

3

# Common Classes and Functions

**Library:** libsys\_s.a

**Header File Directories:** /NextDeveloper/Headers/objc  
/NextDeveloper/Headers/streams  
/NextDeveloper/Headers/defaults  
/NextDeveloper/Headers/appkit

**Import:** appkit/appkit.h  
or individual header files

## Introduction

The classes and functions described in this chapter can serve a wide variety of different kinds of applications. They get the name "common" from the fact that they're generally useful and commonly used. Most provide ways of managing data and resources within a program. They can be used in conjunction with any of the NEXTSTEP software kits.

**Figure 3-1.** Common Classes Inheritance Hierarchy

The table below is a guide to the facilities documented here. The first column states, in a few words, a kind of data-management or resource-management service. The second column says where to look in the chapter; it names the class or principal function that provides the service. Consult the documentation for the functions and classes mentioned to learn whether they're right for your application.

Facility	Class or Function
Storage allocation NXCreateZone() NXMallocCheck() Storage class	NXZoneMalloc()
Unordered collections NXCreateHashTable() NXHashInsert() NXStringTable class	HashTable class
Ordered collections Storage class	List class

Recording user preferences	NXRegisterDefaults()
Localizing resources NXLocalizedString()	NXBundle class
Exception handling NXDefaultExceptionRaiser() NXSetUncaughtExceptionHandler()	NX_RAISE() NXAllocErrorData()
Dynamic loading of code	NXBundle class
Input-output streams NXOpenMemory() NXRead() NXPutc()	NXOpenFile()
Archiving and typed streams	NXOpenTypedStream() NXReadObject() NXReadType()
Classifying and converting characters	NXIsAlpha() NXToAscii()

Because the Application Kit makes use of all the common classes and almost all of the common functions, importing the **appkit.h** header file automatically imports all but one of the header files that declare common classes and functions. Since **appkit.h** corresponds to a precompiled version of the header files, importing it rather than individual files dramatically reduces the time required for compilation. The only common header file that's omitted, and that you might need to import individually, is **streams/streamsimpl.h**. It declares rarely used functions that implement new versions of a stream.