immich

https://docs.immich.app/install/docker-compose/

docker-compose.yml

```
# https://docs.immich.app/install/docker-compose/

#
# WARNING: To install Immich, follow our guide: https://docs.immich.app/install/docker-compose
#
# Make sure to use the docker-compose.yml of the current release:
#
# https://github.com/immich-app/immich/releases/latest/download/docker-compose.yml
#
# The compose file on main may not be compatible with the latest release.

name: immich

services:
  immich-server:
    container_name: immich_server
    image: ghcr.io/immich-app/immich-server:${IMMICH_VERSION:-release}
    # extends:
    #   file: hwaccel.transcoding.yml
    #   service: cpu # set to one of [nvenc, quicksync, rkmpp, vaapi, vaapi-wsl] for
accelerated transcoding
    volumes:
      - upload:/data
      - /etc/localtime:/etc/localtime:ro
    env_file:
      - stack.env
    ports:
      - '2283:2283'
    depends_on:
      - redis
      - database
    restart: unless-stopped
```

```
    healthcheck:
      disable: false

  immich-machine-learning:
    container_name: immich_machine_learning
    # For hardware acceleration, add one of -[armnn, cuda, rocm, openvino, rknn] to the image
tag.
    # Example tag: ${IMMICH_VERSION:-release}-cuda
    image: ghcr.io/immich-app/immich-machine-learning:${IMMICH_VERSION:-release}
    # extends: # uncomment this section for hardware acceleration - see
https://docs.immich.app/features/ml-hardware-acceleration
    #   file: hwaccel.ml.yml
    #   service: cpu # set to one of [armnn, cuda, rocm, openvino, openvino-wsl, rknn] for
accelerated inference - use the `-wsl` version for WSL2 where applicable
    volumes:
      - model-cache:/cache
    env_file:
      - stack.env
    restart: unless-stopped
    healthcheck:
      disable: false

  redis:
    container_name: immich_redis
    image:
docker.io/valkey/valkey:8@sha256:81db6d39e1bba3b3ff32bd3a1b19a6d69690f94a3954ec131277b9a26b95b
3aa
    healthcheck:
      test: redis-cli ping || exit 1
    restart: unless-stopped

  database:
    container_name: immich_postgres
    image: ghcr.io/immich-app/postgres:14-vectorchord0.4.3-
pgvectors0.2.0@sha256:bcf63357191b76a916ae5eb93464d65c07511da41e3bf7a8416db519b40b1c23
    env_file:
      - stack.env
    environment:
      POSTGRES_PASSWORD: ${DB_PASSWORD}
      POSTGRES_USER: ${DB_USERNAME}
```

```
      POSTGRES_DB: ${DB_DATABASE_NAME}
      POSTGRES_INITDB_ARGS: '--data-checksums'
      # Uncomment the DB_STORAGE_TYPE: 'HDD' var if your database isn't stored on SSDs
      # DB_STORAGE_TYPE: 'HDD'
    volumes:
      - db:/var/lib/postgresql/data
    shm_size: 128mb
    restart: unless-stopped

volumes:
  model-cache:
  db:
  upload:
```

.env

```
IMMICH_VERSION=v2
DB_PASSWORD=postgres
DB_USERNAME=postgres
DB_DATABASE_NAME=immich
```

Upload mit immich-go

https://github.com/simulot/immich-go

https://github.com/simulot/immich-go/blob/main/docs/best-practices.md

lokale Ordner Upload

```
.\immich-go upload from-folder --server=https://immich.MEINEDOMAIN.de --api-key=gHwcJ.......jE
--concurrent-tasks=4 --client-timeout=60m --pause-immich-jobs=false --on-errors=continue --
manage-burst=Stack --manage-raw-jpeg=StackCoverJPG --manage-heic-jpeg=StackCoverJPG --exclude-
extensions="*.psd" --session-tag --folder-as-album=FOLDER --folder-as-tags --tag="Import 2026-
```

```
Jan" 'D:\Bilder\anderes\Ford Fiesta\'
```

Google Photos Upload

zuerst Takeout herunterladen: https://takeout.google.com/

dann

```
.\immich-go upload from-google-photos --server=https://immich.MEINEDOMAIN.de --api-key=gHwc........jE --concurrent-tasks=4 --client-timeout=60m --pause-immich-jobs=true --on-errors=continue --manage-burst=Stack --manage-raw-jpeg=StackCoverJPG --manage-heic-jpeg=StackCoverJPG --session-tag E:\googlephotosbackup\takeout-20251202T213051Z-3-*.zip
```

Revision #3
Created 2025-12-02 17:25:43 UTC
Updated 2026-01-01 16:48:50 UTC