

# Manual prático do beets no Unraid

Organizar biblioteca de música, escolher entre modo online/offline, validar resultados e evitar apagar coisa errada

---

Preparado para Pablo - versão 1.0 - 2026-05-02

## Resumo honesto

O beets é excelente para organizar fisicamente uma biblioteca musical, mas não faz milagre: metadata ruim, bootlegs, edits, mixtapes e arquivos sem tags exigem escolhas. O segredo é separar duas fases: primeiro organizar com segurança; depois enriquecer metadata, capas e duplicatas.

**Regra de ouro: rode apenas um import por vez. Vários imports no mesmo banco criam corrida, pulos estranhos e retrabalho.**

# 1. Contexto deste manual

Este manual assume que o beets está rodando em Docker no Unraid, usando o container com estes caminhos internos:

Caminho no container	Uso
/config	Configuração e banco do beets. Ex.: /config/config.yaml e /config/musiclibrary.blb.
/music	Share com suas músicas originais. Cuidado: é a origem.
/music/_beets_organized	Destino organizado criado pelo beets.
/downloads	Entrada alternativa, mas neste caso você decidiu trabalhar só com /music.

Entre no shell do container antes de executar os comandos:

```
docker exec -it beets bash
```

Se o container tiver outro nome:

```
docker ps
```

Valide onde está a configuração:

```
beet config -p  
cat /config/config.yaml
```

## 2. Regras de segurança

- Nunca rode dois ou mais `beet import` ao mesmo tempo no mesmo banco.
- Nunca apague `/music` inteira. A pasta organizada também fica dentro de `/music`.
- Com `copy: yes` e `move: no`, o beets copia para o destino e preserva a origem.
- Use `--pretend` quando quiser ver o que aconteceria sem alterar nada.
- Antes de apagar originais, compare contagem, tamanho, estrutura e toque algumas faixas.

Para conferir se há import rodando:

```
ps aux | grep '[b]eet import'
```

Para parar imports presos ou duplicados:

```
ps aux | grep '[b]eet import'  
# Se ainda existir processo:  
kill -INT -f "beet import"  
# Último caso:  
kill -9 -f "beet import"
```

## 3. Config base recomendada para o seu caso

Esta configuração é conservadora: copia, não move, não busca capa/gênero automaticamente, usa uma pasta de saída separada e permite alternar entre import offline e online apenas pelo comando.

```
cat > /config/config.yaml <<'EOF'  
plugins: musicbrainz fromfilename fetchart embedart convert replaygain lastgenre web duplicates  
missing info  
  
directory: /music/_beets_organized  
library: /config/musiclibrary.blb  
  
art_filename: albumart  
threaded: yes  
original_date: no  
per_disc_numbering: no
```

```

asciify_paths: yes

ignore:
- .*
- '*~'
- System Volume Information
- lost+found
- _beets_organized
- '@eaDir'
- '#recycle'

paths:
default: $albumartist/$album%aunique{}/$track - $title
singleton: Non-Album/$artist/$title%unique{}
comp: Compilations/$album%aunique{}/$track - $title
albumtype_soundtrack: Soundtracks/$album/$track - $title

import:
write: yes
copy: yes
move: no
resume: ask
incremental: yes
group_albums: yes
quiet_fallback: asis
timid: no
log: /config/beet.log

convert:
auto: no
ffmpeg: /usr/bin/ffmpeg
opts: -ab 320k -ac 2 -ar 48000
max_bitrate: 320
threads: 1

lastgenre:
auto: no
source: album

embedart:
auto: no

fetchart:
auto: no

replaygain:
auto: no
command: mp3gain

replace:
'^\.'': _
'[\x00-\x1f]': _
'[<>:"\?*\|]': _
'[\xE8-\xEB]': e
'[\xEC-\xEF]': i
'[\xE2-\xE6]': a
'[\xF2-\xF6]': o
'[\xF8]': o
'\.$': _
'\s+$': ''

web:
host: 0.0.0.0
port: 8337
EOF

```

Depois valide:

```

beet config | head -120
mkdir -p /music/_beets_organized

```

## 4. Modos de importação: online, offline e misto

Objetivo	Comando
Simular sem alterar nada	beet import --pretend -g /music
Organizar rápido sem buscar online	beet import -A -g /music

Objetivo	Comando
Organizar faixas soltas sem buscar online	<code>beet import -A -s /music</code>
Buscar metadados online, perguntando quando duvidar	<code>beet import -g /music</code>
Buscar metadados online sem perguntar	<code>beet import -q -g /music</code>
Buscar online tratando tudo como faixa solta	<code>beet import -q -s /music</code>

Brutalmente simples: ``-A`` é velocidade e menos perfeição; sem ``-A`` é mais metadados e mais demora.

## 4.1 Modo offline/rápido

Use quando você quer organizar fisicamente agora, sem consultar MusicBrainz. Ele usa metadados já existentes nos arquivos. Se as tags estiverem vazias, o resultado pode cair em ``Unknown Artist``, ``Non-Album`` ou nomes ruins.

```
beet import -A -g /music
```

Se a biblioteca tiver muita música solta e poucas pastas de álbum:

```
beet import -A -s /music
```

## 4.2 Modo online/interativo

Use quando você quer qualidade maior. O beets consulta MusicBrainz e pede confirmação quando a correspondência não é confiável.

```
beet import -g /music
```

Atalhos na tela interativa:

Tecla	Quando usar
A / Enter	Aceitar candidato bom. Só use se parece certo.
M	Mostrar mais candidatos.
S	Pular. Bom para caso claramente errado.
U	Use as-is: importa usando as tags atuais.
T	Tracks/faixas soltas. Bom para pasta cheia de MP3 aleatórios.
G	Group albums: tentar separar vários álbuns dentro da pasta.
B	Abortar.

## 4.3 Modo online quieto

Use quando você quer deixar rodando sem ficar respondendo. Com ``quiet_fallback: asis``, o que não tiver match forte entra como está, em vez de ser pulado.

```
beet import -q -g /music
```

**Não combine ``-q`` com ``timid: yes``. O beets vai recusar com: ``error: can't be both quiet and timid``.**

## 5. Limpar e recomeçar uma tentativa

**Perigoso, mas útil. Estes comandos apagam somente o resultado organizado e o banco do beets. Não apagam seus originais em ``/music``.**

```

pkill -INT -f "beet import"
sleep 20
ps aux | grep '[b]eet import'

rm -rf -- /music/_beets_organized
rm -f -- /config/musiclibrary.blb
rm -f -- /config/musiclibrary.blb-shm
rm -f -- /config/musiclibrary.blb-wal
rm -f -- /config/beet.log
mkdir -p /music/_beets_organized

find /music/_beets_organized -type f | wc -l

```

Se aparecer a pergunta abaixo ao iniciar novo import:

Import of the directory /music was interrupted. Resume (Y/n)?

Para recomeçar limpo, responda:

n

## 6. Monitoramento durante importação

Em outro terminal, entre no container e acompanhe sem iniciar outro import.

```
docker exec -it beets bash
```

Comandos úteis:

```

ps aux | grep '[b]eet import'
find /music/_beets_organized -type f | wc -l
du -sh /music/_beets_organized
tail -50 /config/beet.log
tail -f /config/beet.log

```

Comparar origem e destino:

```

# Origem, excluindo a pasta organizada
find /music -path /music/_beets_organized -prune -o -type f \
  \( -iname '*.mp3' -o -iname '*.flac' -o -iname '*.m4a' -o \
    -iname '*.ogg' -o -iname '*.wav' -o -iname '*.aac' -o \
    -iname '*.opus' \) -print | wc -l

# Destino organizado
find /music/_beets_organized -type f \
  \( -iname '*.mp3' -o -iname '*.flac' -o -iname '*.m4a' -o \
    -iname '*.ogg' -o -iname '*.wav' -o -iname '*.aac' -o \
    -iname '*.opus' \) -print | wc -l

```

## 7. Depois que terminar

Quando o prompt voltar, rode estes checks:

```

ps aux | grep '[b]eet import'
beet stats
find /music/_beets_organized -type f | head -100
grep -iE "error|failed|skipping|already in the library|traceback" /config/beet.log | tail -100

```

A estrutura esperada é algo parecido com:

```

/music/_beets_organized/
  Artista/
    Album/
      01 - Musica.flac
  Compilations/
    Album/
      01 - Musica.mp3
  Non-Album/
    Artista/
      Musica.mp3

```

**Se o destino tiver muito menos arquivos que a origem, não finja que está tudo certo. Pode ter ocorrido skip por metadata ruim, duplicata, processo paralelo, banco antigo ou import interrompido.**

## 8. Buscar, listar e auditar a biblioteca

Listar faixas:

```
beet ls
beet ls aphex
beet ls artist:'Aphex Twin'
beet ls year:1994
```

Listar álbuns:

```
beet ls -a
beet ls -a albumartist:'Aphex Twin'
```

Mostrar caminhos dos arquivos:

```
beet ls -p artist:'Aphex Twin'
beet ls -af '$albumartist - $album: $albumtotal' aphex
```

Estatísticas:

```
beet stats
beet stats -e
```

Campos disponíveis para consultas e path formats:

```
beet fields
```

## 9. Capas, embeds, gêneros e online extras

A config base deixa essas tarefas desligadas durante a importação para não transformar uma organização grande em sofrimento. Depois que tudo estiver organizado, rode manualmente.

### Capas

```
# Buscar capas para todos os álbuns que ainda não têm capa
beet fetchart -a
```

```
# Forçar nova busca de capas
beet fetchart -f -a
```

### Embutir capas nos arquivos

```
# Embutir a capa existente nos arquivos dos álbuns
beet embedart -a
```

```
# Embutir uma imagem específica em uma consulta
beet embedart -f /caminho/capa.jpg album:'Nome do Album'
```

### Gêneros

Se quiser usar gêneros depois, ligue ``lastgenre.auto: yes`` temporariamente ou rode comandos específicos do plugin, conforme sua instalação. Gênero automático é opinativo; espere sujeira.

## 10. Duplicatas e álbuns incompletos

Encontrar duplicatas:

```
beet duplicates
beet duplicates -p
beet duplicates -a
beet duplicates -c
beet duplicates -k title -k artist
```

**Não use ``beet duplicates --delete`` no impulso. Primeiro liste com ``-p``, confira caminhos, e só delete quando tiver certeza.**

Ver álbuns com faixas faltando:

```
beet missing
beet missing -c
beet missing -t
```

```
beet missing -a
```

Observação: `missing` pode fazer chamadas online para a fonte de dados do álbum.

## 11. Retag, update, write e move

### Retag online depois de importar as-is

Se você importou rápido com `-A`, pode corrigir metadados depois em lotes menores. Faça por artista/álbum, não na biblioteca inteira de uma vez.

```
# Retag de uma consulta existente na biblioteca
beet import -L artist:'Aphex Twin'
```

```
# Retag de tudo: possível, mas pesado e arriscado
beet import -L
```

### Atualizar banco a partir dos arquivos

```
# Simular atualização
beet update -p
```

```
# Atualizar de verdade
beet update
```

### Escrever metadados do banco nos arquivos

```
# Simular
beet write -p
```

```
# Escrever tags
beet write
```

### Mover/copiar conforme path format

```
# Dry-run: mostra o que moveria
beet move -p
```

```
# Aplicar organização conforme paths
beet move
```

```
# Copiar para outro destino sem alterar banco
beet move -e -c -d /algum/destino artist:'Aphex Twin'
```

## 12. Apagar originais depois de validar

**Sem backup, apagar originais é decisão sem volta. Faça isso só depois de usar a pasta organizada por alguns dias e conferir artistas importantes, raridades, mixtapes e arquivos obscuros.**

Primeiro ensaie o que seria removido. Este comando apenas lista as entradas antigas de primeiro nível, preservando `\_beets\_organized`:

```
find /music -mindepth 1 -maxdepth 1 ! -name '_beets_organized' -print
```

Se a lista estiver certa, o comando perigoso seria:

```
find /music -mindepth 1 -maxdepth 1 ! -name '_beets_organized' -exec rm -rf -- {} +
```

**Repito porque vale dinheiro e sanidade: esse `rm` não tem lixeira. Erro aqui é perder coleção.**

## 13. Troubleshooting rápido

Sintoma	Causa provável e ação
can't be both quiet and timid	Você usou `-q` com `timid: yes`. Coloque `timid: no` ou não use `-q`.

Sintoma	Causa provável e ação
This album is already in the library!	Pode ser importação anterior, incremental, banco antigo ou processos concorrentes. Confira <code>`ps aux   grep '[b]eet import`</code> .
Resume (Y/n)?	Se quer continuar a importação interrompida, `Y`. Se quer recomeçar limpo, `n`.
No matching recordings found	Metadata ruim ou item obscuro. Use `U` para as-is, `S` para pular, ou tente retag depois.
MusicBrainz 400 query= vazio	Tags vazias. `fromfilename` ajuda quando há nomes úteis, mas não faz milagre.
Muito lento	Normal com MusicBrainz. Use `-A` para não buscar online, ou rode lotes menores.
Quero acelerar com 4 imports	Não faça. Um banco, um import. Paralelizar aqui é receita para corrupção lógica.

## 14. Receitas prontas

### Organização rápida, sem online

```

pkill -INT -f "beet import"
rm -rf -- /music/_beets_organized
rm -f -- /config/musiclibrary.blb /config/musiclibrary.blb-shm /config/musiclibrary.blb-wal
rm -f -- /config/beet.log
mkdir -p /music/_beets_organized
beet import -A -g /music

```

### Organização com busca online, sem perguntas

```

# Requer timid: no e quiet_fallback: asis
beet import -q -g /music

```

### Organização com busca online, decidindo manualmente

```

beet import -g /music

```

### Segunda passada para faixas soltas

```

# Offline/as-is
beet import -A -s /music

# Online/quieto
beet import -q -s /music

```

### Auditoria final

```

beet stats
beet duplicates -p
beet missing -t
find /music/_beets_organized -type f | wc -l
du -sh /music/_beets_organized

```

## 15. Fontes consultadas

Documentação oficial do beets e MusicBrainz usada para este manual:

- beets Command-Line Interface: <https://docs.beets.io/en/stable/reference/cli.html>
- beets MusicBrainz Plugin: <https://docs.beets.io/en/stable/plugins/musicbrainz.html>
- MusicBrainz API rate limiting: [https://musicbrainz.org/doc/MusicBrainz\\_API](https://musicbrainz.org/doc/MusicBrainz_API)
- beets FromFilename Plugin: <https://docs.beets.io/en/stable/plugins/fromfilename.html>
- beets FetchArt Plugin: <https://docs.beets.io/en/stable/plugins/fetchart.html>
- beets EmbedArt Plugin: <https://docs.beets.io/en/stable/plugins/embedart.html>
- beets Duplicates Plugin: <https://docs.beets.io/en/stable/plugins/duplicates.html>
- beets Missing Plugin: <https://docs.beets.io/en/stable/plugins/missing.html>