

SFX-Formats

COLLABORATORS

	<i>TITLE :</i> SFX-Formats		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		January 8, 2025	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	SFX-Formats	1
1.1	5.4 Soundformate	1
1.2	FutureSound	1
1.3	IFF-8SVX	1
1.4	IFF-16SV	2
1.5	RAW	2
1.6	RIFF-WAV	2
1.7	SUN/NEXT	3
1.8	VOC	3

Chapter 1

SFX-Formats

1.1 5.4 Soundformate

5.4 Soundformate

SoundFX unterstützt mehrere verschiedene Dateiformate zum speichern der Sampledaten. Folgend eine Liste der eingebauten Module. Diese wird ständig erweitert.

```
FutureSound
IFF-8SVX
IFF-16SV
RAW
RIFF-WAV
SUN/NEXT
VOC
```

1.2 FutureSound

FutureSound

Das FutureSound-Format ist ein sehr altes Format mit wenig Möglichkeiten (und wenig Bedeutung). Im Prinzip ist es ein RAW-Sample mit einem winzigen Datenblock davor, indem die Länge und die Samplingrate steht.

1.3 IFF-8SVX

IFF-8SVX

Dies ist das am meisten verbreitetste Sound-Dateiformat auf dem Amiga. Es ist nach dem IFF-Standard aufgebaut und ist so leicht den eigenen Wünschen anzupassen, ohne dass die Kompatibilität beeinträchtigt wird. Dieses Format unterstützte bisher Multioktav-Samples, Mono/Stereo-samples, ein Kompressionstyp. SoundFX unterstützt auch Quadrosamples und 16-bit bzw. combined Samples. Zu den Erweiterungen...

rungen nachfolgend noch eine Beschreibung.

Den Aufbau der Combined-Samples habe ich aus der Dokumentation des Freewareprogrammes SoundBox von Richard Korbner entnommen. Dieses Format speichert die vollen 16bit-Daten eines Samples. Wenn man dieses Sample in ein herkömmliches Programm (das IFF-8SVX-Samples lädt) einlädt, so wird das Sample automatisch als 8-bit Sample geladen. Ein Programm das den Aufbau kennt, lädt es als 16bit-Sample. Der Trick besteht darin, die niederwertigen 8bit des 16bit Samplewertes in einem extra Chunk zu speichern, der die Kennung 'LBDY' hat und die höherwertigen 8-bit, wie üblich, im 'BODY'-Chunk zu speichern.

Bei Speicherung als 16-bit-Sample legt SoundFX einen BITS-Chunk an. Dieser ist wie folgt aufgebaut

```
struct chunk_bits
{
    char id[4];      /* 'BITS' */
    ULONG len;       /* 4L */
    ULONG bits;      /* derzeit wird 8/16 bit unterstützt */
};
```

Außerdem wurde der 'CHAN'-Chunk erweitert. Bei einem Datenwert von 30, handelt es sich um ein Quadrosample.

1.4 IFF-16SV

IFF-16SV

Dieses Format entnahm ich ebenfalls Richard Korbners Freewareprogrammes SoundBox. Es entspricht im Prinzip dem normalen 8SVX-Format, nur hat es die Kennung '16SV' und im 'BODY'-Chunk werden 16bit Samples gespeichert.

1.5 RAW

RAW

RAW-Sample haben eigentlich gar kein Format. Hier werden nur die "rohen" Sounddaten abgespeichert. Das hat den Vorteil, daß dieses Format sehr einfach zu handhaben ist, aber auch den Nachteil, daß keinerlei zusätzliche Daten wie Samplingrate, Loops, Bitauflösung usw. gespeichert werden können. SoundFX versucht daher zumindestens die Bitauflösung und den Vorzeichentyp zu erkennen.

1.6 RIFF-WAV

RIFF-WAV

Dieses Format wurde unter Windows auf dem PC eingeführt. Dabei hat man sich stark an den IFF-Standart angelehnt. Das WAV-Format ist auf dem PC neben dem VOC-Format, das am meisten verwendete Sounddateiformat und scheint sich dort immer mehr durchzusetzen.

1.7 SUN/NEXT

SUN/NEXT

Diese Samples stammen von SUN- und NEXT-Rechnern. Das Formate ist recht einfach aufgebaut - ein einfacher Header und dann die Sounddaten. Diese sind meistens ULAW gepackt.

1.8 VOC

VOC

Dieses Format wurde von der Firma "Creative Labs", dem Hersteller der Sound-Blaster-Karten für PC's eingeführt. Das Format ist für das direkte Abspielen der Samples von dem Datenträger ausgelegt und hat in dieser Richtung mehrere Vorteile. Allerdings ist dieses Format etwas inkonsequent geplant worden, so das einige Erweiterungen notwendig wurden, die die Handhabung des Formates sehr erschweren. Die meisten Programme können lediglich die Formatversion 1.1 lesen. SoundFX erlaubt daher das Laden und Speichern in allen gängigen Versionen des Formates.