SoX - Sound eXchange

Эта программа - бесплатный конвертер различных аудиоформатов.

Конвертер форматов, простой редактор и консольный плеер в одном флаконе.

http://sox.sourceforge.net/

Краткая справка

Конвейерная обработка звука.

Автор : Адриан Чанг [Adrian J. Chung]

Перевод : С. Скороходов

С помощью программ обработки текста опытные пользователи Unix десятилетиями значительно облегчали себе редактирование документов. Такие консольные утилиты, как sed, awk, cut, paste и join, хотя и полезны каждая в отдельности, реализуют свой полный потенциал только при объединении в конвейеры с помощью программных каналов [pipelines] .

Linux теперь используется не только для работы с простым текстом в формате ASCII. Растущая популярность различных мультимедиа-форматов, содержащих изображения и аудиоданные, подстегнула разработку инструментария для работы с ними. Многие программы имеют графический интерфейс и не работают без взаимодействия с пользователем. Вместе с тем, все возрастающее число программных инструментов могут использоваться в *пакетном* режиме с отключенным пользовательским интерфейсом. Некоторые программы создаются специально для вызова из командной строки или скриптов.

Именно этот класс програмных утилит и будет рассмотрен в данной статье. Задачи сложного, комплексного преобразования медийных данных часто может быть осуществлена комбинацией простых инструментов с использованием техники, обычно применяемой в текстовых фильтрах. Мы сосредоточимся на потоковой аудио-обработке, поскольку именно в обработке этих форматов парадигма программных каналов Unix показала себя особенно успешной.

Установка

```
apt install sox
```

Использование

Конвертация аудиофайлов для Asterisk

Преобразователь аудиосэмплов

Существует множество звуковых форматов и преобразование файлов из одного формата в другой – частая операция. Утилита sox может выполнять эту роль и вызывается из комнадной строки:

```
sox sample.wav sample.aiff
```

Приведенная выше комнада преобразует WAV-файл в формат AIFF. При этом можно изменить частоту дискретизации, число битов на отсчет (8 или 16), а также число каналов:

```
sox sample.aiff -r 8000 -b -c 1 low.aiff
```

Файл low.aiff содержать один канал с 8-битным сигналом, оцифрованном на частоте 8000 в секунду.

```
sox sample.aiff -r 44100 -w -c 2 high.aiff
```

А файл high.aiff – стерео-звучание с частотой дискретизации 44100 Гц, 16-бит на отсчет.

Когда sox не может угадать выходной формат по расширению файла, его необходимо указать явно:

```
sox sample.wav -t aiff sample.000
```

Опция «-t raw» указывает на специальный «беззаголовочный» формат, содержащий только «сырые» байты данных:

```
sox sample.wav -t raw -r 11025 -sw -c 2 sample.000
```

Т.к. у файла нет заголовка, указывающего частоту дискритизации, число битов на отсчет, число каналов и т.д., будет хорошей идеей явно указать в командной строке эту информацию. При преобразовании «файла-сырца» в формате raw это также необходимо:

```
sox -t raw -r 11025 -sw -c 2 sample.000 sample.aiff
```

Нет нужды указывать опцию «-t raw», если у файла есть расширение .raw. Однако это требуется делать, если поток данных поступает со стандартного ввода или посылается в стандартный вывод. Для того, чтобы сделать это, воспользуйтесь ключом «-» на том месте в строке параметров , где должно находиться имя файла:

```
sox -t raw -r 11025 -sw -c 2 - sample.aiff < sample.raw

sox sample.aiff -t raw -r 11025 -sw -c 2 - > sample.raw
```

Зачем нам все это? Так мы сможем использовать sox как фильтр в програмном канале.

Ускоренное/замедленное воспроизведение

Обычно, благодаря использованию интерполяции sox изменяет частоту дискретизации без изменения высоты тона или темпа воспроизведения. Перенаправляя выход одного «sox'а» на вход другого и указав неравные частоты дискретизации, мы можем «обойти» интерполяцию, в результате замедляя звучание отрывка:

```
sox sample.aiff -t raw -r 44100 -sw -c 2 - |
sox -t raw -r 32000 -sw -c 2 - slow.aiff
```

или ускоряя его:

```
sox sample.aiff -t raw -r 32000 -sw -c 2 - |
sox -t raw -r 44100 -sw -c 2 - fast.aiff
```

Простейшее редактирование

Предположим, что требуется звуковой отрывок, содержащий первые две секунды другого файла. Этого можно достичь, запуская sox в «командной трубе»:

```
sox sample.aiff -t raw -r 44100 -sw -c 2 - | head -c 352800 |
sox -t raw -r 44100 -sw -c 2 - twosecs.aiff
```

Входной файл sample.aiff преобразовывается к частоте 44,1 кГц, каждые два байта в два канала. Две секунды звучания, таким образом, представлены в 44100x2x2x2 = 352800 первых байтах, которые обрезаются фильтром «head -c 352800». Затем они преобразуются обратно в формат AIFF и сохраняются в twosecs.aiff.

Точно также можно извлечь последнюю секунду «сэмпла»:

```
sox sample.aiff -t raw -r 44100 -sw -c 2 - | tail -c 176400 |
```

```
sox -t raw -r 44100 -sw -c 2 - lastsec.aiff
```

или его третью секунду:

```
sox sample.aiff -t raw -r 44100 -sw -c 2 - | tail -c +352801 |
head -c 176400 | sox -t raw -r 44100 -sw -c 2 - lastsec.aiff
```

Обратите внимание на то, что при работе с 16-битным звуком параметр, передаваемый «tail -c *+N*» должен быть нечетным, иначе «сырые» отсчеты окажутся неверно выровненными.

Можно извлечь фрагменты разных звуковых отрывков и объединить их в один файл с помощью вложенных шелл-команд:

```
(sox sample-1.aiff -t raw -r 44100 -sw -c 2 - | head -c 176400
sox sample-2.aiff -t raw -r 44100 -sw -c 2 - | head -c 176400 ) |
sox -t raw -r 44100 -sw -c 2 - newsample.aiff
```

В этом примере мы вызываем дочерний шелл, который «выдает сырые байты» из двух различных файлов в стандартный поток вывода, который в свою очередь перенаправляется процессу sox, выполняющемуся в родительском шелле и создающему результирующий файл.

Настольная система звуковоспроизведения

Опция командной строки «-t ossdsp» позволяет посылать звуки на OSS (open sound system) устройство /dev/dsp:

```
sox sample.aiff -t ossdsp /dev/dsp
```

В состав пакета sox обычно входит платформо-независимый скрипт **play**, который вызывает sox с подобающими случаю опциями. Предыдущую команду можно проще выполнить так:

```
play sample.aiff
```

Проигрываемые таким способом аудиофрагменты «монополизируют» воспроизводящую аппаратуру. Прежде, чем проиграть что-либо еще, другим «звучащим» приложениям приходится ждать, пока аудиоустройство освободится. «Рабочие столы» GNOME и KDE предоставляют возможности одновременного проигрывания более чем одного аудиофрагмента. Звуки могут «издаваться» различными программами в любой момент времени без необходимости ожидать своей очереди, однако не каждое аудио-приложение знает, как это делается в каждом из различных «рабочих столов». sox такой способностью не обладает. Но, если дать себе труд хотя бы поверхностно изучить мультимедийные звуковый службы, предоставляемые GNOME и KDE, то можно легко придумать, как обойти это ограничение.

Существует много пакетов, позволяющих совместно использовать звуковые устройства. Одна часто используемая стратегия – запуск фонового сервера, которому приложения-клиенты посылают для проигрывания звуковые фрагменты. Сервер, в свою очередь, получает в свое распоряжение звуковое устройство и направляет на него аудиоданные. В случае, когда звуки поступают от двух и более программ сразу, сервер микширует их и направляет на устройство объединенный поток.

Такой подход используется в Enlightened Sound Daemon (ESD). Пользователи GNOME часто могут обнаружить запущенный в фоновом режиме сервер esd. Пакет ESD под именем esound входит в большинство дистрибутивов и включает несколько простых клиентских приложений:

- **esdplay** – проигрыватель звуковых фрагментов, хранимых в наиболее популярныъ файловых форматах: WAV, AU или AIFF.
- **esdcat** – посылает серверу звуковые данные в «сыром» формате raw. Эта утилита – естественный «оконечкик» объединяемых в конвейер звуковых фильтров.

Вот команда, проигрывающая через ESD первую секунду звукового фрагмента:

```
sox sample.aiff -t raw -r 44100 -sw -c 2 - | head -c 176400 | esdcat
```

Тот же прием используется для воспроизведения фрагментов в формате, который не понимает ESD, но «читает» sox:

sox sample.cdr -t raw -r 44100 -sw -c 2 - | esdcat

Иногда звучание лучше, если использовать этот способ. Некоторые версии ESD вносят в записанные на низкой частоте дискретизации звуки значительные искажения и шум.

ARtS – Analog RealTime Synthsizer – схож с ESD, но чаще используется в KDE. Фоновый сервер называется artsd, а «клиентские программы сопровождения» – artsplay и artscat.

Для проигрывания звукового отрывка выполните команду:

```
sox sample.cdr -t raw -r 44100 -sw -c 2 - | tail -c 352800 |artscat
```

Ни ESD, ни ARtS не зависят от собственно окружения какого-либо «рабочего стола». В теории, приложив некоторые усилия, можно использовать ESD с KDE, а ARtS – в GNOME. Любой из серверов можно использовать и в консольной сессии. Таким образом можно микшировать все изобилие форматов как в графической среде, так и без нее.

Ссылки

http://linux-admin.net.ru/old/content/687

https://sysadminmosaic.ru/sox/sox

2017-06-25 09:03