Introduction

The Objective-C language defers as many decisions as it can from compile time and link time to runtime. Whenever possible, it does things dynamically. This means that the language requires not just a compiler, but also a runtime system to execute the compiled code. The runtime system acts as a kind of operating system for the Objective-C language; it's what makes the language work.

This document looks at the `NSObject` class and how Objective-C programs interact with the runtime system. In particular, it examines the paradigms for dynamically loading new classes at runtime, and forwarding messages to other objects. It also provides information about how you can find information about objects while your program is running.

You should read this document to gain an understanding of how the Objective-C runtime system works and how you can take advantage of it. Typically, though, there should be little reason for you to need to know and understand this material to write a Cocoa application.

Organization of This Document

This document has the following chapters:

- Runtime Versions and Platforms
- Interacting with the Runtime
- Messaging
- Dynamic Method Resolution
- Message Forwarding
- Type Encodings
- Declared Properties

See Also

Objective-C Runtime Reference describes the data structures and functions of the Objective-C runtime support library. Your programs can use these interfaces to interact with the Objective-C runtime system. For example, you can add classes or methods, or obtain a list of all class definitions for loaded classes.

Programming with Objective-C describes the Objective-C language.

Objective-C Release Notes describes some of the changes in the Objective-C runtime in recent releases of OS X.