

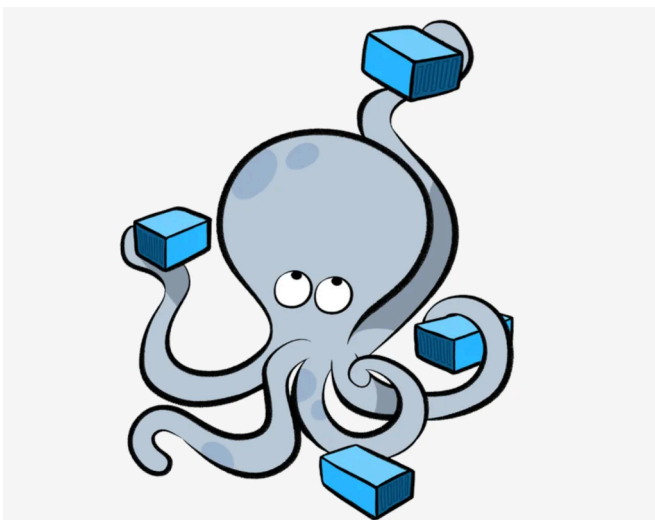
Docker Compose

Ryan Tolboom

New Jersey Institute of Technology



Goals



[Docker Compose logo](#) is used under fair use

- Run multiple containers
- Have containers communicate with each other
- Easily to bring a whole system up and down

The Good

- Already included with Docker Desktop
- Easy to run
- Can be used on dev machines



[“thumbs up”](#) by [anthony kelly](#) is licensed under [CC BY 2.0](#)

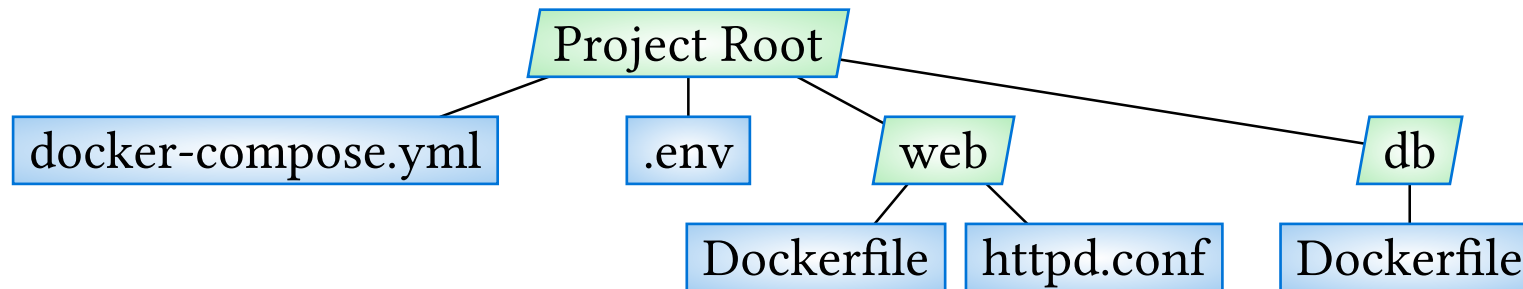
The Bad



[“2 thumbs down”](#) by [sparklemotion0](#) is licensed under [CC BY-NC-SA 2.0](#)

- Strange persistence choices
- Difficult to run on multiple machines
- Not considered production stable
- Path name issues

Setup



- `docker-compose.yml` in the root of the project
- `.env` file will be loaded, environment variables can be used
- directories are used for individual parts of system

What goes in the docker-
compose.yml?

version

info only, not required
or recommended

services

containers that will be
run, with options

networks

individual networks to
be created, if omitted
one network linking all
services will be created

volumes

persistent data stores

configs

volumes for config files

secrets

configs for sensitive data

Services

- largest part of a docker-compose.yml file
- most options in the compose file spec
- services can be resolved by their name internally

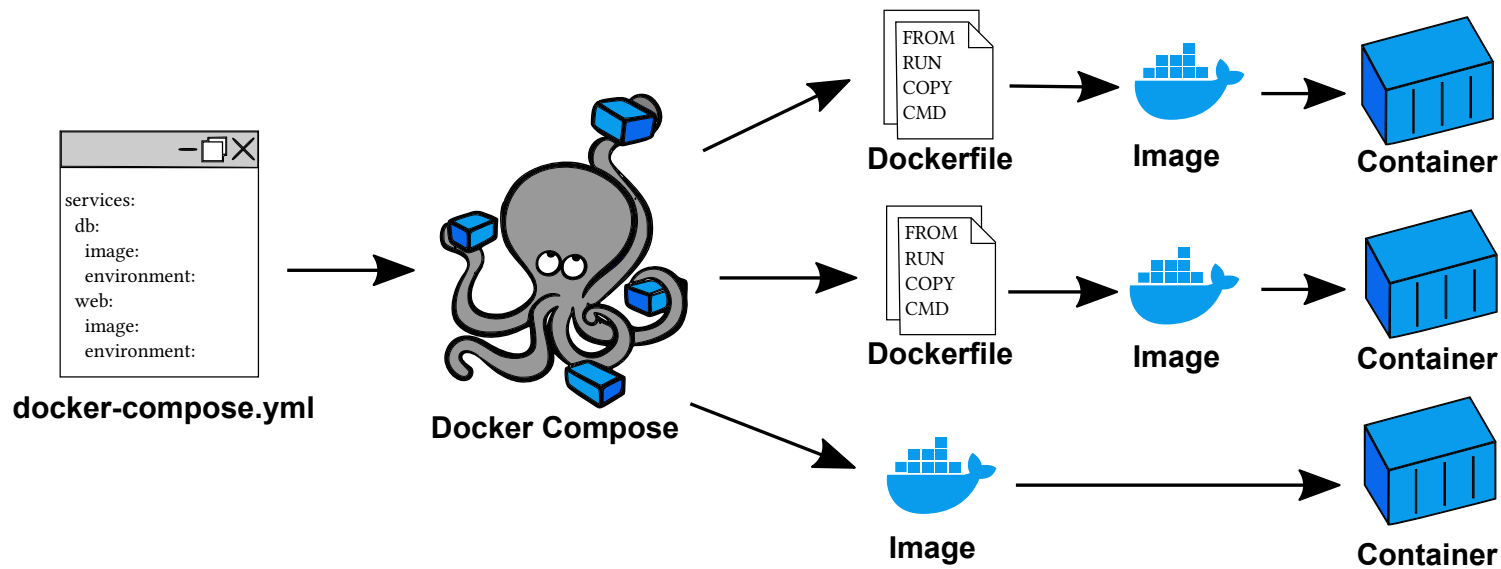
Service Example

```
pihole:
  container_name: pihole
  image: pihole/pihole:latest
  ports:
    - "53:53/tcp"
    - "53:53/udp"
    - "67:67/udp"
    - "8080:80/tcp"
    - "8443:443/tcp"
  environment:
    - TZ=${TIMEZONE}
    - PIHOLE_DNS_=172.20.0.2#5054;1.1.1.1 # referencing by name results in "Invalid IP detected in PIHOLE_DNS_: cloudflared#5054"
    - WEBPASSWORD=${PIHOLE_PW}
    - REV_SERVER=true
    - REV_SERVER_TARGET=${PIHOLE_ROUTER_IP}
    - REV_SERVER_DOMAIN=${PIHOLE_NETWORK_DOMAIN}
    - REV_SERVER_CIDR=${PIHOLE_REVERSE_DNS}
    - ServerIP=${PIHOLE_HOST_IP}
    - ServerIPv6=${PIHOLE_HOST_IPV6}
  #dns:
    #- 127.0.0.1 # "Sets your container's resolve settings to localhost so it can resolve DHCP hostnames [...]" - github.com/pi-hole/docker-pi-hole
    #- 1.1.1.1 # Backup server
  volumes: # store your data between container upgrades
    - "/etc/pihole/:/etc/pihole/"
    - "/etc/dnsmasq.d/:/etc/dnsmasq.d/"
  cap_add:
    - NET_ADMIN # Recommended but not required (DHCP needs NET_ADMIN) https://github.com/pi-hole/docker-pi-hole#note-on-capabilities
  depends_on:
    - "cloudflared"
  restart: always
  networks:
    - dns-net
```

Running

- similar options to `docker` command but with a few things fixed
- `docker-compose.yml` expected to be in the direction where you are running `docker compose`
- `docker compose up` brings everything up in the foreground
- `docker compose down` (Ctrl-C if running) brings things down (gracefully, hopefully)
- `docker compose build` builds all custom Docker images, don't forget!
- `docker compose exec <service>` run something on a running service
- `docker compose run <service>` run a running container

Resources



[“compose-diagram.svg”](#) by [Ryan Tolboom](#) is licensed under [CC BY-NC 4.0](#)

- [Full Compose File Spec](#)
- [Docker Compose CLI reference](#)
- [Awesome Compose](#) (curated list of cool Docker Compose examples)