, visual and other effects of the attempt to change focus are prevented or reversed.

If an action yields both an unfocus or change event and a focus event, the unfocus or change event precedes the focus event.

8.1.5 Select Events

Select events are select and unselect events.

A select event, indicated by an etype value equal to "SELECT" occurs when a user selects a gadget or choice.

For a check box, a select event occurs when the check box is selected.

For a list box, list button, or long list box gadget, a select event occurs for each choice that is selected. Multiple select events may occur as a result of a single user action.

For a menu item, a select event occurs when a choice is selected.

For a push button, a select event occurs when the push button is pushed.

For a radio button set, a select event occurs when a choice is selected. If a choice was previously selected, an unselect event precedes it indicating that a choice was implicitly deselected.

For a scroll gadget, a select event occurs when user action causes the scroll gadget's VALUE elattribute to change.

An unselect event, indicated by an etype value equal to "DESELECT" occurs when a user unselects a gadget or choice.

For a check box, an unselect event occurs when the check box is unselected.

For a list box, list button, or long list box gadget, an unselect event occurs for each choice that is unselected. Multiple unselect events may occur as a result of a single user action.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

8.1.6 Long List Box Events

Long list box events (i.e. go bottom, go down, go down big, go top, go up, go up big events) occur when the user selects a movement control in a long list box gadget. If a long list box event is not enabled when a long list box gadget is created, the corresponding movement control may be absent from the long list box and it may not be possible to add it subsequently.

A go bottom event, indicated by an etype value equal to "GOBOTTOM", occurs for a gadget when the user selects the gadget's go bottom movement control.

A go down event, indicated by an etype value equal to "GODOWN", occurs for a gadget when the user selects the gadget's go down movement control.

A go down big event, indicated by an etype value equal to "GODOWNBIG", occurs for a gadget when the user selects the gadget's go down big movement control.

A go top event, indicated by an etype value equal to "GOTOP", occurs for a gadget when the user selects the gadget's go top movement control.

A go up event, indicated by an etype value equal to "GOUP", occurs for a gadget when the user selects the gadget's go up movement control.

A go up big event, indicated by an etype value equal to "GOUPBIG", occurs for a gadget when the user selects the gadget's go up big movement control.

8.1.7 Help Events

A help event, indicated by an etype value equal to "HELP", occurs when a window created via the MWAPI has focus and the user requests help by pressing a designated key or through some other means specified by the underlying windowing platform or by the MWAPI implementation. If a gadget for which help events are enabled has focus, then a help event occurs for that gadget. Otherwise, a help event occurs for the gadget's parent window.

8.1.8 Timer Events

A timer event, indicated by an etype value equal to "TIMER", occurs when an interval timer expires.

8.1.9 Applicability of Event Types

The applicability of etypes to window, element, and choice types is specified by Table 4. A "Yes" entry indicates that the occurrence of the event type with the window, element, or choice type is defined. An empty entry indicates that the occurrence of the event type with the window, element, or choice type is reserved.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Table 4: Events

	Application Window	M Term Window	Check Box	Document	Generic Box	Group Frame
CHANGE			Yes	Yes		
CLICK	Yes				Yes	
CLOSE	Yes					
DBLCLICK	Yes				Yes	
DESELECT			Yes			
FKEYDOWN	Yes			Yes	Yes	
FKEYUP	Yes			Yes	Yes	
FOCUS	Yes		Yes	Yes	Yes	
GOBOTTOM						
GODOWN						
GODOWNBIG						
GOTOP						
GOUP						
GOUPBIG						
HELP	Yes		Yes	Yes	Yes	
KEYDOWN	Yes			Yes	Yes	
KEYUP	Yes			Yes	Yes	
MAX	Yes					
MIN	Yes					
MOVE	Yes					
PDOWN	Yes				Yes	
PDRAG	Yes				Yes	
PMOVE	Yes				Yes	
PUP	Yes				Yes	
RESIZE	Yes					
RESTORE	Yes					
SELECT			Yes			
TIMER						
UNFOCUS	Yes		Yes	Yes	Yes	

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Table 4: Events (Continued)

	Label	List Box	List Button	List Entry Box	Long List Box	Menu
CHANGE		Yes	Yes	Yes	Yes	
CLICK						
CLOSE						
DBLCLICK		Yes			Yes	
DESELECT		Yes	Yes		Yes	
FKEYDOWN				Yes		
FKEYUP				Yes		
FOCUS		Yes	Yes	Yes	Yes	
GOBOTTOM					Yes	
GODOWN					Yes	
GODOWNBIG					Yes	
GOTOP					Yes	
GOUP					Yes	
GOUPBIG					Yes	
HELP		Yes	Yes	Yes	Yes	
KEYDOWN				Yes		
KEYUP				Yes		
MAX						
MIN						
MOVE						
PDOWN						
PDRAG						
PMOVE						
PUP						
RESIZE						
RESTORE						
SELECT		Yes	Yes		Yes	
TIMER						
UNFOCUS		Yes	Yes	Yes	Yes	

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Table 4: Events (Continued)

	Menu Choice	Push Button	Radio Button Set	Scroll Bar	Symbol	Text	Timer
CHANGE			Yes			Yes	
CLICK							
CLOSE							
DBLCLICK							
DESELECT			Yes				
FKEYDOWN						Yes	
FKEYUP						Yes	
FOCUS		Yes	Yes			Yes	
GOBOTTOM							
GODOWN							
GODOWNBIG							
GOTOP							
GOUP							
GOUPBIG							
HELP		Yes	Yes	Yes		Yes	
KEYDOWN						Yes	
KEYUP						Yes	
MAX							
MIN							
MOVE							
PDOWN							
PDRAG				Yes			
PMOVE							
PUP							
RESIZE							
RESTORE							
SELECT	Yes	Yes	Yes	Yes			
TIMER							Yes
UNFOCUS		Yes	Yes			Yes	

--`,,```,,,,````-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

8.2 Event Specification Attributes

Event specifications are defined by especs which consist of lists of subscripts.

espec ::= L expr

For windows, an espec has the form

wname , ekeyword , etype

where

ekeyword ::= expr V EVENT

For gadgets, an espec has the form

wname , gkeyword , gname , ekeyword , etype

For timers, an espec has the form

wname , tkeyword , tname , ekeyword , etype

For menu choices, an espec has the form

wname , mkeyword , mname , ckeyword , item , ekeyword , etype

All other forms of espec are reserved.

Event specification attributes, identified by especattributes, determine when events and call backs occur.

especattribute ::= expr V		ENABLE FILTERIN

Event specification attributes are specified as:

^$W[INDOW] (espec , especattribute)

8.2.1 ENABLE especattribute

Definition An event identified by etype is enabled for a window, gadget, timer, or menu choice if, when the event occurs, there is an espec defined for the window, gadget, timer or menu choice which has a valid doargument for its value and the corresponding ENABLE especattribute for the espec has a TRUE value. If the ENABLE especattribute is FALSE, the event is not enabled for the window, gadget, timer or menu choice.

Domain truefalse

Access Create, Modify, Reference

Default TRUE

8.2.2 FILTERIN especattribute

Definition FILTERIN specifies, for certain events, additional criteria that must be TRUE for an event to occur.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Domain For a keyboard event:

L keycode

For a pointer event:

L pbuttoncode

Access Create, Modify, Reference

Default None

8.3 Processing Events

If event processing is activated, then for each enabled event of type E that occurs, call back processing occurs by execution of an implicit DO command of the form

DO @doargument

where doargument is the value of the EVENT wattribute, elattribute, or cattribute for a window, element, or choice, respectively and for etype equal to E.

For each ESTART command, call back processing is carried out for events in the order of their occurrence. If an enabled event occurs while call back processing is not activated, call back processing for it may be deferred until call back processing is activated. While there is no explicit limit on the number of deferred events, deferred events may be lost if not processed quickly.

Call back processing does not occur as a result of modification of ^$DISPLAY or ^$WINDOW by the application.

For close, data key up, and data key down events, if the event is enabled, the normal actions of the underlying windowing platform for that event are inhibited; if the event is not enabled, the normal actions of the underlying windowing platform for that event occur. For other events, the normal actions of the underlying windowing platform occur for the event, whether or not the event is enabled.

8.4 Event Commands

8.4.1 ESTART

ESTA [RT] postcond | [SP] / SP timeout |

The M command ESTART activates event processing. Event processing remains activated until the termination of execution of the ESTART command, except that event processing is implicitly deactivated at the initiation of call back processing for each event. At the conclusion of call back processing for each event, event processing is implicitly reactivated and all einfoattribute nodes in ^$EVENT become undefined.

During call back processing for an event E, event processing can be explicitly activated by executing a nested ESTART command. This causes call back processing for event E to be suspended, and causes the values of all einfoattribute nodes in ^$EVENT to be stacked and to become undefined. When execution of this nested ESTART command is terminated, the values of the einfoattribute nodes that it stacked are restored to ^$EVENT.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

When a nested ESTART command is executed, the value of the FOCUS dattribute for the display on which the window associated with event E appears must identify a window W which has a MODAL wattribute value equal to "APPLICATION". If it does not identify such a window, an error occurs with $ECODE containing ",M?10,". If it does identify such a window, the nested ESTART command activates event processing for (a) any enabled events for window W and (b) any enabled timer events for all other windows; processing of all other enabled events is deferred.

The execution of an ESTART command is terminated if (a) an ESTOP command is executed during call back processing for that ESTART command or (b) if timeout is present, the timeout expires. If the timeout period expires during callback processing for an event, the ESTART command is terminated at the conclusion of processing for that event.

When execution of an ESTART command is terminated, execution continues with the command following that ESTART command.

If timeout is present, $TEST indicates the condition that caused event processing to terminate. If event processing was terminated with an ESTOP command, the value of $TEST is 1. If event processing was not terminated with an ESTOP command, the value of $TEST is 0. If timeout is not present, execution of ESTART does not change $TEST.

8.4.2 ESTOP

ESTO [P] postcond [SP]

The M command ESTOP implicitly performs the number of M QUIT commands necessary to return to the execution level of the most recently executed ESTART command and then terminates that ESTART command.

8.4.3 ETRIGGER

ETR [IGGER] postcond SP especref [: einforef]

The M command ETRIGGER causes an event to occur. If this event is enabled, call back processing for it will occur subsequently.

especref ::= expr V ^$W [INDOW] (espec)

einforef ::= | glvn
@ expratom V einforef |

The event occurs for the window, element, or menu choice specified by the especref. If the window, element, or menu choice specified by the especref does not exist, an error condition occurs with $ECODE containing ",M?3," and $EREF containing a value that indicates the non-existent window, element, or menu choice.

The event that occurs is specified by the etype component of the especref. If the especref does not contain a valid etype, an error condition occurs with $ECODE containing ",M?2,".

No additional events occur as consequence of an event that occurs as a result of execution of an ETRIGGER command. For example, executing an ETRIGGER command that causes a focus event to occur does not automatically cause an unfocus or change event to occur.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

When call back processing occurs for the event, the following einfoattribute nodes are provided automatically in ^$EVENT:

For all events, CLASS, SEQUENCE, TYPE and WINDOW.

When an event occurs for an element, ELEMENT.

When an event occurs for a menu choice, CHOICE and ELEMENT.

For change and unfocus events, OK.

Other einfoattribute nodes can be provided by the application via the einforef argument. If einforef is present, then for each immediately descendant subscript of einforef which matches an einfoattribute that (a) is defined for the event type specified in the especref and (b) is not provided automatically in ^$EVENT, as described above, the value of the ^$EVENT einfoattribute is set to the value of the corresponding einforef node.

8.5 Event Information

During call back processing for an event, the ssvn ^$EVENT contains event information attributes, identified by einfoattributes, in the form

^$E[VENT] (einfoattribute)

which provide information about the event.

einfoattribute ::= expr V

CHOICE
CLASS
ELEMENT
KEY
NEXTFOCUS
OK
PBUTTON
PPOS
PRIORFOCUS
PSTATE
SEQUENCE
TYPE
WINDOW

8.5.1 CHOICE einfoattribute

Definition CHOICE identifies the choice associated with a select event for an element.

Domain item

Access Reference

Default Not Applicable.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

8.5.2 CLASS einfoattribute

Definition CLASS specifies the event class to which the event belongs. For MWAPI events, CLASS is equal to "WINDOW".

Domain chars

Access Reference

Default Not Applicable.

8.5.3 ELEMENT einfoattribute

Definition If the event is associated with an element, ELEMENT specifies that element's type and ename. If the event is not associated with an element, ELEMENT is not defined.

Domain For a gadget

G,ename

For a menu

M,ename

For a timer

T,ename

Access Reference

Default Not Applicable.

8.5.4 KEY einfoattribute

Definition KEY identifies the key sequence associated with a keyboard event.

For a pointer event, KEY identifies all keys that were pressed when the event occurred. If no keys were pressed, KEY is equal to emptystring.

Domain For a keyboard event, keycode.

For a pointer event, L keycode.

Access Reference

Default Not Applicable.

8.5.5 NEXTFOCUS einfoattribute

Definition During call back processing for an unfocus or change event, NEXTFOCUS identifies the window or gadget for which a related focus event will occur. If no related focus event will occur for a window or gadget that is defined in ^$WINDOW, NEXTFOCUS is equal to emptystring.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Domain focusloc

Access Reference

Default Not Applicable.

8.5.6 PRIORFOCUS einfoattribute

Definition During call back processing for a focus event, PRIORFOCUS identifies the window or gadget for which a related unfocus or change event occurred. If no related unfocus or change event occurred for a window or gadget that is defined in ^$WINDOW, PRIORFOCUS is equal to emptystring.

Domain focusloc

Access Reference

Default Not Applicable.

8.5.7 OK einfoattribute

Definition OK specifies whether the event should be completed normally. If OK is present, processing for the event completes normally. If OK is not defined, processing for the event is terminated in an event-specific fashion.

An application cannot assign a value to the OK einfoattribute but it can remove the OK einfoattribute using the M KILL command.

Domain emptystring

Access Modify, Reference

Default emptystring

8.5.8 PBUTTON einfoattribute

Definition PBUTTON identifies the pointer button associated with a pointer event.

Domain pbuttoncode

Access Reference

Default Not Applicable.

8.5.9 PPOS einfoattribute

Definition PPOS identifies the pointer position. If the event is associated with a gadget, the position is measured with respect to the gadget's origin and is expressed in the unit of measure specified by the gadget's UNITS elattribute. Otherwise, the event is associated with a window and the position is measured with respect to the origin of the application area and is expressed in the unit of measure specified by the window's UNITS wattribute.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Domain pos

Access Reference

Default Not Applicable.

8.5.10 PSTATE einfoattribute

Definition PSTATE identifies any pointer buttons that were pressed when a pointer event occurred. If no buttons were pressed, PSTATE is equal to emptystring.

Domain L pbuttoncode

Access Reference

Default Not Applicable.

8.5.11 SEQUENCE einfoattribute

Definition SEQUENCE uniquely identifies each event. The value of SEQUENCE is the number of events for which call back processing has been initiated for a process.

Domain intlit

Access Reference

Default Not Applicable.

8.5.12 TYPE einfoattribute

Definition TYPE identifies the type of event.

Domain etype

Access Reference

Default Not Applicable.

8.5.13 WINDOW einfoattribute

Definition If the event is associated with a window, WINDOW specifies that window's wname. If the event is associated with an element, WINDOW specifies the wname of the element's parent. If the event is not associated with a window or element, WINDOW is not defined.

Domain wname

Access Reference

Default Not Applicable.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

8.5.14 Applicability of Event Information Attributes

The applicability of einfoattribute nodes to etypes is specified by Table 5. A "Yes" entry indicates that the einfoattribute is defined for the etype. An empty entry indicates that the einfoattribute is reserved for the etype.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Table 5: Event Information

	CLASS	TYPE	WINDOW	SEQUENCE	ELEMENT	CHOICE	KEY
CHANGE	Yes	Yes	Yes	Yes	Yes		
CLICK	Yes	Yes	Yes	Yes	Yes*		Yes
CLOSE	Yes	Yes	Yes	Yes			
DBLCLICK	Yes	Yes	Yes	Yes	Yes*		Yes
DESELECT	Yes	Yes	Yes	Yes	Yes	Yes	
FKEYDOWN	Yes	Yes	Yes	Yes	Yes*		Yes
FKEYUP	Yes	Yes	Yes	Yes	Yes*		Yes
FOCUS	Yes	Yes	Yes	Yes	Yes*		
GOBOTTOM	Yes	Yes	Yes	Yes	Yes		
GODOWN	Yes	Yes	Yes	Yes	Yes		
GODOWNBIG	Yes	Yes	Yes	Yes	Yes		
GOTOP	Yes	Yes	Yes	Yes	Yes		
GOUP	Yes	Yes	Yes	Yes	Yes		
GOUPBIG	Yes	Yes	Yes	Yes	Yes		
HELP	Yes	Yes	Yes	Yes	Yes*		
KEYDOWN	Yes	Yes	Yes	Yes	Yes*		Yes
KEYUP	Yes	Yes	Yes	Yes	Yes*		Yes
MAX	Yes	Yes	Yes	Yes			
MIN	Yes	Yes	Yes	Yes			
MOVE	Yes	Yes	Yes	Yes			
PDOWN	Yes	Yes	Yes	Yes	Yes*		Yes
PDRAG	Yes	Yes	Yes	Yes	Yes*		Yes
PMOVE	Yes	Yes	Yes	Yes	Yes*		Yes
PUP	Yes	Yes	Yes	Yes	Yes*		Yes
RESIZE	Yes	Yes	Yes	Yes			
RESTORE	Yes	Yes	Yes	Yes			
SELECT	Yes	Yes	Yes	Yes	Yes	Yes	
TIMER	Yes	Yes	Yes	Yes	Yes		
UNFOCUS	Yes	Yes	Yes	Yes	Yes*		

* The ELEMENT einfoattribute is defined if the event occurs for an element and is undefined otherwise.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Table 5: Event Information (Continued)						
	NEXT-FOCUS	OK	PBUTTON	PPOS	PRIOR-FOCUS	PSTATE
CHANGE	Yes	Yes				
CLICK			Yes	Yes		Yes
CLOSE						
DBLCLICK			Yes	Yes		Yes
DESELECT						
FKEYDOWN						
FKEYUP						
FOCUS					Yes	
GOBOTTOM						
GODOWN						
GODOWNBIG						
GOTOP						
GOUP						
GOUPBIG						
HELP						
KEYDOWN						
KEYUP						
MAX						
MIN						
MOVE						
PDOWN			Yes	Yes		Yes
PDRAG			Yes	Yes		Yes
PMOVE			Yes	Yes		Yes
PUP			Yes	Yes		Yes
RESIZE						
RESTORE						
SELECT						
TIMER						
UNFOCUS	Yes	Yes				

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

9 Additional Functions and Special Variables

9.1 Functions

When the MWAPI is available, the following functions are added to M:

$WFONT
$WTFIT
$WTWIDTH

9.1.1 $WFONT

$WFONT (fontface , fontsize , fontstyle , unitspec)

$WFONT returns information about the font identified by fontface, fontsize, and fontstyle. The value returned has the form

fontheight , fontmaxchar , fontavgchar

where

fontheight ::= numlit

fontmaxchar ::= numlit

fontavgchar ::= numlit

fontheight is the height of the character cell plus external leading for the font. fontmaxchar is the maximum width of a single character for the font. fontavgchar is an average width of a single character for the font.

If unitspec is equal to "PIXEL", fontheight, fontmaxchar, and fontavgchar are measured in pixels. If unitspec is equal to "POINT", fontheight, fontmaxchar, and fontavgchar are measured in points. If any other value is specified for unitspec, an error condition occurs with $ECODE containing ",M?8,".

If information is not available for the specified font, the return value is equal to the emptystring.

9.1.2 $WTFIT

$WTFIT (expr , numexpr , fontface , fontsize , fontstyle , unitspec)

$WTFIT returns a value equal to the number of characters in the text string expr that could be displayed fully, using an indicated font, in an area whose width is numexpr in the unit of measure specified by unitspec. wide. The indicated font is specified by fontface, fontsize and fontstyle.

For a string expr containing c characters, $WTFIT returns the following values:

If the entire string expr can be displayed in an area numexpr units wide, $WTFIT returns c.

If the first character in the string expr is too wide to be displayed fully in an area numexpr units wide, $WTFIT returns 0 (zero).

If the first m characters in the string expr can be displayed fully in an area numexpr units wide, but the first m+1 characters are too wide to be displayed in an area numexpr units wide, $WTFIT returns m.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

If information is not available for the specified font, the return value is equal to the emptystring.

If unitspec is not equal to "PIXEL" or "POINT", an error condition occurs with $ECODE containing ",M?8,".

9.1.3 $WTWIDTH

$WTWIDTH (expr , fontface , fontsize , fontstyle , unitspec)

$WTWIDTH returns a value equal to the width of the text string expr displayed using the font indicated by fontface, fontsize and fontstyle. The width is expressed in the unit of measure indicated by unitspec.

If information is not available for the specified font, the return value is equal to the emptystring.

If unitspec is not equal to "PIXEL" or "POINT", an error condition occurs with $ECODE containing ",M?8,".

9.2 Special Variables

The following svn is added to M:

$PD [ISPLAY]

The intrinsic special variable $PDISPLAY indicates the principal display of an M process. The value of $PDISPLAY is a display.

10 Error Handling

Errors that occur during use of the MWAPI are handled in accordance with the error processing mechanisms specified for M.

If one or more MWAPI errors occurs, at least one error is reported. If one or more errors occurs during the execution of an M MERGE command that references ^$DISPLAY or ^$WINDOW as a glvn on the left hand side of the equal sign, the resulting state of ^$DISPLAY or ^$WINDOW is not defined.

10.1 Error Information

For certain MWAPI errors, an additional svn, $EREF, identifies the ssvn reference associated with the error.

$ER[EF]

The value provided by $EREF is a namevalue.

If the error involves a display attribute, $EREF has the form:

^$DISPLAY (display , dattribute)

If the error involves a window attribute, $EREF has the form:

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

^$WINDOW (wname , wattribute)

If the error involves a gadget attribute, $EREF has the form:

^$WINDOW (wname , gkeyword , gname , elattribute)

If the error involves a menu attribute, $EREF has the form:

^$WINDOW (wname , mkeyword , mname , elattribute)

If the error involves a timer attribute, $EREF has the form:

^$WINDOW (wname , tkeyword , tname , elattribute)

If the error involves a choice attribute for a gadget, $EREF has the form:

^$WINDOW (wname , gkeyword , gname , ckeyword , item , cattribute)

If the error involves a choice attribute for a menu, $EREF has the form:

^$WINDOW (wname , mkeyword , mname , ckeyword , item , cattribute)

10.2 Errors

M46	Invalid attribute name.
M47	Invalid attribute value.
M48	Nonexistent window, element or choice.
M49	Invalid attempt to set focus.
M50	Attempt to reference a non-MTERM window in an OPEN command.
M51	Attempt to destroy MTERM window prior to CLOSE.
M52	Required attribute missing.
M53	Invalid argument for font function.
M54	Attempt to create non-modal child of modal parent.
M55	Invalid nested ESTART.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

11 Conformance

11.1 Implementations

11.1.1 Conforming Implementation

A conforming implementation of the MWAPI shall:

a. Correctly implement all features conforming to this International Standard.

b. Detect all errors required by this International Standard.

c. Complete the following conformance specification:

XXX version V implements X11.6-YYYY with the following conformance specification:

Implementation Definition		
Section	Feature	Option/Value
0	PLATFORM dattribute	
	Minimum hardware configuration	
	Operating system name	
	Operating system version	
	Required software	
	Special configuration requirements	

d. Use the implementationattribute and implementationvalue name spaces for implementation-specific features.

e. Define all implementation-specific features in the following table.

Implementation-Specific Features		
Keyword	Window/Element/ Event/Attribute	Description

f. Provide, at a minimum, the limits defined in the following table.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Portability Limits			
Section	Feature	Description	Limit
0	application-attribute	Aggregate size of applicationattributes for a process, calculated according to the M local variable storage specification.	M local variable storage limit
0	wname	Number of characters in window name	31
0		Number of windows	31
0		Number of elements per M process	2,047
0	gname	Number of characters in gadget name	31
0		Number of gadgets per window	255
0	mname	Number of characters in menu name	31
0	tname	Number of characters in timer name	31
0		Number of timers per window	7
0	CHARMAX elattribute	Maximum number of characters in the text value of a list entry box or text gadget	M string length
0	item	Total number of menu choices per window	255
0	CHOICE elattribute	Number of characters in a single choice	255
0	item	Number of choices in a list box, list entry box, or long list box.	1023
0	item	Number of choices in a list button	31
0	CHOICE elattribute	Aggregate number of characters in all choices for a gadget	16,383
0	TITLE elattribute	Maximum number of characters in an element title	255
0	longchars	Maximum number of characters for the text value of a document gadget	32,767
0	SUBMENU cattribute	Menu depth	7
0	drawnum	Aggregate number of drawcommands for a process	2,047

g. Define all features that are not implemented in the following table.

Unimplemented Features		
Section	Feature	Description

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

A conforming implementation may, but is not required to, permit an application to modify the values of attributes for which this International Standard does not specify Modify access.

11.1.2 MDC Conforming Implementation

An MDC conforming implementation shall be a conforming implementation except that the conforming document shall be this International Standard together with any such current MDC documents that the vendor chooses to implement. The unimplemented features table must be empty in order to have a conforming implementation. The conformance statement shall be in the form:

XXX version V implements X11.6-YYYY with the following conformance specification:
:
:
as modified by the following MDC documents ddd (MDC status sss)

11.1.3 MDC Strictly Conforming Implementation

An MDC strictly conforming implementation is an MDC conforming implementation that implements only MDC documents with MDC Type A status. An MDC strictly conforming implementation may not permit an application to modify the values of attributes for which this International Standard does not specify Modify access.

11.1.4 MDC Partial Implementation

An MDC partial implementation is an implementation that:

a. Provides a subset of the functionality required by this International Standard.

b. Conforms in all other respects to this International Standard.

c. Documents all areas of non-compliance with this International Standard in the unimplemented features table.

11.1.5 Multiple Levels of Conformance

An implementation may claim more than one of the previously defined levels of conformance if it provides a mechanism by which the user is able to select a conformance level.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Index

^$DISPLAY 2, 4
^$EVENT 2, 4
^$WINDOW 2, 4
ACCELERATOR 52, 53
Access 5
ACTIVE 20, 22, 23, 27, 34, 36, 37, 38, 42, 52, 53
Ampersand character 6
Ancestor 2
ANCESTORS 14, 26
Application 1, 14, 19, 26, 30
ARC 56
Assignment of attribute value 4
Attribute 4
BOX 57
BUTTON 31, 33
CANCEL 34, 36, 37, 38
CHANGE 60, 63
CHANGED 34, 36, 37, 38, 63
Characters 6
CHECK 31
Child 2
Choice 1, 34, 36, 37, 39, 52, 71, 82
Choice layout 14, 15
CLIPBOARD 15, 16
CLOSE 60, 61, 65, 66, 67, 76, 77
color 6, 16, 23, 24, 38, 40, 48, 57, 58
Color 6
Create 5
Creation 4
Default 5
Default value 4
Definition 5
Descendant 2
Destruction of window, element or choice 5
Display 2, 11, 79
DISPLAY 20, 22
DOCUMENT 31, 32
Domain 5
Drawing 2
Element 1, 71, 72
Empty string 6
ENABLE 68
Event 2
EVENT 21, 22, 24, 34, 36, 37, 40, 52, 53, 54, 68, 69
FALSE 7
FOCUS 15, 17, 60, 62, 65, 66, 67
Font 8
Font face 8
Font size 8
Font style 9
Gadget 1
Graphical user interface 1
GUI 1
HELP 60, 64, 65, 66, 67, 76, 77
HORIZONTAL 12, 58
ICON 20, 22, 25
ICONIFY 20, 22, 25
Identifier 9
insert 13, 41
Insertion point 13
INTERVAL 34, 36, 37, 42
item 5, 10, 39, 49, 51, 52, 68, 71, 80, 82
KEY 71, 72
Key code 13
Key sequence 13
Keyboard 2
KEYBOARD 15
Label 32, 41, 47, 49
LABEL 31, 32
LEFT 13
LINE 59
LIST 31, 32
Look and feel 1
Menu 1
Menu choice marker 8
Metalanguage 3
MODAL 14, 20, 22, 26, 70
Modify 5
Mouse 2
name 3
Origin 11
Parent 2
PARENT 14, 20, 22, 26, 27
Parent menu 2
PEN 15, 17
POINT 10, 60, 78, 79
Pointer 2
Pointer button 13
Pointer button code 13
pos 10, 11, 27, 43, 74
POS 20, 22, 27, 28, 34, 36, 37, 43
Position 10
RADIO 31, 33
Reference 5
Reserved attribute name 4
RESIZE 20, 22, 28, 60, 61, 65, 66, 67, 76, 77
RESOURCE 34, 36, 37, 43
RESTORE 60, 61, 65, 66, 67, 76, 77
RIGHT 13
ROW 14, 44
rowcol 14, 44
ROWCOL 34, 36, 37, 44
SCROLL 20, 22, 28, 31, 33, 34, 36, 37, 44
Scroll gadget 14, 33, 45, 46, 47, 51
scrollby 14, 45
SCROLLBY 34, 36, 37, 45
scrollrange 14, 46
SCROLLRANGE 34, 36, 37, 45, 46, 52
Selection range 13

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

SEQUENCE 71, 74
size 11, 18, 28, 29, 45, 47
SIZE 15, 18, 20, 22, 28, 29, 34, 36, 37, 44, 47
SOLID 12, 57, 58
Submenu 2
SUBMENU 52, 53, 55, 82
symbol 8, 43
SYMBOL 31, 33
Symbol gadget 33, 43, 47
TEXT 31, 33
Text insertion point 13
Text selection range 13
TIED 20, 22, 27, 28, 29
Timer 1
TIMER 60, 64, 65, 66, 67, 76, 77
TITLE 20, 22, 25, 30, 34, 36, 37, 49, 82
Title position 13
TOP 13, 50
TRANSPARENT 12, 57
TRUE 7
TYPE 20, 22, 23, 30, 31, 32, 33, 34, 36, 37, 50, 71, 74
Type font 8
Underlying windowing platform 1, 5
UNITS 10, 11, 15, 18, 19, 20, 22, 27, 28, 29, 30, 34, 36, 37, 43, 47, 50, 56, 73
User 1
VALUE 14, 34, 36, 37, 38, 42, 45, 46, 50, 51, 62, 63
VERTICAL 12, 58
VISIBLE 20, 22, 27, 30
Window 1
WINDOW 71, 72, 74

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

© ISO/IEC

ICS 35.060

Price based on 85 pages

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale