

# Mixed Sound with ARTS and ALSA

A small HOWTO found at <http://wiki.splitbrain.org/alsa> and which I edited according to my own views. It explains to configure your system to play sounds from various applications simultaneously. Think of applications like XMMS, GAIM, MPlayer and Firefox (with Flashplugin).

ALSA comes with a method to mix multiple sound streams which is called *dmix*. ALSA-enabled applications should be able to all access the device at the same time when it is set up. To do this you'll need to set up either a system-wide `/etc/asound.conf` or a `~/.asoundrc` file in your home directory.



For ALSA 1.0.9rc2 and higher you don't need to setup *dmix*. *Dmix* is enabled as default for soundcards which don't support hardware mixing. Slackware 10.2 has a mix of `alsa-driver 1.0.9b`, `alsa-lib 1.0.9` and `alsa-utils 1.0.9a` so I can't guarantee it will work out of the box. Slackware 11.0 however will definitely not need the setup as described below!

## `/etc/asound.conf` (1st version)

Here is an example `/etc/asound.conf` which I took from <http://alsa.opensrc.org/index.php?page=Dmix+Kde+-+arts%2C+ESD+and+SDL+quick+and+dirty+HOWTO>. It works for me, but you might also want to have a look at the rest of this page.

```
pcm.ossmix {
    type dmix
    ipc_key 1027          # must be unique!
    slave {
        pcm "hw:0,0"     # you cannot use a "plug"
                        # device here, darn.

        period_time 0
        period_size 1024
        buffer_size 4096
        #format "S32_LE"
        #periods 128     # dito.
        rate 44100       # with rate 8000 you *will*
                        # hear, if ossmix is used :)
    }
    bindings {
        0 0              # from 0 => to 0
        1 1              # from 1 => to 1
    }
}
pcm.!default {
    type plug
    slave.pcm "ossmix"
}
```

```
# mixer0 like above
ctl.mixer0 {
    type hw
    card 0
}
```

## **/etc/asound.conf (2nd version)**

The instructions at <http://alsa.opensrc.org/index.php?page=asym> are the source of what follows on the rest of this page. If the above does not work for you, you might want to take a look at the story below.

This is what `/etc/asound.conf` could look like also:

```
#asym fun start here. we define one pcm device called "dmixed"
pcm.dmixed {
    ipc_key 1025
    type dmix
    slave {
        pcm "hw:0,0"
        period_time 0
        period_size 1024
        buffer_size 8192
        rate 48000
    }
}

#one called "dsnooped" for capturing
pcm.dsnooped {
    ipc_key 1026
    type dsnoop
    slave.pcm "hw:0,0"
}

#and this is the real magic
pcm.asymed {
    type asym
    playback.pcm "dmixed"
    capture.pcm "dsnooped"
}

#a quick plug plugin for above device to do the converting magic. saves
#typing when setting the pcm name in an alsa app
pcm.pasymed {
    type plug
    slave.pcm "asymed"
}

#a ctl device to keep xmms happy
```

```
ctl.pasymed {
    type hw
    card 0
}

#here we try to point the aoss script to our asymed device
pcm.dsp0 {
    type plug
    slave.pcm "asymed"
}

ctl.mixer0 {
    type hw
    card 0
}

#this sets the default device
pcm.!default {
    type plug
    slave.pcm "asymed"
}
```

It's pretty much the same as described in the given link. However, a problem is to get applications to work with it that do not support ALSA.

## ARTS

The solution to the above mentioned problems is to use a sound daemon, like KDE's ARTS.

Arts needs to be started on login. You can do that for instance from your `~/.xsession` file. Just add the following line before your the where your windowmanager gets started:

```
/opt/kde/bin/artsd -b 16 -r 48000 -a alsa -D asymed &
```

## Application Setup

### Firefox

To make firefox use the ARTS daemon we use a library wrapper called `artsdsp` - so for running Firefox with ARTS just run the following command:

```
artsdsp firefox
```

To be sure it works install the Flash Plugin and try any Flash animation - like this [cute kitten](#). If you don't hear sound then something is wrong.

## XMMS

For xmms, you need to select the *aRts-Output* as output plugin in your *Preferences*.

## GAIM

Also easy - in the *Sound* preferences tab, select *Automatic* as the *Sound Method*.

## MPlayer

The [MPlayer documentation](#) says there is an arts driver. Running

```
mplayer -ao help
```

shows something like this:

```
Available audio output drivers:
mpegpes    Mpeg-PES audio output
oss        OSS/ioctl audio output
alsa9     ALSA-0.9.x audio output
esd        Esound audio output
sdl        SDLlib audio output
null       Null audio output
pcm        RAW PCM/WAVE file writer audio output
plugin     Plugin audio output
```

Now the dmix feature configured above comes in handy again. We just use the *asymed* device which is able to mix multiple sources. So it's possible to have both - ARTS with multiple Soundsources and MPlayer - accessing the same ALSA device.

Just add this to the `~/mplayer/config`

```
ao=alsa9:asymed          #or ao=alsa1x ...
srate=48000
```

From:

<https://wiki.alienbase.nl/> - Alien's Wiki

Permanent link:

<https://wiki.alienbase.nl/doku.php?id=slackware:alsa>

Last update: **2006/10/04 09:23**

