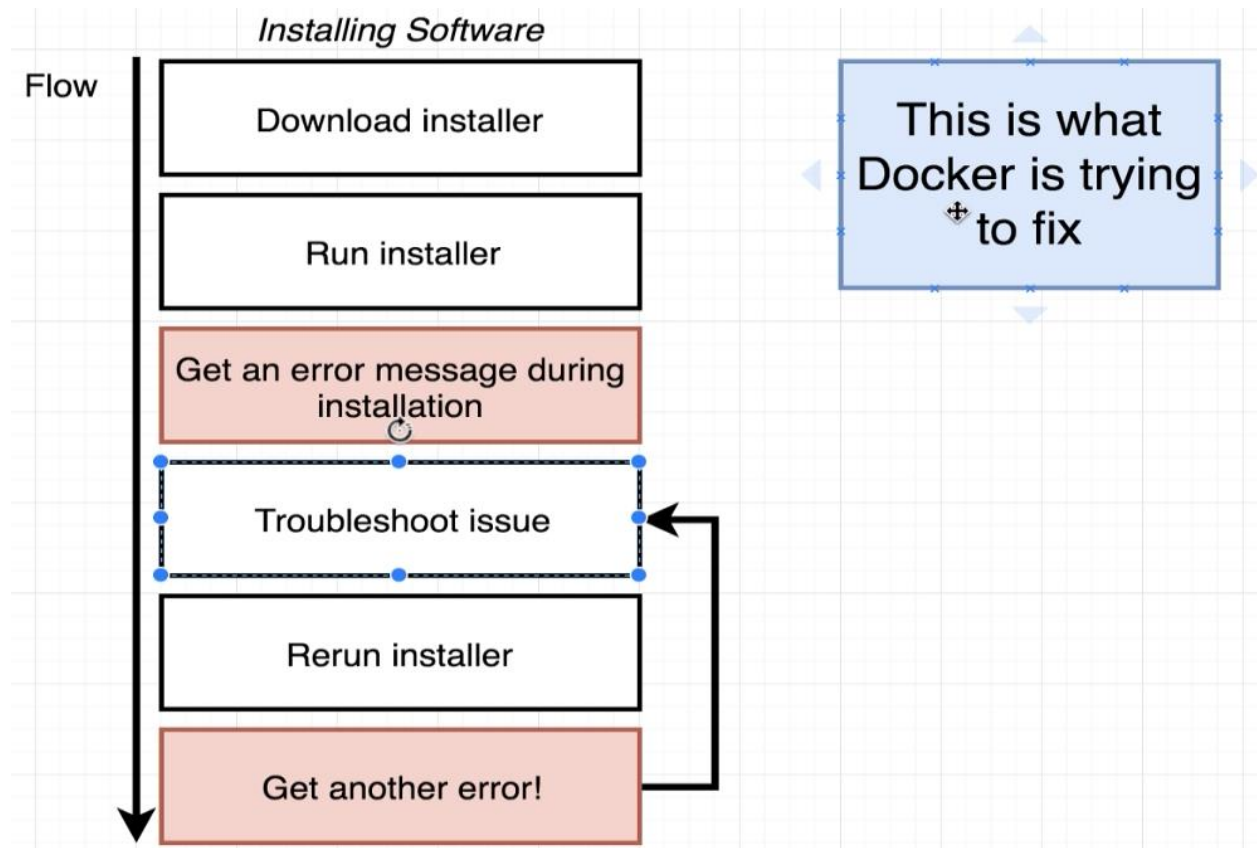


Docker a Kubernetes

Build, test, a deploy Docker applications s Kubernetes a zároveň sa naučiť pracovné postupy vývoja v produkčnom štýle.

Docker teoretická časť -

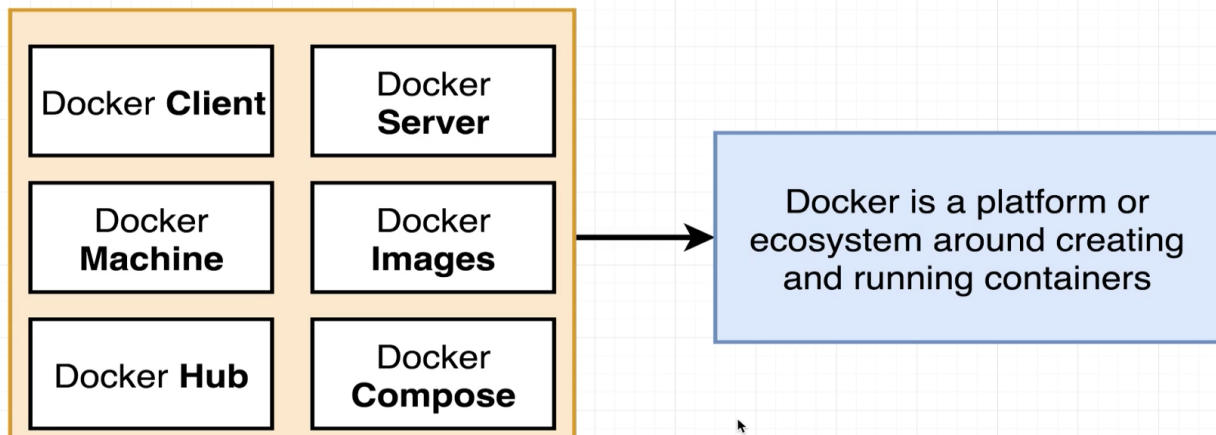
Prečo používať docker ?



Docker sa snaží čo najľahšie a najrychlejšie nainštalovať nejaký software na počítač či už osobný alebo akykoľvek iný (webservice a cloud base).

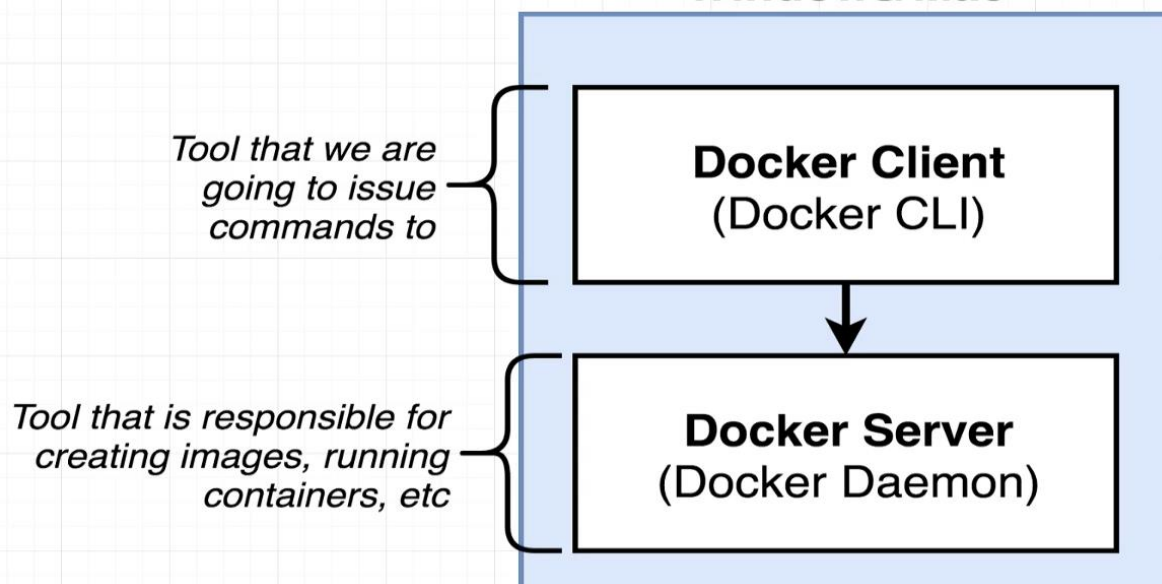
Co je docker ?

Docker Ecosystem

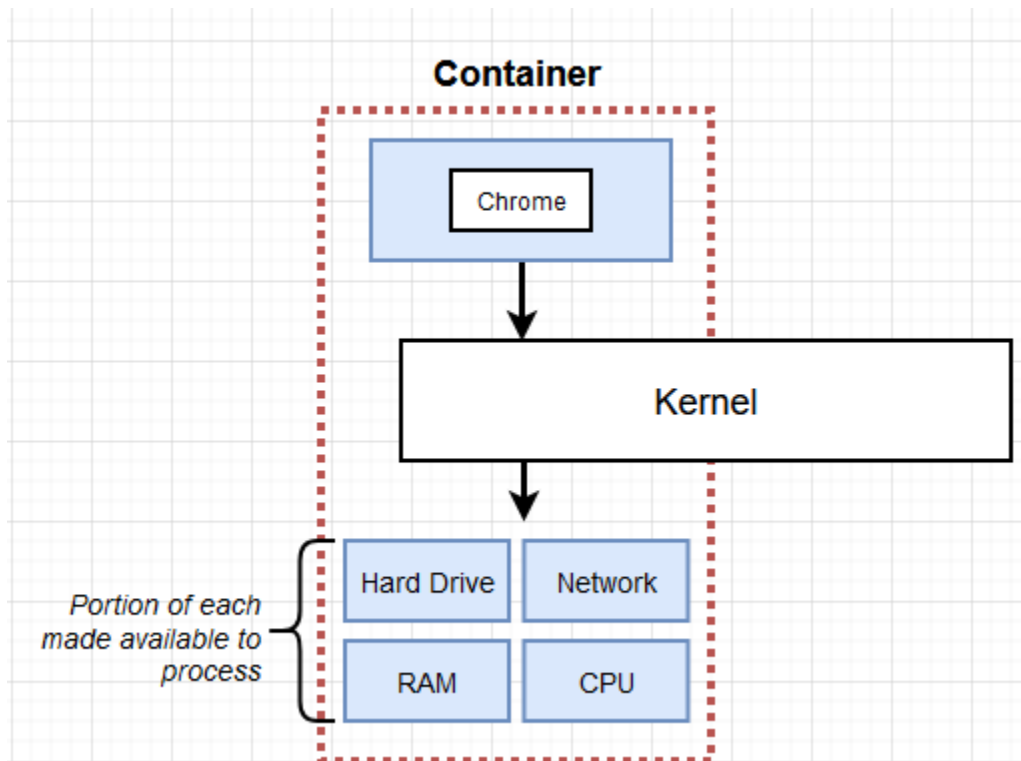


Container je instancia imagu ktora spusta program predom zadefinovaneho image templatu , Docker CLI -> docker hub -> stiahne image -> a docker na zaklade toho spusti a vytvori container ktory berie template z toho imagu

Docker for Windows/Mac

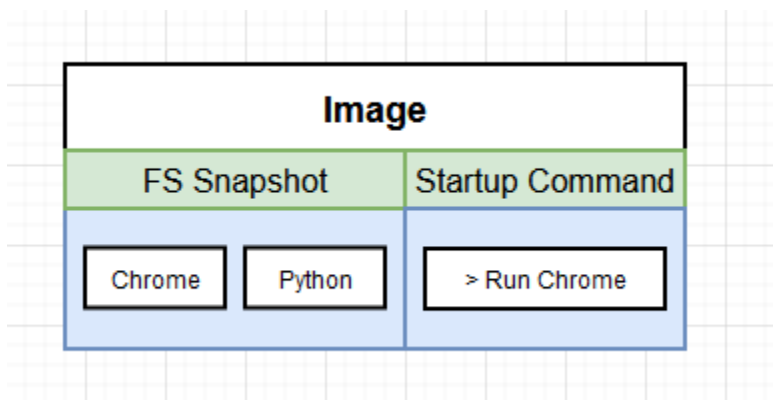


Co je container ?



container (kontajner) izolované prostredie, ktoré umožňuje spustenie aplikácie alebo služby s všetkými potrebnými závislosťami a konfiguráciou, bez ohľadu na to, na akom operačnom systéme alebo hardvéri sa spúšťa.

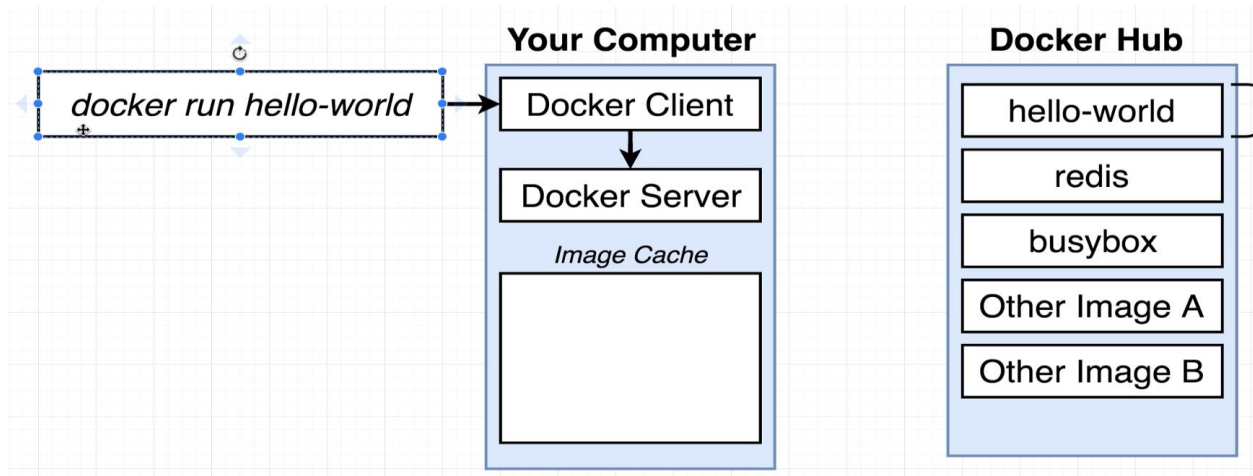
Ako jeden image môže spustiť viac containerov ?



Docker prakticka časť

Hello-world

https://hub.docker.com/_/hello-world



Tým že sme tento image nepoužívali tak nevedle najst ten image v cache tak ho stiahol pomocou docker servre z docker hub (dokumentacia dockerhub hello-world)

Ak použijeme prikaz

```
user@DESKTOP-F5J917O:~$ docker run busybox ls
```

bin

dev

etc

home

lib

lib64

proc

root

sys

tmp

usr

Var

Tak všetko sa vykona v containery

Toto však nemožeme spraviť pri hello-world lebo v imagine hello-world file systeme tento príkaz nie je preto ho nepozná a nevie ho spustiť.

```
user@DESKTOP-F5J917O:~$ docker run hello-word echo hi there
```

Unable to find image 'hello-word:latest' locally

docker: Error response from daemon: pull access denied for hello-word, repository does not exist or may require 'docker login'.

See 'docker run --help'.

Rozdiel medzi spustiním a vytvorením

```
user@DESKTOP-F5J917O:~$ docker create hello-world
```

```
e8409ce15f415540b05d22b6959528dbe47e65b964158d923d3fee6c6dd84f6e
```

```
user@DESKTOP-F5J917O:~$ docker start -a
```

```
e8409ce15f415540b05d22b6959528dbe47e65b964158d923d3fee6c6dd84f6e
```

Hlavné rozdiely:

Príkaz	Čo robí	Výstup
docker create	Vytvorí kontajner, ale nevykoná ho .	Vráti ID kontajnera, ale neukáže výstup.
docker start	Spustí už vytvorený kontajner.	Ak použiješ -a, pripojí sa na výstup kontajnera.
docker run	Vytvorí a spustí nový kontajner zobrazený v príkaze.	Spustí kontajner a pripojí sa na jeho výstup, až kým sa kontajner neskončí.

Kedy použiť ktorý príkaz?

- `docker run` je najjednoduchší a najpoužívanejší príkaz, keď chceš rýchlo spustiť kontajner na základe obrazu, pretože v sebe kombinuje vytvorenie aj spustenie kontajnera.
- `docker create` a `docker start` môžeš použiť v situáciách, kde chceš oddeliť proces vytvárania kontajnera od jeho spustenia, čo môže byť užitočné napríklad v prípade, že chceš najprv upraviť konfigurácie kontajnera, alebo potrebujete spustiť kontajner viackrát s rôznymi parametrami.

Docker stop a docker kill

	Bezpečne		
	zastaví	<code>SIGTERM</code> (pre pokus o	10 sekúnd (predvolené),
<code>docker</code>	kontajner,	bezpečné ukončenie),	po ktorých sa pošle
<code>stop</code>	umožní aplikácii	následne <code>SIGKILL</code> ak sa	<code>SIGKILL</code> ak kontajner
	správne ukončiť	kontajner neukončí včas.	stále beží.
	činnosť.		
<code>docker</code>	Okamžite zabije kontajner, nečaká na	<code>SIGKILL</code> (alebo iný	Okamžité
<code>kill</code>	ukončenie procesov, bez ohľadu na to,	signál, ak je	zastavenie
	či aplikácia je pripravená.	špecifikovaný).	kontajnera.

Ak sa container po 10 sekundach ak zadame prikaz `docker stop` nestopne -> vykona sa `docker kill` automaticky

Vytvorenie Docker image

Postup :

Dockerfile -> docker client -> docker server -> použiteľny image

Vytvorenie dockerfilu:

Specifikovať základný image -> spustenie príkazov na podporné service -> špecifický command na spustenie containera

Vytvorenie docker image, ktorý nám spustí redis server.

Informácie v docker_linux_CLI_commands.txt

V sekcii “DOCKERFILE”

Zhrnutie :

V tomto docker file sme prv nainštalovali alpine linux verziu z docker hubu , to je základny blok pre náš container a nasledne sme nainštlovali redis (je to linux comand ktorý sme mohli dat lebo používame alpine linux) , a posledny command spusti redis pri štarte containera