

Application Kit Functions

Rectangle Functions

Modify a rectangle

```
void          NXSetRect(NXRect *aRect, NXCoord x, NXCoord y, NXCoord width, NXCoord
                height)
void          NXOffsetRect(NXRect *aRect, NXCoord dx, NXCoord dy)
void          NXInsetRect(NXRect *aRect, NXCoord dx, NXCoord dy)
void          NXIntegralRect(NXRect *aRect)
NXRect *      NXDivideRect(NXRect *aRect, NXRect *bRect, NXCoord slice, int edge)
```

Test graphic relationships

```
BOOL          NXMouseInRect(const NXPoint *aPoint, const NXRect *aRect, BOOL flipped)
BOOL          NXPointInRect(const NXPoint *aPoint, const NXRect *aRect)
BOOL          NXIntersectsRect(const NXRect *aRect, const NXRect *bRect)
BOOL          NXContainsRect(const NXRect *aRect, const NXRect *bRect)
BOOL          NXEqualRect(const NXRect *aRect, const NXRect *bRect)
BOOL          NXEmptyRect(const NXRect *aRect)
```

Compute third rectangle from two rectangles

```
NXRect *      NXUnionRect(const NXRect *aRect, NXRect *bRect)
NXRect *      NXIntersectionRect(const NXRect *aRect, NXRect *bRect)
```

Optimize drawing

```
void          NXRectClip(const NXRect *aRect)
void          NXRectClipList(const NXRect *rects, int count)
void          NXRectFill(const NXRect *aRect)
void          NXRectFillList(const NXRect *rects, int count)
void          NXRectFillListWithGrays(const NXRect *rects, const float *grays, int count)
void          NXEraseRect(const NXRect *aRect)
void          NXHighlightRect(const NXRect *aRect)
```

Draw a bordered rectangle

```
void          NXDrawButton(const NXRect *aRect, const NXRect *clipRect)
void          NXDrawGrayBezel(const NXRect *aRect, const NXRect *clipRect)
void          NXDrawGroove(const NXRect *aRect, const NXRect *clipRect)
void          NXDrawWhiteBezel(const NXRect *aRect, const NXRect *clipRect)
NXRect *      NXDrawTiledRects(NXRect *boundsRect, const NXRect *clipRect, const int
                *sides, const float *grays, int count)

void          NXFrameRect(const NXRect *aRect)
void          NXFrameRectWithWidth(const NXRect *aRect, NXCoord frameWidth)
```

Query an NXRect structure

NXCoord	NX_X (NXRect <i>*aRect</i>)
NXCoord	NX_Y (NXRect <i>*aRect</i>)
NXCoord	NX_WIDTH (NXRect <i>*aRect</i>)
NXCoord	NX_HEIGHT (NXRect <i>*aRect</i>)
NXCoord	NX_MAXX (NXRect <i>*aRect</i>)
NXCoord	NX_MAXY (NXRect <i>*aRect</i>)
NXCoord	NX_MIDX (NXRect <i>*aRect</i>)
NXCoord	NX_MIDY (NXRect <i>*aRect</i>)

Color Functions

Specify a color value

NXColor	NXConvertRGBAToColor (float <i>red</i> , float <i>green</i> , float <i>blue</i> , float <i>alpha</i>)
NXColor	NXConvertCMYKAToColor (float <i>cyan</i> , float <i>magenta</i> , float <i>yellow</i> , float <i>black</i> , float <i>alpha</i>)
NXColor	NXConvertHSBAToColor (float <i>hue</i> , float <i>saturation</i> , float <i>brightness</i> , float <i>alpha</i>)
NXColor	NXConvertGrayAlphaToColor (float <i>gray</i> , float <i>alpha</i>)
NXColor	NXConvertRGBToColor (float <i>red</i> , float <i>green</i> , float <i>blue</i>)
NXColor	NXConvertCMYKToColor (float <i>cyan</i> , float <i>magenta</i> , float <i>yellow</i> , float <i>black</i>)
NXColor	NXConvertHSBToColor (float <i>hue</i> , float <i>saturation</i> , float <i>brightness</i>)
NXColor	NXConvertGrayToColor (float <i>gray</i>)

Convert a color value to its standard components

void	NXConvertColorToRGBA (NXColor <i>color</i> , float <i>*red</i> , float <i>*green</i> , float <i>*blue</i> , float <i>*alpha</i>)
void	NXConvertColorToCMYKA (NXColor <i>color</i> , float <i>*cyan</i> , float <i>*magenta</i> , float <i>*yellow</i> , float <i>*black</i> , float <i>*alpha</i>)
void	NXConvertColorToHSBA (NXColor <i>color</i> , float <i>*hue</i> , float <i>*saturation</i> , float <i>*brightness</i> , float <i>*alpha</i>)
void	NXConvertColorToGrayAlpha (NXColor <i>color</i> , float <i>*gray</i> , float <i>*alpha</i>)
void	NXConvertColorToRGB (NXColor <i>color</i> , float <i>*red</i> , float <i>*green</i> , float <i>*blue</i>)
void	NXConvertColorToCMYK (NXColor <i>color</i> , float <i>*cyan</i> , float <i>*magenta</i> , float <i>*yellow</i> , float <i>*black</i>)
void	NXConvertColorToHSB (NXColor <i>color</i> , float <i>*hue</i> , float <i>*saturation</i> , float <i>*brightness</i>)
void	NXConvertColorToGray (NXColor <i>color</i> , float <i>*gray</i>)

Modify a color by changing one of its components

NXColor	NXChangeRedComponent (NXColor <i>color</i> , float <i>red</i>)
NXColor	NXChangeGreenComponent (NXColor <i>color</i> , float <i>green</i>)
NXColor	NXChangeBlueComponent (NXColor <i>color</i> , float <i>blue</i>)
NXColor	NXChangeCyanComponent (NXColor <i>color</i> , float <i>cyan</i>)
NXColor	NXChangeMagentaComponent (NXColor <i>color</i> , float <i>magenta</i>)
NXColor	NXChangeYellowComponent (NXColor <i>color</i> , float <i>yellow</i>)
NXColor	NXChangeBlackComponent (NXColor <i>color</i> , float <i>black</i>)
NXColor	NXChangeHueComponent (NXColor <i>color</i> , float <i>hue</i>)
NXColor	NXChangeSaturationComponent (NXColor <i>color</i> , float <i>saturation</i>)
NXColor	NXChangeBrightnessComponent (NXColor <i>color</i> , float <i>brightness</i>)
NXColor	NXChangeGrayComponent (NXColor <i>color</i> , float <i>gray</i>)
NXColor	NXChangeAlphaComponent (NXColor <i>color</i> , float <i>alpha</i>)

Isolate one component of a color

float	NXRedComponent (NXColor <i>color</i>)
float	NXGreenComponent (NXColor <i>color</i>)
float	NXBlueComponent (NXColor <i>color</i>)
float	NXCyanComponent (NXColor <i>color</i>)
float	NXMagentaComponent (NXColor <i>color</i>)
float	NYellowComponent (NXColor <i>color</i>)
float	NXBlackComponent (NXColor <i>color</i>)
float	NXHueComponent (NXColor <i>color</i>)
float	NXSaturationComponent (NXColor <i>color</i>)
float	NXBrightnessComponent (NXColor <i>color</i>)
float	NXGrayComponent (NXColor <i>color</i>)
float	NXAlphaComponent (NXColor <i>color</i>)

Test whether two colors are the same

BOOL	NXEqualColor (NXColor <i>oneColor</i> , NXColor <i>anotherColor</i>)
------	--

Get information about color space and window depth

NXColorSpace	NXColorSpaceFromDepth (NXWindowDepth <i>depth</i>)
int	NXBPSFromDepth (NXWindowDepth <i>depth</i>)
int	NXNumberOfColorComponents (NXColorSpace <i>space</i>)
BOOL	NXGetBestDepth (NXWindowDepth <i>*depth</i> , int <i>numColors</i> , int <i>bps</i>)

Read and write a color from a typed stream

NXColor	NXReadColor (NXTypedStream <i>*stream</i>)
void	NXWriteColor (NXTypedStream <i>*stream</i> , NXColor <i>color</i>)

Read and write a color from a pasteboard

NXColor	NXReadColorFromPasteboard (id <i>pasteboard</i>)
void	NXWriteColorToPasteboard (id <i>pasteboard</i> , NXColor <i>color</i>)

Set the current color

void	NXSetColor (NXColor <i>color</i>)
------	---

Reading the color at a screen position

NXColor	NXReadPixel (const NXPoint <i>*location</i>)
---------	--

Associate named colors with their color lists

const char *	NXColorListName (NXColor <i>color</i>)
const char *	NXColorName (NXColor <i>color</i>)
BOOL	NXFindColorNamed (const char <i>*colorList</i> , const char <i>*colorName</i> , NXColor <i>*color</i>)

Text Functions

Filter characters entered into Text object

unsigned short	NXFieldFilter (unsigned short <i>theChar</i> , int <i>flags</i> , unsigned short <i>charSet</i>)
unsigned short	NXEditorFilter (unsigned short <i>theChar</i> , int <i>flags</i> , unsigned short <i>charSet</i>)

Calculate or draw a line of text (in Text object)

int	NXScanALine (id <i>self</i> , NXLayerInfo <i>*layInfo</i>)
int	NXDrawALine (id <i>self</i> , NXLayerInfo <i>*layInfo</i>)

Calculate font ascender, descender, and line height (in Text object)

void	NXTextFontInfo (id <i>fontId</i> , NXCoord <i>*ascender</i> , NXCoord <i>*descender</i> , NXCoord <i>*lineHeight</i>)
------	---

Access Text object's word tables

void	NXReadWordTable (NXZone <i>*zone</i> , NXStream <i>*stream</i> , unsigned char <i>**preSelSmart</i> , unsigned char <i>**postSelSmart</i> , unsigned char <i>**charCategories</i> , NXFSM <i>**wrapBreaks</i> , int <i>*wrapBreaksCount</i> , NXFSM <i>**clickBreaks</i> , int <i>*clickBreaksCount</i> , BOOL <i>*charWrap</i>)
void	NXWriteWordTable (NXStream <i>*stream</i> , const unsigned char <i>*preSelSmart</i> , const unsigned char <i>*postSelSmart</i> , const unsigned char <i>*charCategories</i> , const NXFSM <i>*wrapBreaks</i> , int <i>wrapBreaksCount</i> , const NXFSM <i>*clickBreaks</i> , int <i>clickBreaksCount</i> , BOOL <i>charWrap</i>)

Provide table-driven string ordering service

int	NXOrderStrings (const unsigned char <i>*string1</i> , const unsigned char <i>*string2</i> , BOOL <i>caseSensitive</i> , int <i>length</i> , NXStringOrderTable <i>*table</i>)
NXStringOrderTable *	NXDefaultStringOrderTable (void)

Imaging Functions

Copy an image

void	NXCopyBits (int <i>gstate</i> , const NXRect <i>*aRect</i> , const NXPoint <i>*aPoint</i>)
void	NXCopyBitmapFromGstate (int <i>gstate</i> , const NXRect <i>*srcRect</i> , const NXRect <i>*destRect</i>)

Render and read bitmap images

void	NXDrawBitmap (const NXRect <i>*rect</i> ,int <i>pixelsWide</i> ,int <i>pixelsHigh</i> ,int <i>bitsPerSample</i> , int <i>samplesPerPixel</i> , int <i>bitsPerPixel</i> , int <i>bytesPerRow</i> , BOOL <i>isPlanar</i> ,BOOL <i>hasAlpha</i> , NXColorSpace <i>colorSpace</i> , const unsigned char <i>*const data</i> [5])
void	NXReadBitmap (const NXRect <i>*rect</i> , int <i>pixelsWide</i> , int <i>pixelsHigh</i> , int <i>bps</i> , int <i>spp</i> , int <i>config</i> , int <i>mask</i> , void <i>*data1</i> , void <i>*data2</i> , void <i>*data3</i> , void <i>*data4</i> , void <i>*data5</i>)
void	NXSizeBitmap (const NXRect <i>*rect</i> , int <i>*size</i> , int <i>*pixelsWide</i> , int <i>*pixelsHigh</i> , int <i>*bps</i> , int <i>*spp</i> , int <i>*config</i> , int <i>*mask</i>)

Object Management Functions

Refer to objects by name

id	NXGetNamedObject (const char <i>*name</i> , id <i>owner</i>)
const char *	NXGetObjectName (id <i>theObject</i>)
int	NXNameObject (const char <i>*name</i> , id <i>theObject</i> , id <i>owner</i>)
int	NXUnnameObject (const char <i>*name</i> , id <i>owner</i>)

Get information about an application's windows

void	NXCountWindows (int <i>*count</i>)
void	NXWindowList (int <i>size</i> , int <i>list</i> [])

Convert local and global window numbers

void	NXConvertWinNumToGlobal (int <i>winNum</i> , unsigned int <i>*globalNum</i>)
void	NXConvertGlobalToWinNum (int <i>globalNum</i> , unsigned int <i>*winNum</i>)

Set up a pop-up list

void	NXAttachPopUpList (id <i>button</i> , PopUpList <i>*popUpList</i>)
id	NXCreatePopUpListButton (PopUpList <i>*popUpList</i>)

Create or free an attention panel

int	NXRunAlertPanel (const char <i>*title</i> , const char <i>*msg</i> , const char <i>*defaultButton</i> , const char <i>*alternateButton</i> , const char <i>*otherButton</i> , ...)
int	NXRunLocalizedAlertPanel (const char <i>*table</i> , const char <i>*title</i> , const char <i>*msg</i> , const char <i>*defaultButton</i> , const char <i>*alternateButton</i> , const char <i>*otherButton</i> , ...)
id	NXGetAlertPanel (const char <i>*title</i> , const char <i>*msg</i> , const char <i>*firstButton</i> , const char <i>*alternateButton</i> , const char <i>*otherButton</i> , ...)
void	NXFreeAlertPanel (id <i>alertPanel</i>)

Error-Handling Functions

Set and return an error handler

void	NXDefaultTopLevelErrorHandler (NXHandler <i>*errorState</i>)
NXTopLevelErrorHandler *	NXSetTopLevelErrorHandler (NXTopLevelErrorHandler <i>*proc</i>)

/ a macro */*

NXTopLevelErrorHandler *	NXTopLevelErrorHandler (void) <i>/* a macro */</i>
--------------------------	---

Manage error reporting

void	NXRegisterErrorReporter (int <i>min</i> , int <i>max</i> , NXErrorReporter <i>*proc</i>)
void	NXRemoveErrorReporter (int <i>code</i>)
void	NXReportError (NXHandler <i>*errorState</i>)

Write a formatted error string

void **NXLogError**(const char **format*, ...)

Typed Stream Functions

Read or write NeXT-defined data types from or to a typed stream

void **NXReadPoint**(NXTypedStream **typedStream*, NXPoint **aPoint*)
void **NXWritePoint**(NXTypedStream **typedStream*, const NXPoint **aPoint*)
void **NXReadSize**(NXTypedStream **typedStream*, NXSize **aSize*)
void **NXWriteSize**(NXTypedStream **typedStream*, const NXSize **aSize*)
void **NXReadRect**(NXTypedStream **typedStream*, NXRect **aRect*)
void **NXWriteRect**(NXTypedStream **typedStream*, const NXRect **aRect*)

Remote Messaging Functions

Save data received in a remote message

char * **NXCopyInputData**(int *parameter*)
char * **NXCopyOutputData**(int *parameter*)

Get send rights to an application port

port_t **NXPortFromName**(const char **name*, const char **host*)
port_t **NXPortNameLookup**(const char **name*, const char **host*)

Match an Objective-C method and a receiver to a remote message

NXRemoteMethod * **NXRemoteMethodFromSel**(SEL *aSelector*, NXRemoteMethod **methods*)
id **NXResponsibleDelegate**(id *aListener*, SEL *aSelector*)

Services Menu Functions

Determine whether an item is included in Services menus

int **NXSetServicesMenuItemEnabled**(const char **item*, BOOL *flag*)
BOOL **NXIsServicesMenuItemEnabled**(const char **item*)

Programatically invoke a service

BOOL **NXPerformService**(const char **item*, Pasteboard **pboard*)

Force Services menu to update based on new services

void **NXUpdateDynamicServices**(void)

Event Function

Access event record in event queue

NXEvent *

NXGetOrPeekEvent(DPSContext *context*, NXEvent **anEvent*, int *mask*, double *timeout*, int *threshold*, int *peek*)

Memory Allocation Functions

Macros to allocate memory

type-name *

type-name *

void

NX_MALLOC(*type-name* **var*, *type-name*, int *num*)

NX_REALLOC(*type-name* **var*, *type-name*, int *num*)

NX_FREE(void **pointer*)

Allocate a variable-sized array

NXChunk *

NXChunk *

NXChunk *

NXChunk *

NXChunk *

NXChunk *

NXChunk *

NXChunk *

NXChunkMalloc(int *growBy*, int *initUsed*)

NXChunkRealloc(NXChunk **pc*)

NXChunkGrow(NXChunk **pc*, int *newUsed*)

NXChunkCopy(NXChunk **pc*, NXChunk **dpc*)

NXChunkZoneMalloc(int *growBy*, int *initUsed*, NXZone **zone*)

NXChunkZoneRealloc(NXChunk **pc*, NXZone **zone*)

NXChunkZoneGrow(NXChunk **pc*, int *newUsed*, NXZone **zone*)

NXChunkZoneCopy(NXChunk **pc*, NXChunk **dpc*, NXZone **zone*)

Macros to allocate zone memory

type-name *

type-name *

NX_ZONEMALLOC(NXZone **zone*, *type-name* **var*, *type-name*, int *num*)

NX_ZONEREALLOC(NXZone **zone*, *type-name* **var*, *type-name*, int *num*)

Other Application Kit Functions

Get user's home directory and name

const char *

const char *

NXHomeDirectory(void)

NXUserName(void)

Synchronize the application with the Window Server

void

NXPing(void)

Find dimensions of specified paper type

const NXSize *

NXFindPaperSize(const char **paperName*)

Play the system beep

void

NXBeep(void)

Set up timer events

NXTrackingTimer *

void

NXBeginTimer(NXTrackingTimer **timer*, double *delay*, double *period*)

NXEndTimer(NXTrackingTimer **timer*)

Allow journaling during direct mouse tracking

void NXJournalMouse(void)

Set or copy current graphics state object

void NXSetGState(int *gstate*)
void NXCopyCurrentGState(int *gstate*)

Report user's request to abort

BOOL NXUserAborted(void)
void NXResetUserAbort(void)

Return file-related pasteboard types

NXAtom NXCreateFileContentsPboardType(const char **fileType*)
NXAtom NXCreateFilenamePboardType(const char **filename*)
const char * NXGetFileType(const char **pboardType*)
const char ** NXGetFileTypes(const char *const **pboardTypes*)

Find unique files from a path

int NXCompleteFilename(char **path*, int *maxPathSize*)

Draw a distinctive outline around linked data

void NXFrameLinkRect(const NXRect **aRect*, BOOL *isDestination*)
float NXLinkFrameThickness(void)

Get the amount of memory used by the Window Server

int NXGetWindowServerMemory(DPSContext *context*, int **virtualMemory*, int
 **windowBackingMemory*, NXStream **windowDumpStream*)

Macro to write an error message

void NX_ASSERT(int *exp*, char **msg*)

Macro for debugging Display PostScript

void NX_PSDEBUG(void)