

Spleet me up, Scotty! (SMUS) - Version 2

Anwender-Dokumentation

Ersteller: www.ansatheus.de

Datum: 03.06.2026

Version: 2.00

Inhalt

Einleitung	2
Spleeter-Installation	2
Standard mit AVX-Support	2
Ohne AVX-Support	2
Spleeter aufrufen.....	3
Spleet me up, Scotty!	5
Systemvoraussetzungen.....	5
Installation	5
Aufruf in conda	5
Funktionalitäten	6
Audiofiles auswählen.....	9
Audiofile entfernen	9
Ausgabepfad wählen	9
Stems erzeugen	9
Konfiguration	10

Einleitung

Mit **Spleeter** hat Deezer eine pythonbasierte Anwendung erstellt, die komplette Musikstücke in bis zu 5 Einzelspuren nach Instrumenten getrennt erstellt (Stems). Spleeter ist Open-Source und kann frei genutzt werden, siehe auch -> <https://github.com/deezer/spleeter> .

Desweiteren wird **Tensorflow** von Google benötigt, welches KI-Algorithmen enthält. Es ist ebenfalls frei verfügbar, siehe auch -> <https://www.tensorflow.org> .

Zu guter Letzt wird dann noch **ffmpeg** für die Audioverarbeitung benötigt, siehe auch -> <https://ffmpeg.org> .

Mit Spleet me up, Scotty! steht eine kleine, aber effektive GUI bereit, Musikstücke bequem in Stems zu separieren, siehe auch -> <http://music.ansatheus.de> .

Spleeter kann Audiofiles folgender Formate einlesen: .wav, .mp3, .ogg, .aif. Ausgabeformat der Stems ist bis dato immer .wav.

Spleeter-Installation

Die Installation erfordert etwas Geduld und Mühe. Zudem wird eine CPU benötigt, die den AVX-Befehlssatz kennt, da Tensorflow AVX benutzt, eigentlich...

...es geht aber auch ohne AVX, ist aber leider noch kniffliger.

Im Nachfolgenden wird beides beschrieben. Dennoch muss es nicht immer gleich auf Anhieb funktionieren. Nachforschen, mitdenken und ausprobieren ist dann angesagt. Klappt es dann, wird man belohnt.

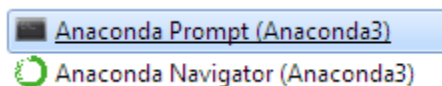
Wichtig

Es wird permanenter Internetzugriff benötigt.

Standard mit AVX-Support

Nennt man einen aktuellen Rechner, der eine AVX-CPU hat, sein eigen, ist die Installation sehr entspannt. Hier z.B. wird das ganz easy beschrieben -> <https://ezstems.com/installing-spleeter-windows> . Allerdings auch schon etwas älter – Step 5 und 7 stimmen so nicht mehr. Aber egal, wir wollen ja eh nur installieren.

1. **Anaconda-Windows-Installer** herunterladen und installieren -> <https://docs.anaconda.com/anaconda/install/windows>
2. **Anaconda-Prompt** öffnen über Windows-Startknopf->Suche->Anaconda



Hinweis: Das Prompt öffnet sich im `c:\Users\<BENUTZER>`-Verzeichnis -> ist jetzt Standardpfad

3. Im **Anaconda-Prompt** ffmpeg installieren -> `conda install -c conda-forge ffmpeg`
4. Im **Anaconda-Prompt** Spleeter installieren -> `pip install spleeter`

Wenn keine Fehler bei der Installation auftreten, dann ist alles einsatzbereit.

Ohne AVX-Support

Das ist jetzt schon eher eine Herausforderung, weil hier viel passieren kann und muß. Es wird auch ein zusätzliches Wheel-File benötigt, also ein Python-Installationspaket. Das enthält Tensorflow ohne AVX-Support. Es gibt im Netz einige so kompilierte WHL-Files. Im nachfolgenden wird Tensorflow 2.5.0 für ein Python 3.6-Environment genutzt:

URL: <https://github.com/yaroslavvb/tensorflow-community-wheels/issues/164>

Download: 2.5.0-cp36-cp36m-win_amd64.whl ->

<https://drive.google.com/file/d/1PPcIH6BD7XzbU96wpJTb9FDm7LI9NAU3/view?usp=sharing>

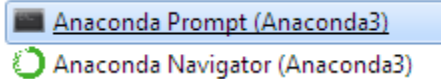
Hinweis

Mittlerweile wurde dieses Wheel-File in die Installation von Tensorflow integriert, so daß es nicht mehr separat heruntergeladen und installiert werden muß.

Als Basis dient die oben beschriebene Python/Tensorflow/ffmpeg/Spleeter-Installation mit AVX-Support. Das kann ruhig die allerneueste Version sein.

Auf geht's!

1. **Anaconda-Prompt** öffnen über Windows-Startknopf->Suche->Anaconda



2. Im **Anaconda-Prompt** ein Python-3.6-Environment mit Namen z.B. `atspleeter` erstellen:
`conda create -n atspleeter python=3.6`
3. Environment `atspleeter` aktivieren: `conda activate atspleeter`
4. Jetzt ändert sich das `(base)` vor `C:\Users\<USER>` in `(atspleeter)` -> hier erfolgen jetzt alle weiteren Installationsschritte - will man wieder zurück zu `(base)`, reicht ein `conda deactivate`
5. **ffmpeg** installieren ohne conda-forge -> `conda install -c ffmpeg`
6. **Spleeter** installieren, es wird die für Python 3.6 passende Version geholt -> `pip install spleeter`
7. Installiertes **Tensorflow** deinstallieren -> `pip uninstall tensorflow`
8. **Tensorflow** im Wheel-File installieren -> `pip install <LW:>\<PFAD>\ tensorflow-2.5.0-cp36-cp36m-win_amd64.whl` -> Dies ist in aktuellen Versionen nicht mehr nötig, weil es jetzt in Tensorflow enthalten ist.
9. Inkompatible **Numpy**-Version deinstallieren, falls Fehler -> `pip uninstall numpy` -> auch das ist nicht mehr nötig, weil die entsprechende Numpy-Version jetzt installiert wird – Punkt 10 entfällt damit ebenfalls.
10. Kompatible **Numpy**-Version installieren -> `pip install numpy=1.18.5`

```
Requirement already satisfied: oauthlib==3.0.0 in c:\programdata\anaconda3\envs\atspleeter\lib\site-packages (from requests-oauthlib==0.7.0->google-auth-oauthlib<0.5,=>0.4.1->tensorboard==2.4->tensorflow==2.5.0) (3.1.0)
Requirement already satisfied: zipp==0.5 in c:\programdata\anaconda3\envs\atspleeter\lib\site-packages (from importlib-metadata==2.6.8->tensorboard==2.4->tensorflow==2.5.0) (3.4.0)
Installing collected packages: numpy, tensorflow
  Attempting uninstall: numpy
    Found existing installation: numpy 1.18.5
    Uninstalling numpy-1.18.5:
      Successfully uninstalled numpy-1.18.5
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the source of the following dependency conflicts.
spleeter 2.1.2 requires numpy<1.19.0,>=1.16.0, but you have numpy 1.19.5 which is incompatible.
spleeter 2.1.2 requires tensorflow==2.3.0, but you have tensorflow 2.5.0 which is incompatible.
Successfully installed numpy-1.19.5 tensorflow-2.5.0
```

Wenn keine Fehler bei der Installation auftreten, dann ist auch hier alles einsatzbereit.

Spleeter aufrufen

Der Aufruf, um Spleeter zum Arbeiten zu bringen, ist relativ simpel. Dazu einfach das Anaconda-Prompt über das Windows-Startmenü öffnen. Es wird automatisch das `(base)`-Environment des angemeldeten Benutzers geöffnet. Alle benötigten Variablen zeigen dann auf dieses Verzeichnis -> `C:\Users\<BENUTZER>`. Sofern eine Spleeter-Installation mit AVX-Support ohne Fehler geglückt ist, kann in diesem Environment gearbeitet werden. Andernfalls mit `conda activate atspleeter` in das Ohne-AVX-Environment wechseln.

Dann folgende Befehlszeile eingeben:

```
spleeter separate -p spleeter:2stems -o <AUSGABEPFAD> <AUDIodatei>
```

Beispiele

mit AVX

```
(base) C:\Users\benutzer>spleeter separate -p spleeter:2stems -o
d:\spleeter\out d:\spleeter\audiofile.wav
```

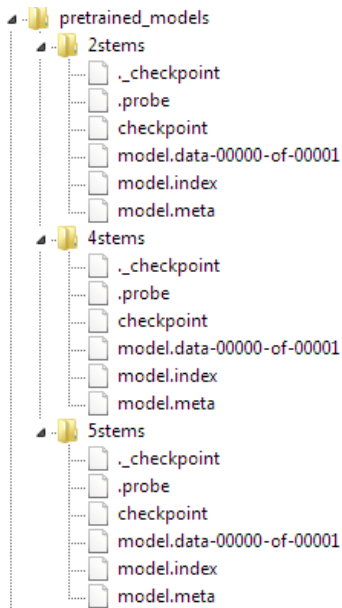
ohne AVX

```
(atspleeter) C:\Users\benutzer>spleeter separate -p spleeter:5stems -o  
d:\spleeter\out d:\spleeter\audiofile.wav
```

Es wird im Ausgabepfad ein Verzeichnis mit dem Namen der zu separierenden Audiodatei ohne Endung erzeugt und alle separierten Dateien darin abgelegt. Diese können dann in eine DAW oder Audio-Editor nach Wahl geladen und bearbeitet werden.

Hinweis 1

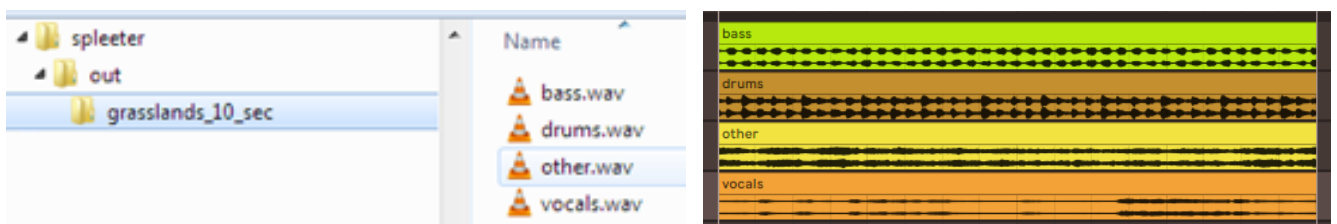
Beim Erstaufruf von Spleeter wird immer ein zur Stems-Option passendes **Pretrained Model** heruntergeladen. Es wird abgelegt unter `C:\Users\<BENUTZER>\pretrained_models`. Das Ganze sieht dann so aus:



Hinweis 2

Ein 5-Minuten-Track kann schon mal locker eine halbe Stunde in der Bearbeitung dauern. Also nicht ungeduldig werden.

Hat alles funktioniert, sind im `<AUSGABEPFAD>` alle separierten WAV-Dateien zu finden. Diese können dann in einer DAW weiterbearbeitet werden.



Optionen

Separationen

`2stems` -> vocals, accompaniment (Begleitung)

`4stems` -> vocals, drums, bass, other

`5stems` -> vocals, drums, bass, piano, other

Ein/Ausgabe

-o <AUSGABPFAD> -> Verzeichnis, in dem die separierten Audio-Dateien landen
<AUDIODEI> -> Audiodatei, die separiert werden soll

Es gibt noch weitere optionale Feinheiten, die dann in der Spleeter-Doku nachgelesen werden können.

Anmerkung:

Mittlerweile gibt es weitere Technologien zur Stem-Separation (Demucs) und Tools als Plugin (Acon) oder in DAWs (Ableton). Diese mögen und sind teilweise besser. Dennoch arbeiten viele weiter mit Spleeter, weil es einfach schneller ist. Nachdem Songs in Stems aufgeteilt werden, durchlaufen sie ohnehin verschiedene Bearbeitungs-Prozesse (EQ, Sättigung, Effekte, Mastering etc.). Dadurch gehen entweder Feinheiten verloren oder werden andere herausgestellt. Am Ende interessiert den Hörer das sowieso nicht, denn er hört das Gesamtergebnis, also den Song.

Spleet me up, Scotty!

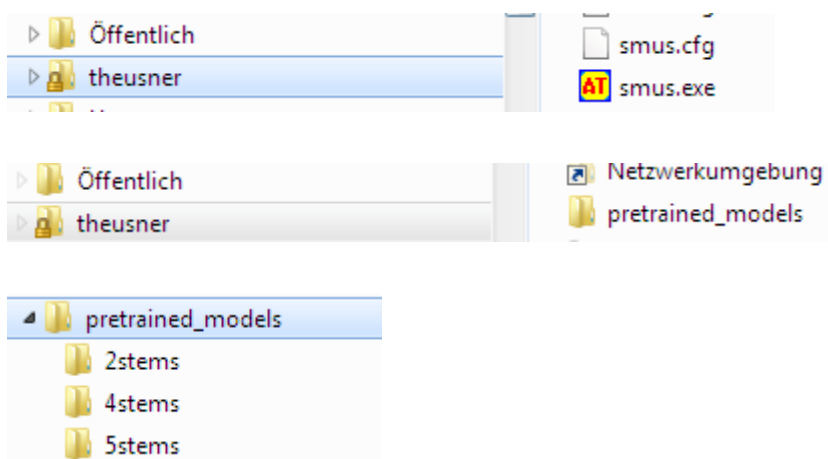
In Anlehnung an Captain Kirk's Spruch „Beam me up, Scotty!“ heisst diese kleine GUI „Spleet me up, Scotty!“ (SMUS) – ein kleines Wortspiel, was genau das Gegenteil vom Beamen in einem Stück macht -> aufsplitten.

Systemvoraussetzungen

SMUS selbst ist als 64-bit-Anwendung für Windows verfügbar und läuft ressourcensparend ab Windows XP 64bit bis zum aktuellen Windows 11 64bit.

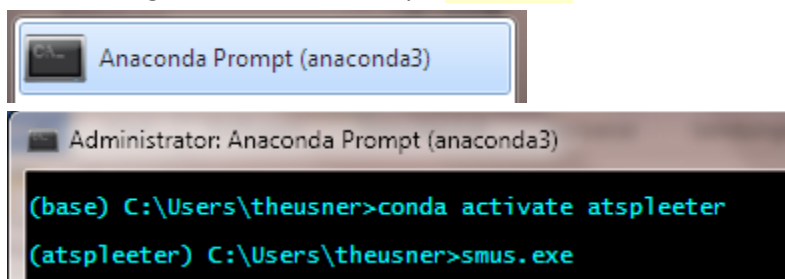
Installation

Die Installation ist absolut unkompliziert. Einfach die Dateien **smus.exe** und **smus.cfg** (sofern schon vorhanden) nach `C:\Users\<BENUTZER>` kopieren, also in den conda-Bereich. Wenn der Ordner `pretrained_models` noch nicht vorhanden ist, dann diesen Ordner ebenfalls nach `C:\Users\<BENUTZER>` kopieren (sofern er von www.ansatheus.de bzw. <http://music.ansatheus.de> heruntergeladen werden kann).

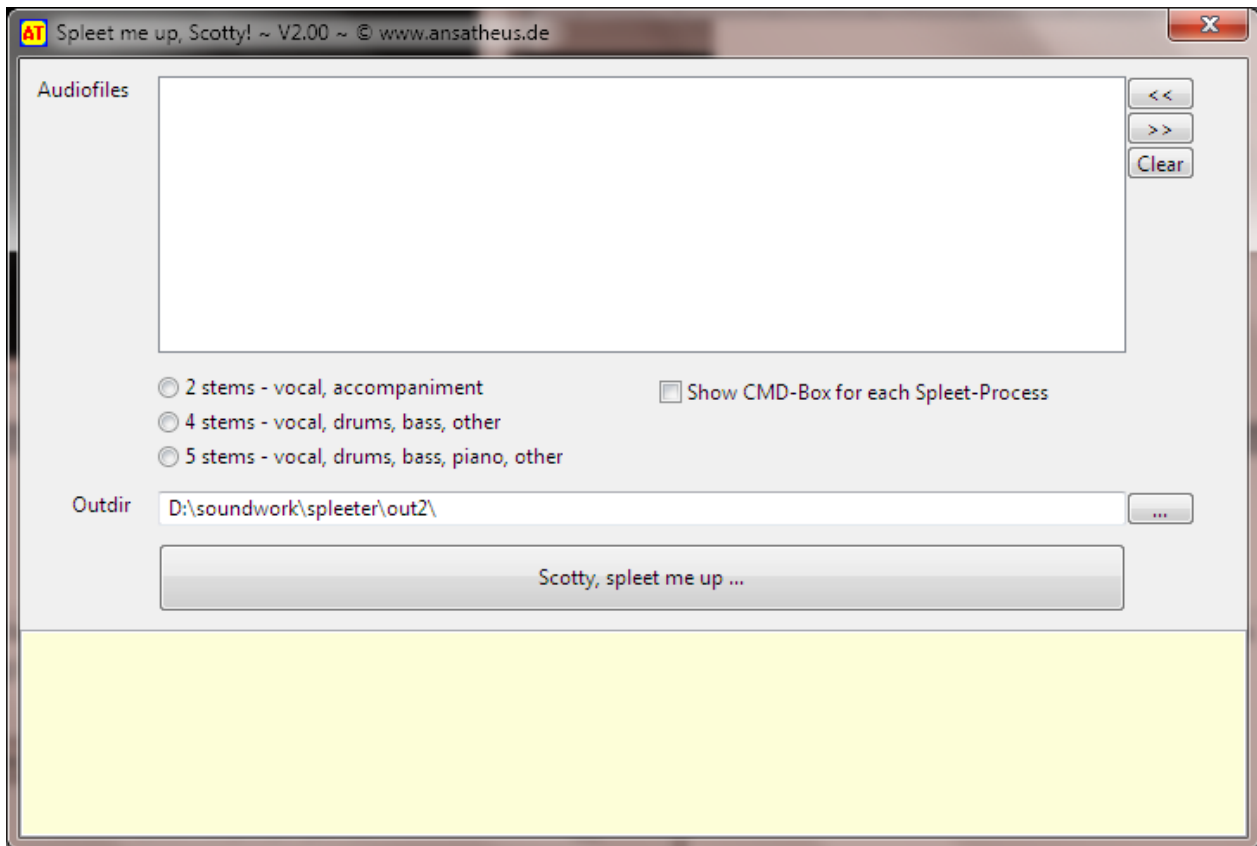


Aufruf in conda

Aufruf erfolgt im Anaconda-Prompt: `smus.exe`



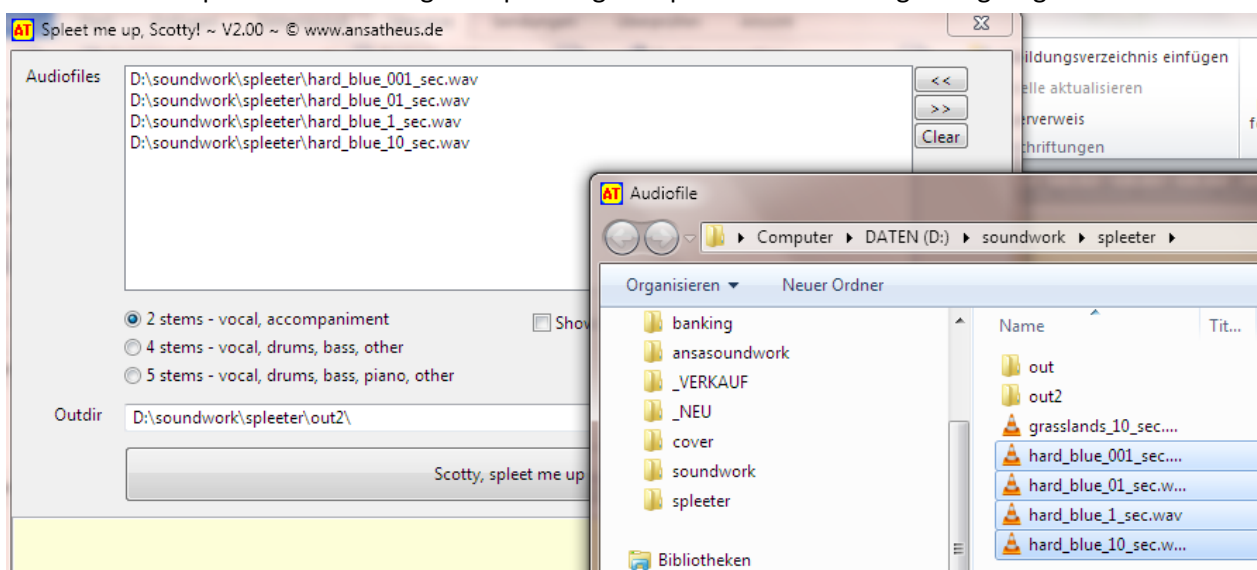
Beim Erstaufruf des Programms kann es eine Weile dauern, bis es erscheint – es initialisiert sich. Danach ist es zukünftig immer innerhalb weniger Sekundenbruchteilen einsatzbereit.

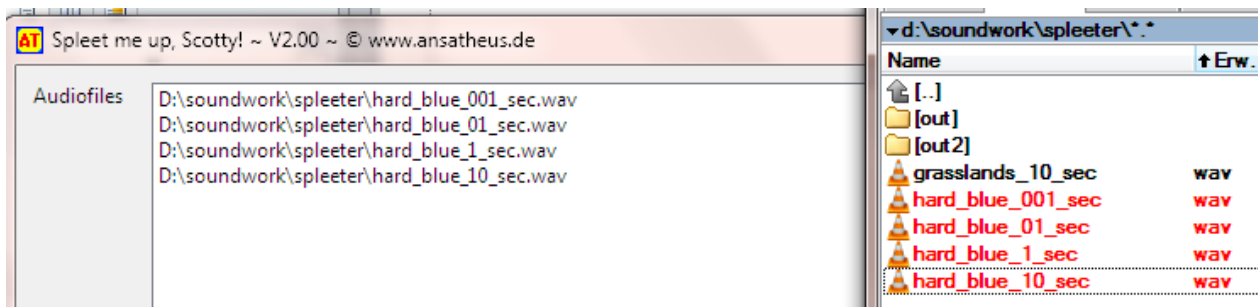


Funktionalitäten

Funktional ist das Programm auf das Wesentlichste reduziert, aber dennoch gegenüber Version 1 sinnvoll erweitert.

Dateien können per Auswahl-Dialog oder per Drag&Drop in die Bearbeitungsliste gezogen werden.

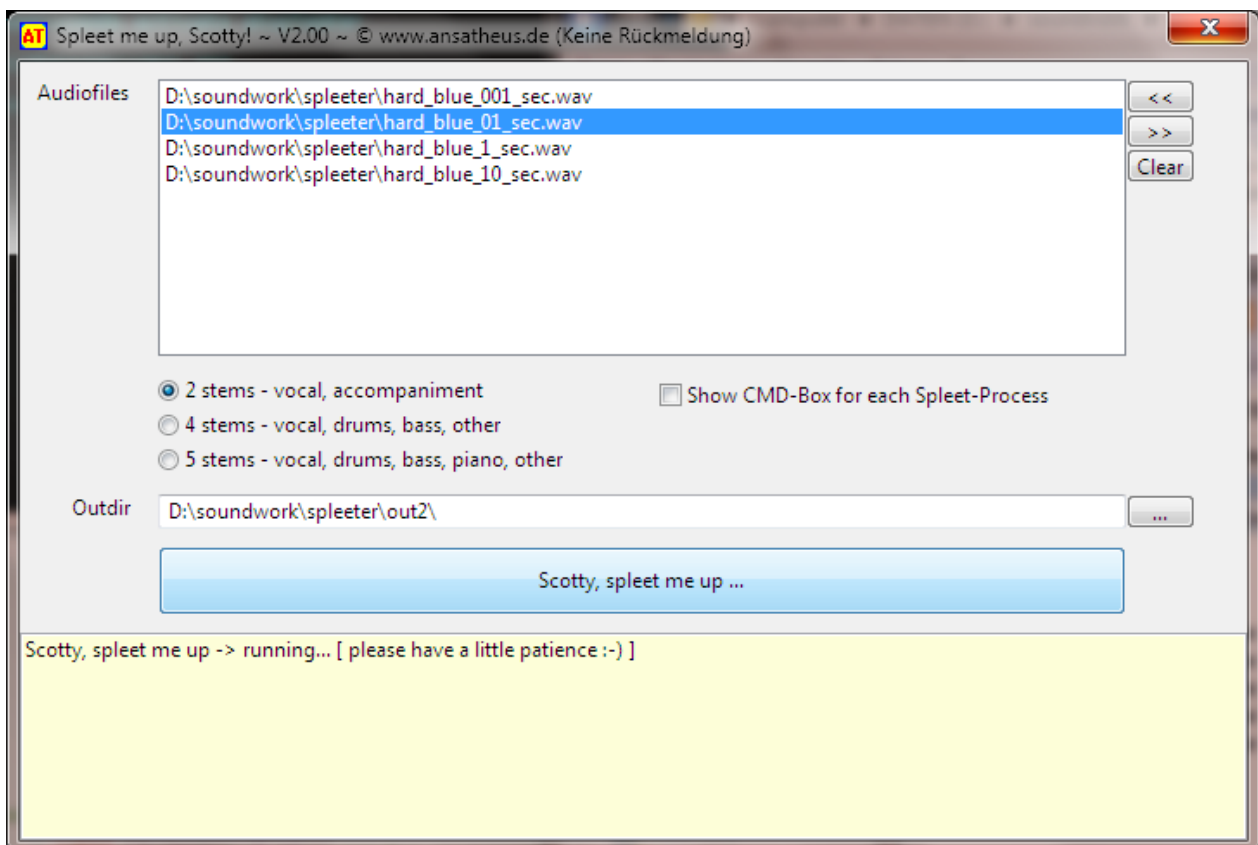


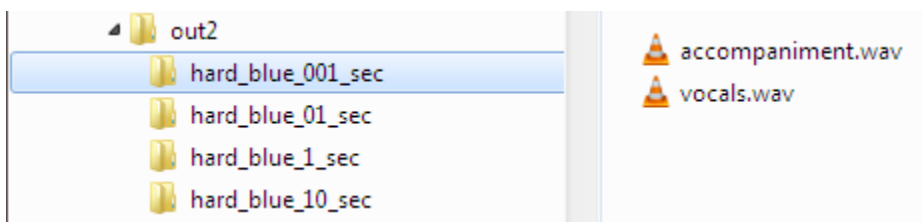
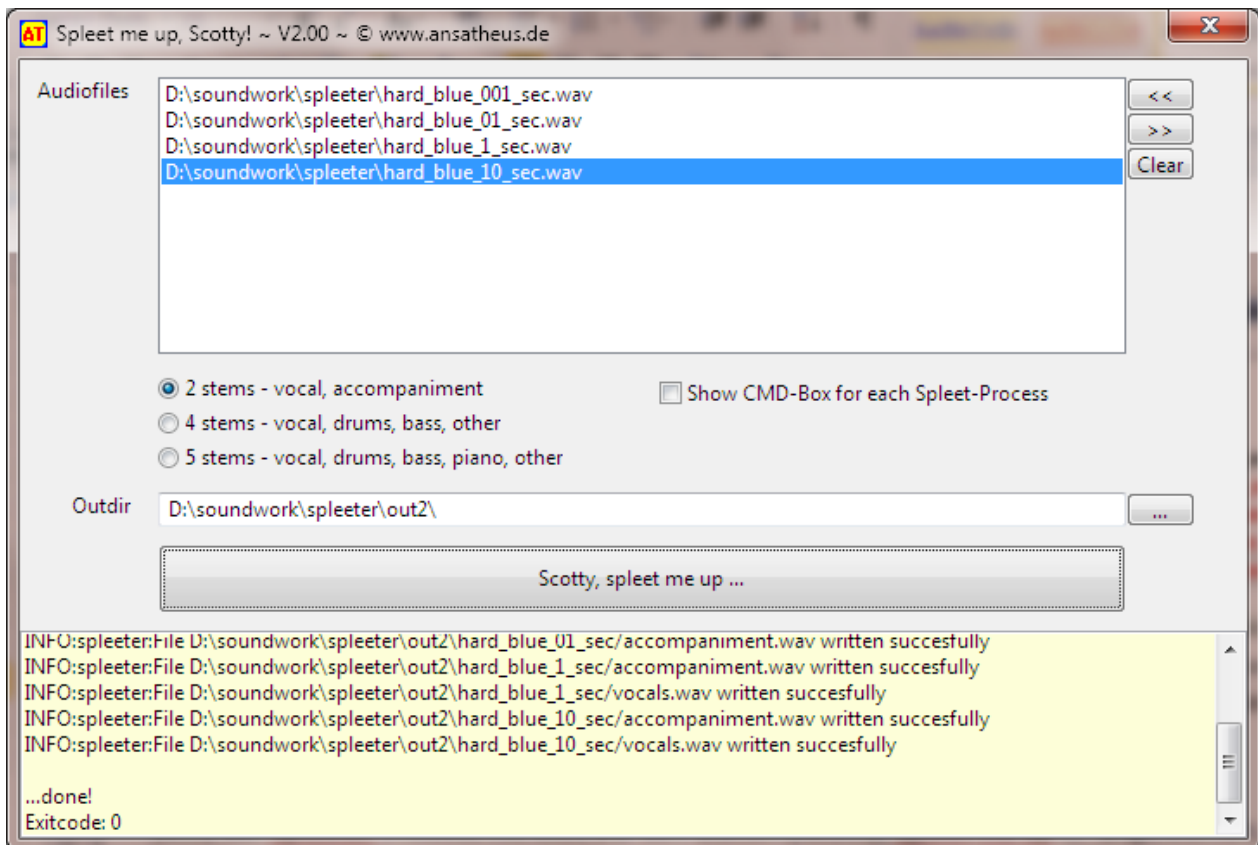


Es ist eine Mehrfachauswahl möglich, da SMUS jetzt einen eigenen „Batch“-Modus besitzt, nicht zu verwechseln mit dem Batch-Modus von Spleeter selbst. So können in einem Rutsch viele Songs separiert werden. Die jeweiligen Stems landen in Unterordner, die Spleeter pro Song anlegt. Das Programm kann dann so quasi die ganze Nacht arbeiten.

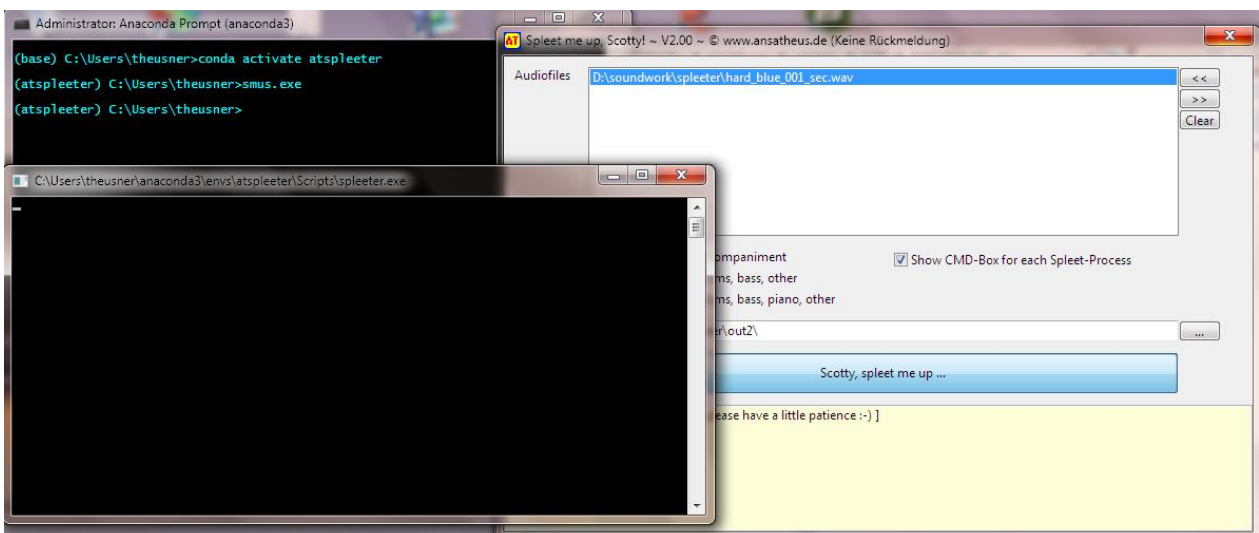
Wird der Spleeter-Prozess mit Klick auf den Button „Scotty, spleet me up ...“ angeworfen, öffnet Spleeter eine DOS-Box, sofern diese Option gewählt ist, in der es dann selbst läuft. Das ist insofern sinnvoll, das man sehen kann, ob sich etwas tut. Ist die DOS-Box versteckt, kann am markierten Eintrag in der Abarbeitungsliste gesehen werden, wo SMUS jetzt ist. Die GUI wartet den Prozess ab und bekommt dann vom Spleeter-Prozess Rückmeldung über das Endergebnis, welches im unteren gelben Log-Feld ausgegeben wird.

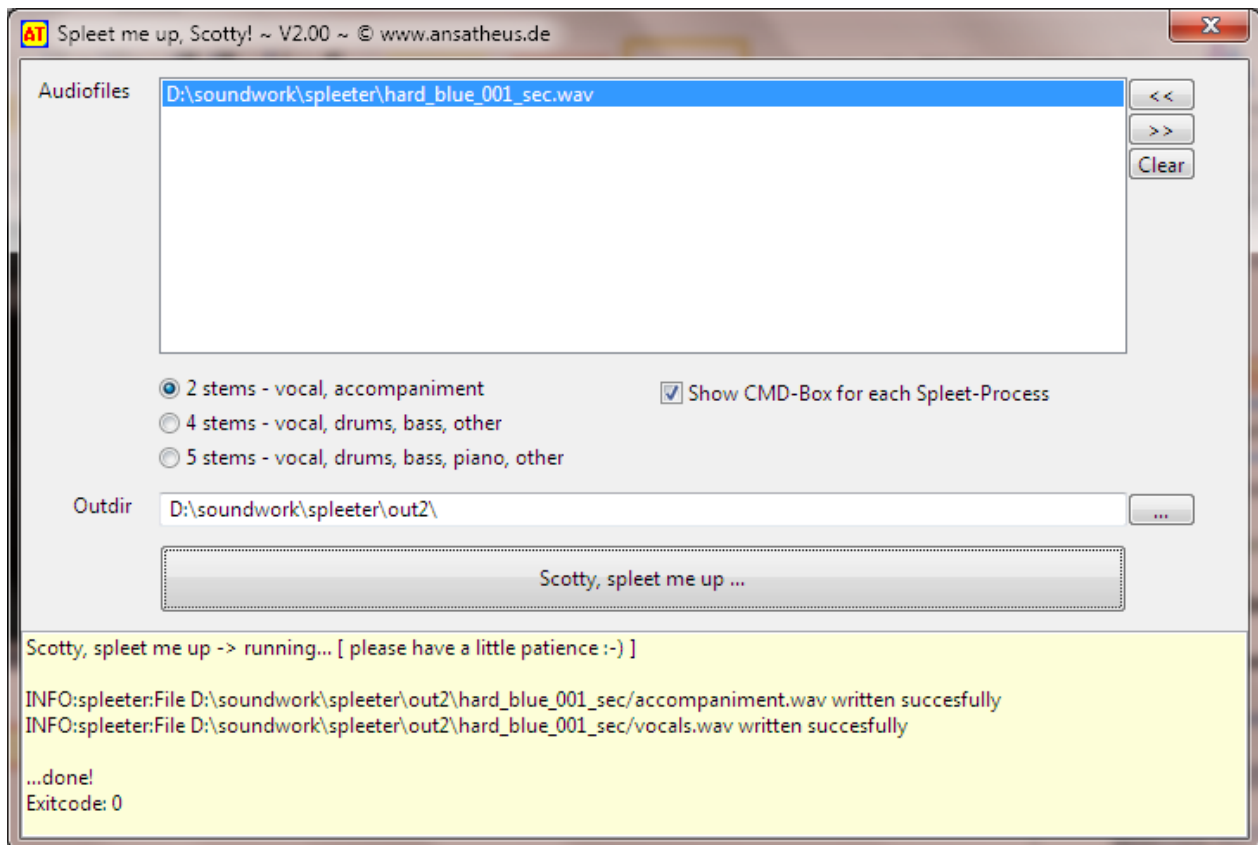
ohne DOS-Box





mit DOS-Box








Hinweis

Wie bereits weiter oben erwähnt, dauert es eine Weile, bis Spleeter die Stems separieren kann. Also nicht wundern, wenn in der Titelzeile der GUI die Meldung (keine Rückmeldung) erscheint. Einfach warten, bis Spleeter fertig ist. Ein guter Richtwert ist: ein 5-Minuten-Song kann schon mal bis zu einer halben Stunde dauern.


Audiofiles auswählen

1. Per Auswahl-Dialog -> 
2. Per Drag&Drop

Audiofile entfernen

1. Audiofiles wählen und auf Entfernen-Button klicken -> 
2. Gesamte Liste löschen über Clear-Button -> 

Ausgabepfad wählen

1. Über Pfad-Button -> 
2. Manuelle hineinkopieren

Stems erzeugen

Folgende Vorgehensweise legt die GUI nahe, um Stems zu erzeugen:

1. Audiofile/s auswählen, das separiert werden soll
2. Ausgabepfad (Outdir) einstellen, in dem die separierten Audiofiles landen sollen
3. Anzahl der Stems wählen
4. Auf Button „Scotty, spleet me up ...“ klicken
5. Warten, bis Spleeter fertig ist
6. Im Log unten wird das Bearbeitungsergebnis angezeigt

7. Hatte Spleeter Erfolg, können die Stems im Ordner, welcher den Namen des Audiofiles ohne Endung im Ausgabepfad trägt, in einer DAW oder Audio-Editor weiterbearbeitet werden

Konfiguration

Der Ausgabepfad kann manuell hineinkopiert oder per Dialog ausgewählt werden. Beim Schliessen der GUI werden Ausgabepfad und DOS-Box-Option automatisch in der Datei `smus.cfg` gespeichert und beim nächsten Start der GUI auch wieder automatisch geladen.

Sind Datei und Pfade ungültig, hagelt es Fehlermeldungen ...Stems-Optionen werden nicht gespeichert.

Das war's!

Viel Spaß beim Spleeten (armer Scotty) ! ☺