Using the Sound Manager

Getting started with sound creation

The **Sound Manager** provides a wide range of methods for creating sound and manipulating audio data on the Macintosh. Usually, your application needs to use only a few of the many routines or sound commands that are available. You can also use **Sound Manager** routines to record sounds through any available sound input hardware.

The **Sound Manager** routines can be divided into high-level routines and low-level routines. The high-level routines (like **SndRecord**, **SndPlay**, and **SysBeep**) give you the ability to produce very complex audio output at very little programming expense. The next section shows how your application can produce sounds simply by obtaining a handle to an existing 'snd ' resource and passing that handle to the **SndPlay** function. Moreover, if the data in the 'snd ' resource is stored in a compressed format, **SndPlay** automatically expands it for play-back in real time without further intervention from your application.

Although the high-level **Sound Manager** routines are sufficient for many applications, low-level **Sound Manager** routines are available to provide your application with much greater control over sound recording and production than is provided by the high-level routines. Using these low-level routines, your application can record directly from sound input devices, allocate and release sound channels, queue sound commands to a channel or bypass a sound queue altogether, perform modifications on sound data and commands sent into a channel, create and mix multiple channels of sound, compress and expand audio data, disable and enable the system alert sound, obtain information about current sound activity, and play sounds continuously from disk.

Some of these operations are carried out by specialized low-level routines, but most of them are accomplished by passing appropriate sound commands to the **SndDoCommand**, **SndDoImmediate**, and **SndControl** functions. For example, your application can alter the pitch of a sampled sound that is currently playing by calling **SndDoImmediate** with the rateCmd command as one of its parameters.

Some of the **Sound Manager** routines and commands cannot be called at interrupt time because they attempt to allocate or release memory. In particular, the routines **SndNewChannel**, **SndDisposeChannel**, **SndAddModifier**, **SysBeep**, **SndPlay**, **SndStartFilePlay**, **SndRecord**, and **SndRecordToFile** cannot be called at interrupt time. In addition, callback procedures, specified in calls to **SndNewChannel** and **SndStartFilePlay**, and doubleback routines, specified in calls to **SndPlayDoubleBuffer**, are executed at interrupt time and therefore must not allocate, release, or move memory. You can safely call all other **Sound Manager** routines at interrupt time.