Functions

Character Classification Functions

Classify NeXTSTEP-Encoded Values

int **NXIsAlpha(**unsigned int *c***)**
int **NXIsUpper(**unsigned int *c***)**
int **NXIsLower(**unsigned int *c***)**
int **NXIsDigit(**unsigned int *c***)**
int **NXIsXDigit(**unsigned int *c***)**
int **NXIsAlNum(**unsigned int *c***)**
int **NXIsSpace(**unsigned int *c***)**
int **NXIsPunct(**unsigned int *c***)**
int **NXIsPrint(**unsigned int *c***)**
int **NXIsGraph(**unsigned int *c***)**
int **NXIsCntrl(**unsigned int *c***)**
int **NXIsAscii(**unsigned int *c***)**

Convert NeXTSTEP-Encoded Characters

unsigned char * **NXToAscii(**unsigned int *c***)**
int **NXToLower(**unsigned int *c***)**
int **NXToUpper(**unsigned int *c***)**

Defaults System Functions

Set or Read Default Parameters

int **NXRegisterDefaults(**const char **owner*, const NXDefaultsVector *vector***)**
const char * **NXGetDefaultValue(**const char **owner*, const char **name***)**
const char * **NXReadDefault(**const char **owner*, const char **name***)**
int **NXRemoveDefault(**const char **owner*, const char **name***)**
int **NXSetDefault(**const char **owner*, const char **name*, const char *value**)**
const char * **NXUpdateDefault(**const char **owner*, const char **name***)**
void **NXUpdateDefaults(**void**)**
int **NXWriteDefault(**const char **owner*, const char **name*, const char *value**)**
int **NXWriteDefaults(**const char **owner*, NXDefaultsVector *vector***)**
const char * **NXSetDefaultsUser(**const char **newUser***)**

Error-Handling Functions

Macros to Raise an Exception

void **NX_RAISE(**int *code*, const void **data1*, const void **data2***)**

void **NX_RERAISE(**void**)**
val **NX_VALRETURN(**val**)**
void **NX_VOIDRETURN**

Set and Return an Exception Raiser

void **NXDefaultExceptionRaiser(**int *code*, const void **data1*, const void **data2***)**
void **NXSetExceptionRaiser(**NXExceptionRaiser **procedure***)**
NXExceptionRaiser * **NXGetExceptionRaiser(**void**)**

Macros to Handle Uncaught Exceptions

void **NXSetUncaughtExceptionHandler(**NXUncaughtExceptionHandler **proc***)**
NXUncaughtExceptionHandler * **NXGetUncaughtExceptionHandler(**void**)**

Manage the Error Data Buffer

void **NXAllocErrorData(**int *size*, void ***data***)**
void **NXResetErrorData(**void**)**

Stream Functions

Manipulate a Memory Stream

NXStream * **NXOpenMemory(**const char **address*, int *size*, int *mode***)**
NXStream * **NXMapFile(**const char **pathName*, int *mode***)**
int **NXSaveToFile(**NXStream **stream*, const char **name***)**
void **NXCloseMemory(**NXStream **stream*, int *option***)**
void **NXGetMemoryBuffer(**NXStream **stream*, char ***streambuf*, int **len*, int **maxLen***)**

Open a File Stream or a Mach Port Stream

NXStream * **NXOpenFile(**int *fd*, int *mode***)**
NXStream * **NXOpenPort(**port_t *port*, int *mode***)**

Close a Stream

void **NXClose(**NXStream **stream***)**

Read From or Write to a Stream

int **NXRead(**NXStream **stream*, void **buf*, int *count***)**
int **NXWrite(**NXStream **stream*, const void **buf*, int *count***)**

Read or Write Formatted Data from or to a Stream

int **NXPutc(**NXStream **stream*, char *c***)** /* a macro */
int **NXGetc(**NXStream **stream***)** /* a macro */
void **NXUngetc(**NXStream **stream***)**
int **NXScanf(**NXStream **stream*, const char **format*, ...**)**
void **NXPrintf(**NXStream **stream*, const char **format*, ...**)**
int **NXVScanf(**NXStream **stream*, const char **format*, va_list *argList***)**
void **NXVPrintf(**NXStream **stream*, const char **format*, va_list *argList***)**

Register a Procedure for Formatting Data Written to a Stream

void **NXRegisterPrintfProc(**char *formatChar*, NXPrintfProc **proc*, void **procData***)**

Flush a Stream

int **NXFlush(**NXStream **stream***)**

Set or Report Current Position in a Stream

void **NXSeek(**NXStream **stream*, long *offset*, int *ptrName***)**
long **NXTell(**NXStream **stream***)**
BOOL **NXAtEOS(**NXStream **stream***)** /* a macro */

Support a User-defined Stream

NXStream * **NXStreamCreate(**int *mode*, int *createBuf***)**
NXStream * **NXStreamCreateFromZone(**int *mode*, int *createBuf*, NXZone **zone***)**
void **NXStreamDestroy(**NXStream **stream***)**
int **NXDefaultRead(**NXStream **stream*, void **buf*, int *count***)**
int **NXDefaultWrite(**NXStream **stream*, const void **buf*, int *count***)**
int **NXFill(**NXStream **stream***)**
void **NXChangeBuffer(**NXStream **stream***)**

Typed Stream Functions

Open or Close a Typed Stream

NXTypedStream* **NXOpenTypedStream(**NXStream **stream*, int *mode***)**
void **NXCloseTypedStream(**NXTypedStream **stream***)**
NXTypedStream* **NXOpenTypedStreamForFile(**const char **fileName*, int *mode***)**

Read or Write Objective C Objects from or to a Typed Stream

id **NXReadObject(**NXTypedStream **stream***)**
void **NXWriteObject(**NXTypedStream **stream*, id *object***)**
void **NXWriteObjectReference(**NXTypedStream **stream*, id *object***)**
void **NXWriteRootObject(**NXTypedStream **stream*, id *rootObject***)**

Read or Write Arbitrary Data from or to a Typed Stream

void **NXReadType(**NXTypedStream **stream*, const char **type*, void **data***)**
void **NXWriteType(**NXTypedStream **stream*, const char **type*, const void **data***)**
void **NXReadTypes(**NXTypedStream **stream*, const char **types*, ...**)**
void **NXWriteTypes(**NXTypedStream **stream*, const char **types*, ...**)**

Read or Write Arrays from or to a Typed Stream

void **NXReadArray(**NXTypedStream **stream*, const char **dataType*, int *count*, void **data***)**
void **NXWriteArray(**NXTypedStream **stream*, const char **dataType*, int *count*, const void **data***)**

Read or Write an Object from or to a Typed-Stream Memory Buffer

id **NXReadObjectFromBuffer(**const char ******buffer*, int *length***)**
char * **NXWriteRootObjectToBuffer(**id *object*, int ******length***)**
void **NXFreeObjectBuffer(**char ******buffer*, int *length***)**

Determine Whether There's More Data to Be Read

BOOL **NXEndOfTypedStream(**NXTypedStream ******stream***)**

Flush a Typed Stream

void **NXFlushTypedStream(**NXTypedStream ******stream***)**

Get the Version Number of a Class

int **NXTypedStreamClassVersion(**NXTypedStream ******stream*, const char ******className***)**

Get or Set the Zone for a Typed Stream

NXZone **NXGetTypedStreamZone(**NXTypedStream ******stream***)**
void **NXSetTypedStreamZone(**NXTypedStream ******stream*, NXZone ******zone***)**
id **NXReadObjectFromBufferWithZone(**const char ******buffer*, int *length*, NXZone ******zone***)**

Memory Allocation Functions

Zone Memory Allocation

void * **NXZoneMalloc**(NXZone ******zone*, size_t *size*)
void * **NXZoneCalloc**(NXZone ******zone*, size_t *numElems*, size_t *numBytes*)
void * **NXZoneRealloc**(NXZone ******zone*, void ******ptr*, size_t *size*)
void **NXZoneFree**(NXZone ******zone*, void ******ptr*)
NXZone * **NXDefaultMallocZone**(void)
NXZone * **NXCreateZone**(size_t *startSize*, size_t *granularity*, int *canFree*)
NXZone * **NXCreateChildZone**(NXZone ******parentZone*, size_t *startSize*, size_t *granularity*, int *canFree*)
void **NXMergeZone**(NXZone ******zone*)
void **NXDestroyZone**(NXZone ******zone*)
NXZone * **NXZoneFromPtr**(void ******ptr*)
void **NXZonePtrInfo**(void ******ptr*)
int **NXMallocCheck**(void)
void **NXNameZone**(NXZone ******zonep*, const char ******name*)

Hash and String Table Functions

Create, Manipulate, and Free a Hash Table

NXHashTable * **NXCreateHashTable(**NXHashTablePrototype *prototype*, unsigned *capacity*, const void ******info***)**
NXHashTable * **NXCreateHashTableFromZone(**NXHashTablePrototype *prototype*, unsigned

capacity, const void *_info_, NXZone *_zone_)

void	**NXFreeHashTable(**NXHashTable *_table_**)**
void	**NXEmptyHashTable(**NXHashTable *_table_**)**
void	**NXResetHashTable(**NXHashTable *_table_**)**
NXHashTable *	**NXCopyHashTable(**NXHashTable *_table_**)**
BOOL	**NXCompareHashTables(**NXHashTable *_table1_, NXHashTable *_table2_**)**
unsigned	**NXPtrHash(**const void *_info_, const void *_data_**)**
unsigned	**NXStrHash(**const void *_info_, const void *_data_**)**
int	**NXPtrIsEqual(**const void *_info_, const void *_data1_, const void *_data2_**)**
int	**NXStrIsEqual(**const void *_info_, const void *_data1_, const void *_data2_**)**
void	**NXNoEffectFree(**const void *_info_, void *_data_**)**
void	**NXReallyFree(**const void *_info_, void *_data_**)**

Manipulate the Elements of a Hash Table

void *	**NXHashInsert(**NXHashTable *_table_, const void *_data_**)**
void *	**NXHashInsertIfAbsent(**NXHashTable *_table_, const void *_data_**)**
int	**NXHashMember(**NXHashTable *_table_, const void *_data_**)**
void *	**NXHashGet(**NXHashTable *_table_, const void *_data_**)**
void *	**NXHashRemove(**NXHashTable *_table_, const void *_data_**)**
unsigned	**NXCountHashTable(**NXHashTable *_table_**)**
NXHashState	**NXInitHashState(**NXHashTable *_table_**)**
int	**NXNextHashState(**NXHashTable *_table_, NXHashState *_state_, void **_data_**)**

String Functions

Get Localized Versions of Strings

const char *	**NXLocalizedString(**const char *_key_, const char *_value_, _comment_**)**
const char *	**NXLocalizedStringFromTable(**const char *_table_, const char *_key_, const char *_value_, _comment_**)**
const char *	**NXLocalizedStringFromTableInBundle(**const char *_table_, NXBundle *_bundle_, const char *_key_, const char *_value_, _comment_**)**
const char *	**NXLoadLocalizedStringFromTableInBundle(**const char *_table_, NXBundle *_bundle_, const char *_key_, const char *_value_**)**

Create a Unique String

NXAtom	**NXUniqueString(**const char *_buffer_**)**
NXAtom	**NXUniqueStringWithLength(**const char *_buffer_, int _length_**)**
NXAtom	**NXUniqueStringNoCopy(**const char *_buffer_**)**
char *	**NXCopyStringBuffer(**const char *_buffer_**)**
char *	**NXCopyStringBufferFromZone(**const char *_buffer_, NXZone *_zone_**)**

Miscellaneous Functions

Get a Pointer to the Objects Stored in a List

id *	**NX_ADDRESS(**List *_aList_**)**

Search for and Read a File

int **NXFilePathSearch(**const char **envVarName*, const char **defaultPath*, int *leftToRight*, const char **fileName*, int (**funcPtr*)(), void **funcArg***)**