

# YouTube to Plex pipeline

Requirements:

TubeSync - <https://github.com/meeb/tubesync>

TubeSync NFO Agent - <https://github.com/go2tom42/TubeSync-NFO-Agent.bundle>

Plex - <https://www.plex.tv/>

I run everything through Docker using **caddy-docker-proxy** for ssl, this is my config for TubeSync, drop the **network** & **labels** sections if you run a different proxy-

```
tubesync:
  image: ghcr.io/meeb/tubesync:latest
  container_name: tubesync
  restart: unless-stopped
  ports:
    - 4848:4848
  volumes:
    - /docker/configs/tubesync/models.py:/app/sync/models.py #NOT Required
    - /docker/configs/tubesync/settings.py:/app/tubesync/settings.py #NOT Required
    - /docker/configs/tubesync:/config
    - /nas/Videos/YouTube:/downloads/video
  environment:
    - TUBESYNC_WORKERS=8
    - TZ=Europe/London
    - PUID=1000
    - PGID=1000
  networks:
    - caddy
  labels:
    caddy: tubesync.XXXXX.pw
    caddy.reverse_proxy: "{{upstreams 4848}}"
    caddy.tls: "internal"
```

The two "NOT Required" entries are because I wanted larger thumbnails and to change the default settings for new video sources

For the models.py file I changed. Currently lines 237-300 in hash 5bf53b3 (current as of 3-14-22)

```
prefer_60fps = models.BooleanField(
    _('prefer 60fps'),
    default=False,
    help_text=_('Where possible, prefer 60fps media for this source')
)
prefer_hdr = models.BooleanField(
    _('prefer hdr'),
    default=True,
    help_text=_('Where possible, prefer HDR media for this source')
)
fallback = models.CharField(
    _('fallback'),
    max_length=1,
    db_index=True,
    choices=FALLBACK_CHOICES,
    default=FALLBACK_NEXT_BEST,
    help_text=_('What do do when media in your source resolution and codecs is not
available')
)
copy_thumbnails = models.BooleanField(
    _('copy thumbnails'),
    default=True,
    help_text=_('Copy thumbnails with the media, these may be detected and used by some
media servers')
)
write_nfo = models.BooleanField(
    _('write nfo'),
    default=True,
    help_text=_('Write an NFO file in XML with the media info, these may be detected and
used by some media servers')
)
```

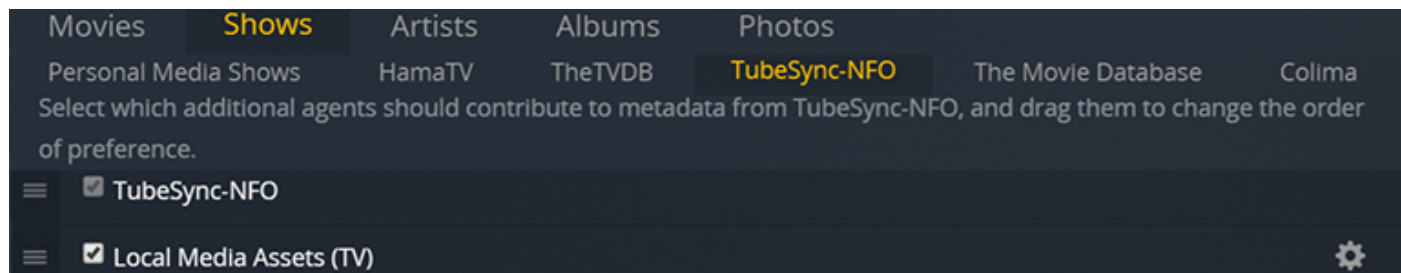
For settings.py I changed. Currently lines 148-149 in hash 5bf53b3 (current as of 3-14-22)

```
MEDIA_THUMBNAIL_WIDTH = 1280          # Width in pixels to resize thumbnails to
MEDIA_THUMBNAIL_HEIGHT = 720         # Height in pixels to resize thumbnails to
```

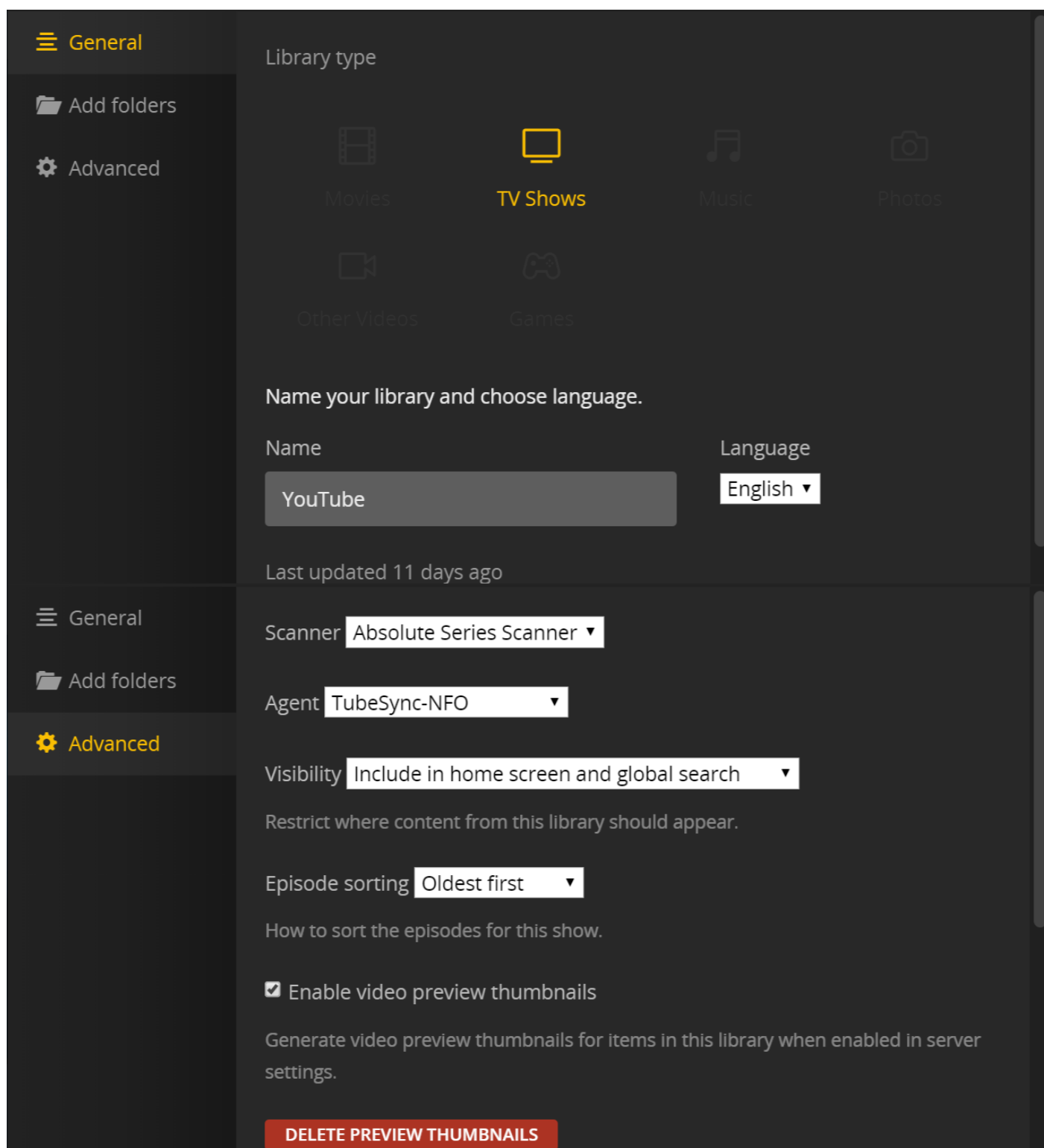
AND Currently line167 in hash 5bf53b3 (current as of 3-14-22)

```
MEDIA_FORMATSTR_DEFAULT = '{source_full} - {yyyy_mm_dd} - [{key}] - {resolution}.{ext}'
```

But like I said not required. Make sure you enable `Local Media Assets (TV)` in the plex server agents settings for [TubeSync NFO Agent](#)



My Plex library looks like



☰ General

📁 Add folders

⚙️ Advanced

☒ Enable video preview thumbnails

Generate video preview thumbnails for items in this library when enabled in server settings.

DELETE PREVIEW THUMBNAILS

Collections Disabled ▼

How to display collections.

Seasons Hide for single-season series ▼

Choose whether to display seasons.

☐ Enable intro detection

Generate intro detection for items in this library when enabled in server settings.

A "Source" in [TubeSync](#) looks like using Playlist URL [https://www.youtube.com/playlist?list=PL0-LSnSBNInfr7MDRuXmfk1LJ\\_KjvaXt](https://www.youtube.com/playlist?list=PL0-LSnSBNInfr7MDRuXmfk1LJ_KjvaXt)

SOURCE TYPE:

YouTube playlist

 Source type


KEY:

PL0-LSnSBNInfr7MDRuXmfk1LJ\_KjvaXt

 Source key, such as exact YouTube channel name or playlist ID

NAME:

Trek, Actually

 Friendly name for the source, used locally in TubeSync only

DIRECTORY:

Trek, Actually

 Directory name to save the media into

MEDIA FORMAT:

{source\_full} - {yyyy\_mm\_dd} - [{key}] - {resolution}.{ext}


 File format to use for saving files, detailed options at bottom of page.

INDEX SCHEDULE:

Every 24 hours

 Schedule of how often to index the source for new media

 **DOWNLOAD MEDIA**

 Download media from this source, if not selected the source will only be indexed

DOWNLOAD CAP:

No cap

 Do not download media older than this capped date

☐ **DELETE OLD MEDIA**

 Delete old media after "days to keep" days?

DAYS TO KEEP:

14

 If "delete old media" is ticked, the number of days after which to automatically