NAME audioplay – play audio files

SYNOPSIS **audioplay** [**–iV**] [**–v** *vol*] [**–b** *bal*] [**–p speaker** | **headphone** | **line**] [**–d** *dev*] [*file* ...]

AVAILABILITY SUNWaudio

DESCRIPTION

audioplay copies the named audio files (or the standard input if no filenames are present) to the audio device. If no input file is specified and standard input is a tty, the port, volume, and balance settings specified on the command line will be applied and the program will exit.

The input files must contain a valid audio file header. The encoding information in this header is matched against the capabilities of the audio device and, if the data formats are incompatible, an error message is printed and the file is skipped. Compressed ADPCM (G.721) monaural audio data is automatically uncompressed before playing.

Minor deviations in sampling frequency (that is, less than 1%) are ordinarily ignored. This allows, for instance, data sampled at 8012 Hz to be played on an audio device that only supports 8000 Hz. If the **–V** option is present, such deviations are flagged with warning messages.

OPTIONS

–i *Immediate*: If the audio device is unavailable (that is, another process currently has write access), **audioplay** ordinarily waits until it can obtain access to the device. When the **–i** option is present, **audioplay** prints an error message and exits immediately if the device is busy.

–V *Verbose*: Print messages on the standard error when waiting for access to the audio device or when sample rate deviations are detected.

–v *vol* *Volume*: The output volume is set to the specified value before playing begins, and is reset to its previous level when **audioplay** exits. The *vol* argument is an integer value between 0 and 100, inclusive. If this argument is not specified, the output volume remains at the level most recently set by any process.

–b *bal* *Balance*: The output balance is set to the specified value before playing begins, and is reset to its previous level when **audioplay** exits. The *bal* argument is an integer value between -100 and 100, inclusive. A value of -100 indicates left balance, 0 middle, and 100 right. If this argument is not specified, the output balance remains at the level most recently set by any process.

–p speaker | **headphone** | **line**
Output Port: Select the built-in **speaker**, (the default), **headphone** jack, or **line** out as the destination of the audio output signal. If this argument is not specified, the output port will remain unchanged. *Not all audio adapters support all of the output ports. If the named port does not exist, an appropriate substitute will be used.*

–d *dev* *Device*: The *dev* argument specifies an alternate audio device to which output should be directed. If the **–d** option is not specified, the **AUDIODEV**

environment variable is consulted (see below). Otherwise, **/dev/audio** is used as the default audio device.

file *File Specification*: Audio files named on the command line are played sequentially. If no filenames are present, the standard input stream (if it is not a tty) is played (it, too, must contain an audio file header). The special filename '–' may be used to read the standard input stream instead of a file. If a relative path name is supplied, the **AUDIOPATH** environment variable is consulted (see below).

–**?** *Help*: Print a command line usage message.

ENVIRONMENT

AUDIODEV
The full path name of the audio device to write to, if no –d argument is supplied. If the **AUDIODEV** variable is not set, **/dev/audio** is used.

AUDIOPATH
A colon-separated list of directories in which to search for audio files whose names are given by relative pathnames. The current directory ("**.**") may be specified explicitly in the search path. If the **AUDIOPATH** variable is not set, only the current directory will be searched.

SEE ALSO

audioconvert(1), **audiorecord**(1)

Sparc Only
audio(7I), **audioamd**(7D), **dbri**(7D)

x86 Only
sbpro(7D)

BUGS

audioplay currently supports a limited set of audio format conversions. If the audio file is not in a format supported by the audio device, it must first be converted. For example, to convert to voice format on the fly, use the command:

```
example% audioconvert –f voice myfile | audioplay
```

The format conversion will not always be able to keep up with the audio output. If this is the case, you should convert to a temporary file before playing the data.