

DevOps Development & Operation

Pertemuan ke-1



Pesantren Teknologi Informasi dan Komunikasi

Jln. Mandor Basar No. 54 RT 01/RW 01 Rangkapanjaya,
Pancoran Mas, Depok 16435 | Telp. (021) 77 88 66 91
Koordinat (-6.386680 S, 106.777305 E)

www.petik.or.id





Jalan Mandor Basar Nomor 54, RT.
01/001, Rangkapanjaya, Pancoran
Mas, Kota Depok 16435



www.petik.or.id



021 7788 6691



info@petik.or.id

السلام عليكم

Wahyu Januar A



0838-1934-7140



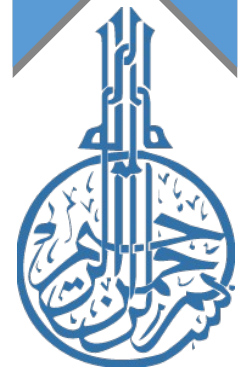
wahyu.pyan88@gmail.com



Wahyu Pyan



wahyu_pyan



مَنْ سَلَكَ طَرِيقًا يَلْتَمِسُ فِيهِ عِلْمًا، سَهَّلَ اللَّهُ لَهُ بِهِ طَرِيقًا إِلَى الْجَنَّةِ

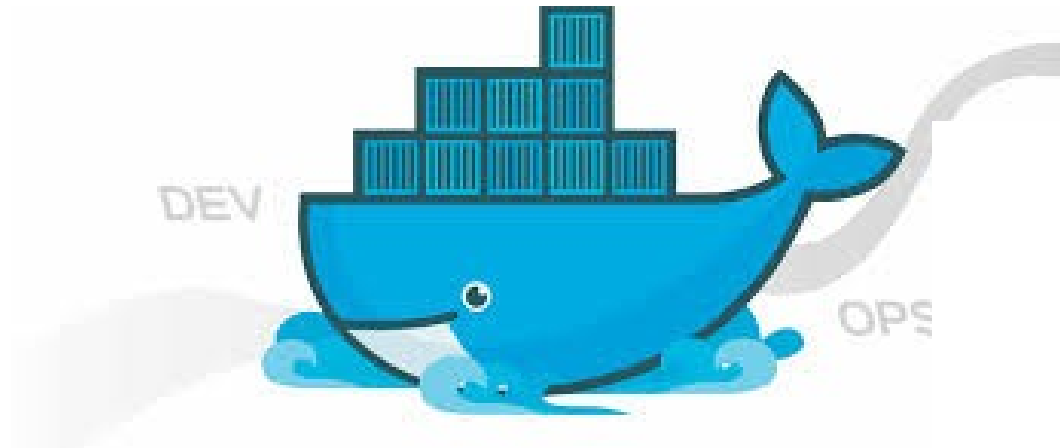
|

“

“Barang siapa menelusuri jalan untuk mencari ilmu padanya, Allah akan memudahkan baginya jalan menuju surga.”
(HR. Muslim)

“

Rencana Perkuliahan



kubernetes

Virtualisasi



Pembuatan suatu sistem virtual dari suatu sistem fisik, misalnya untuk menjalankan sistem operasi virtual atau tamu (guest), misal Linux, pada sistem operasi induk (host), misal Windows.

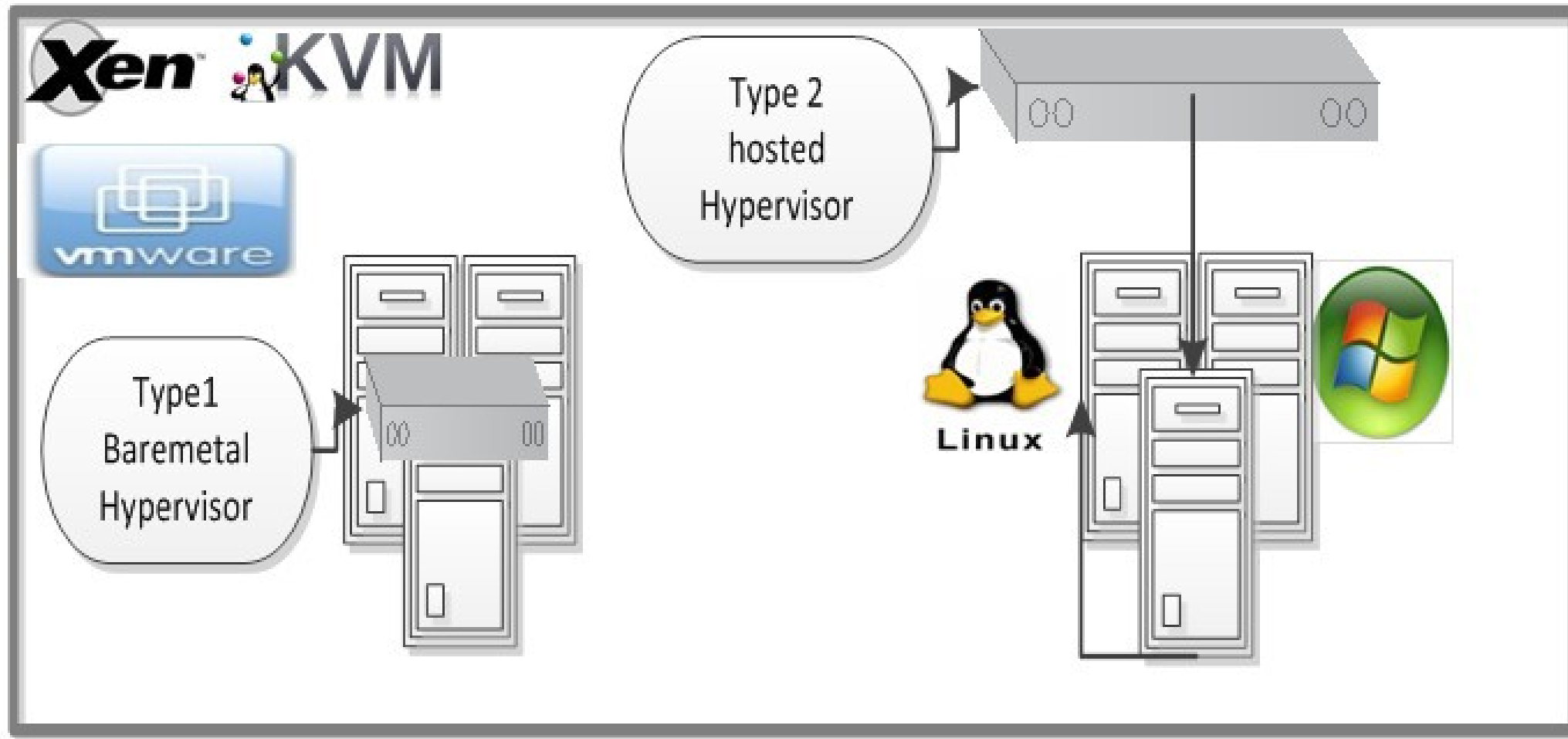
- Full Virtualization: guest menjalankan mesin sendiri.
Contoh: VirtualBox, VMWare, Microsoft Hyper-V, dll.
- Para Virtualization: kernel host dan guest dimodifikasi agar lebih cepat. Contoh: Xen, KVM, dll.
- OS level Virtualization. Contoh: Docker, openVZ, dll.

Hypervisor

Hypervisor adalah virtualisasi di level hardware (Full/Para Virtualization), yang terdiri dua jenis:

- **Bare Metal Hypervisor:**
Komputer yang hanya berfungsi sebagai mesin virtual. Kuat dan cepat, tapi mahal harganya. Contoh: VMWare ESXi
- **Hosted Hypervisor:**
mesin virtual (misal VirtualBox dan VMWare) yang berjalan di komputer dan sistem operasi biasa. Contoh: VMWare Workstation dan VirtualBox.

Bare Metal VS Hosted Hypervisor



Virtualisasi vs Kontainerisasi



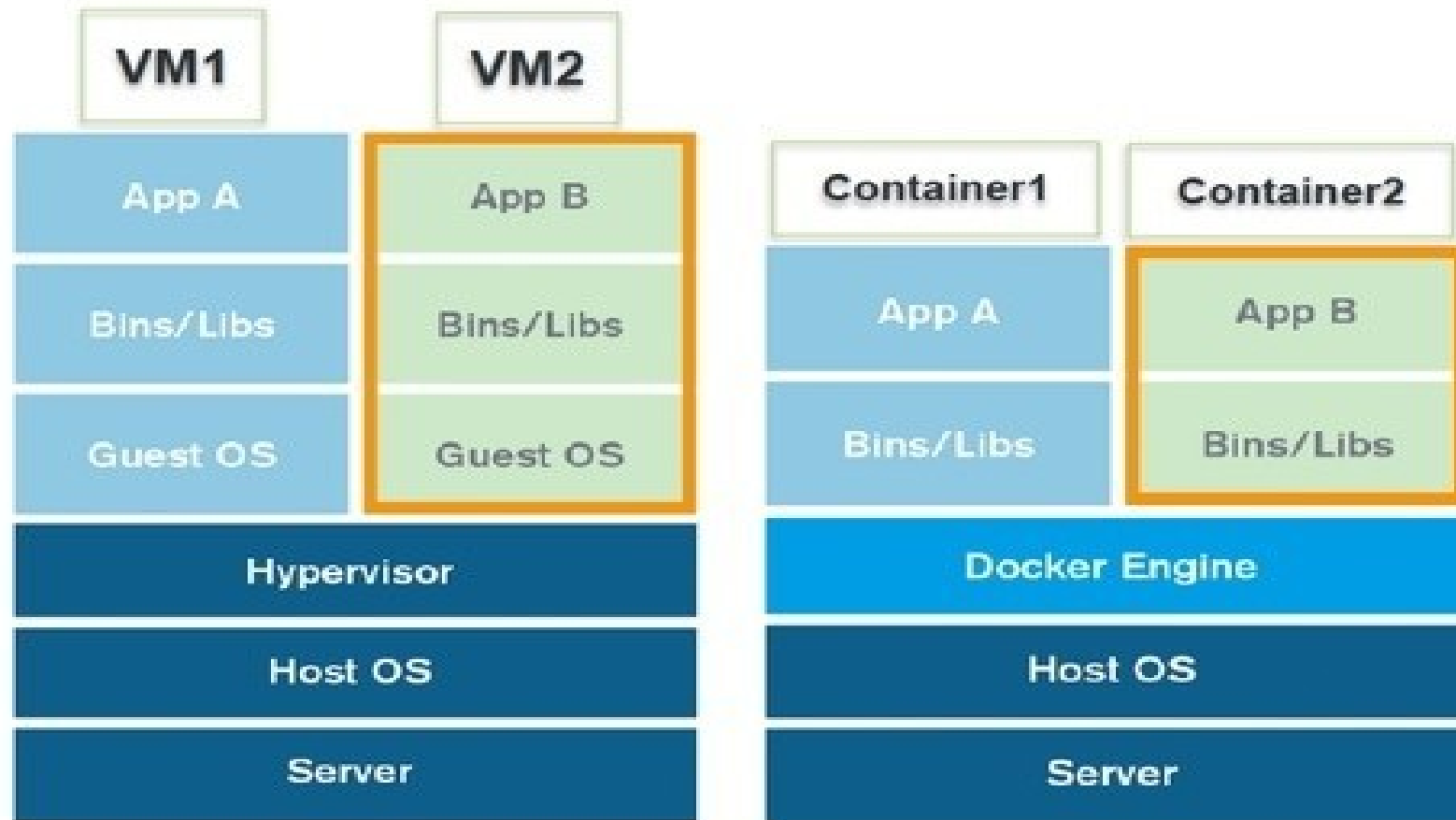
- **Virtualisasi (Full/Para Virtualization):**

Sistem operasi induk (host) membagikan prosesor dan memori (RAM) kepada komputer virtual (guest) untuk menjalankan sistem operasi secara penuh mulai dari booting.

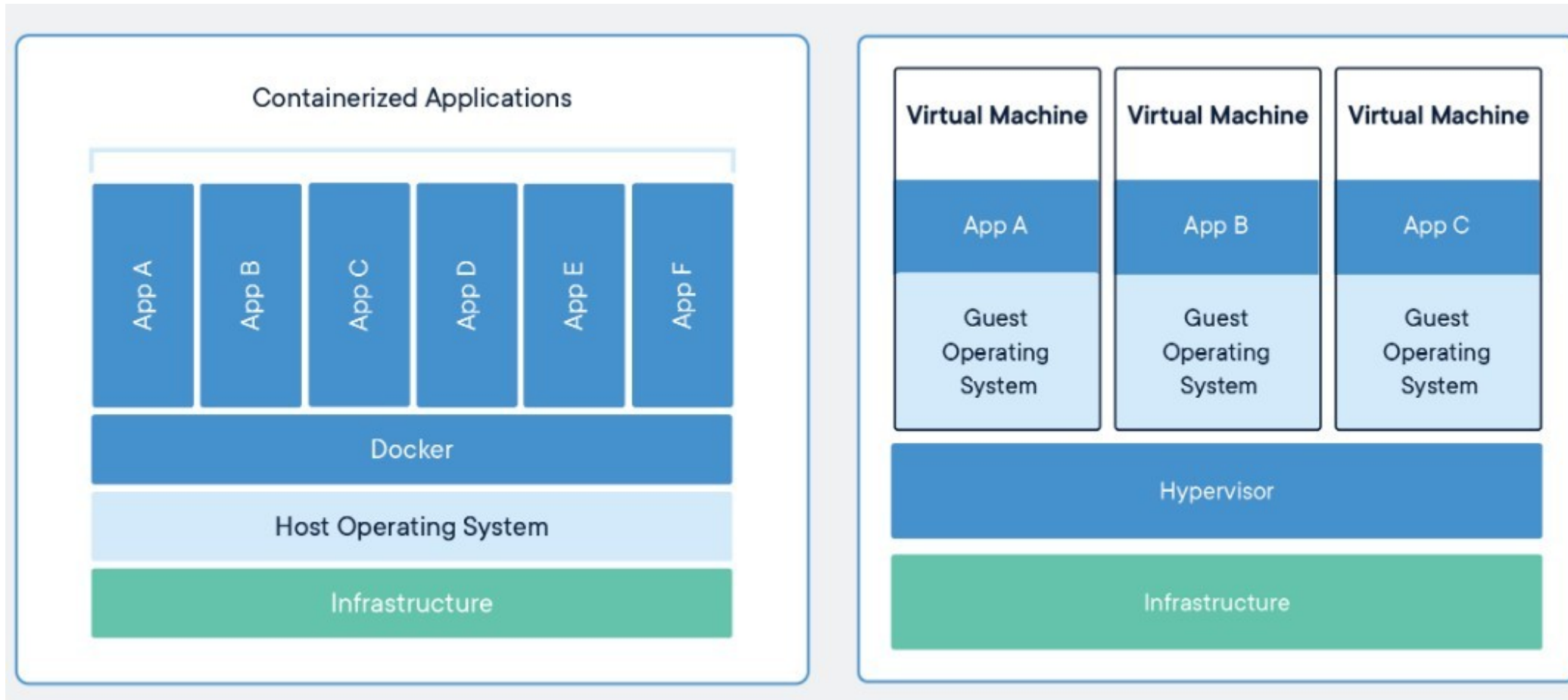
- **Kontainerisasi (OS Level Virtualization):**

Sistem operasi induk (host) hanya membagikan sumber daya sistem operasi (kernel dan fungsi dasar sistem operasi) kepada komputer virtual (container), sehingga kerja host ringan (hemat CPU dan RAM) dan kerja container cepat (tidak booting komputer dari awal).

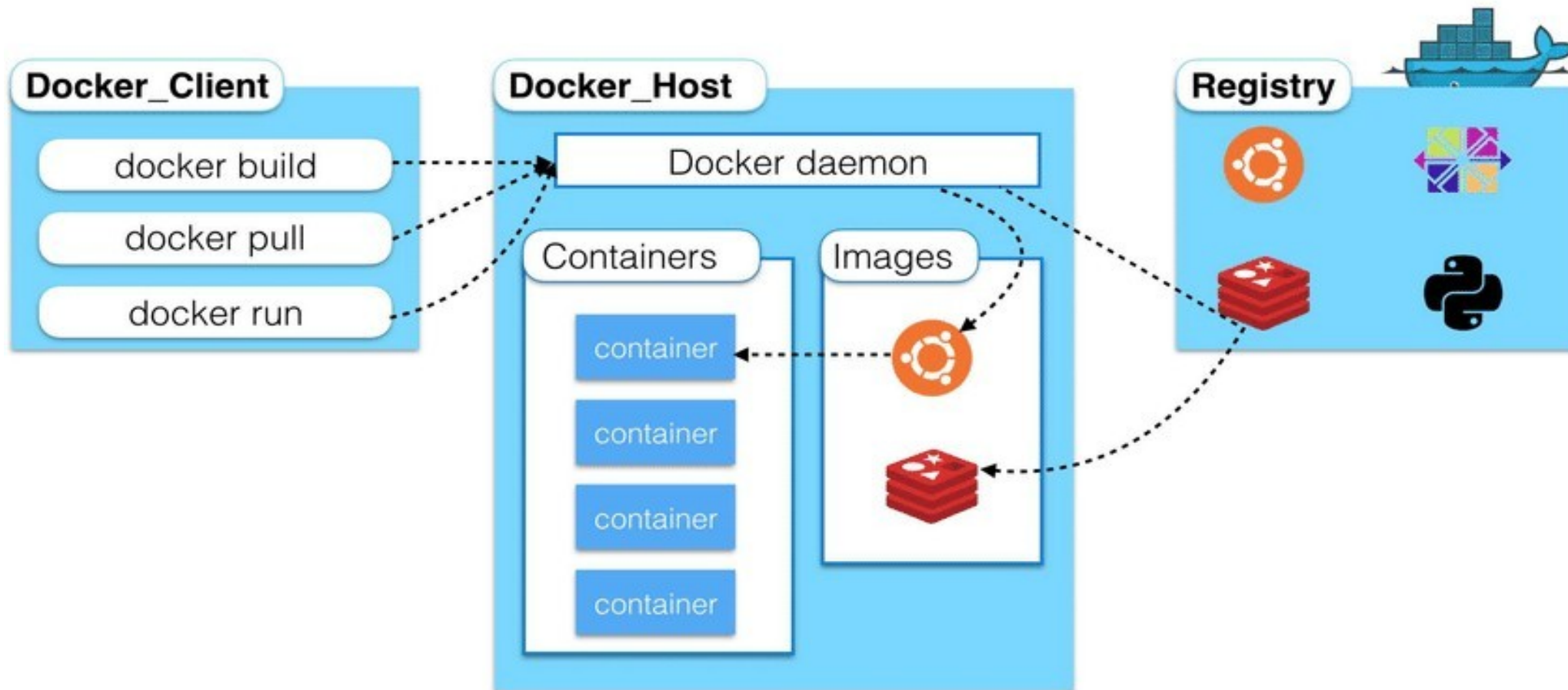
Virtualisasi vs Kontainerisasi



Container vs Hypervisor



Arsitektur Docker



Images & Containers



- ***Images*** - Cetak biru aplikasi yang membentuk dasar container. Dalam demo di atas, kita menggunakan perintah `docker pull busybox` untuk mengunduh image busybox.
- ***Containers*** - Dibuat dari image Docker dan menjalankan aplikasi yang sebenarnya.

Images & Containers



- Sebuah *Container* dibuat dengan perintah **docker run** yang lebih dahulu mengunduh (pull) image jika image belum ada di Docker Host.
- Lebih dari satu *container* dapat dibuat dengan **docker-compose**.
- *Container* yang sedang berjalan dapat dilihat dengan perintah **docker ps**. Semua *container* (termasuk yang sudah quit/exit atau mati) dapat dilihat dengan perintah **docker ps -a..**

Docker Daemon



- ***Docker Daemon*** :

Layanan di background (berjalan di belakang layar) yang berjalan pada host (komputer yang menjalankan docker) dan berfungsi membangun, menjalankan, dan mendistribusikan container Docker.

Daemon adalah proses yang berjalan di sistem operasi (host) yang melayani komunikasi dengan ***Docker Client***.

Docker Daemon



- ***Docker Client*** -

Command line tool (alat program berupa baris perintah) yang memungkinkan pengguna untuk berinteraksi dengan Docker Daemon. Secara umum, ada bentuk lain docker client -- seperti Kitematic (<https://kitematic.com/>) -- yang menyediakan GUI kepada pengguna.

Docker Hub

- Docker Hub - Repositori atau Registry yang berisi kumpulan image untuk Docker. Registry adalah direktori semua image Docker yang tersedia.
- Kita dapat meng-host registry Docker sendiri dan dapat menggunakannya untuk mengunggah (push) dan mengunduh (pull) image.

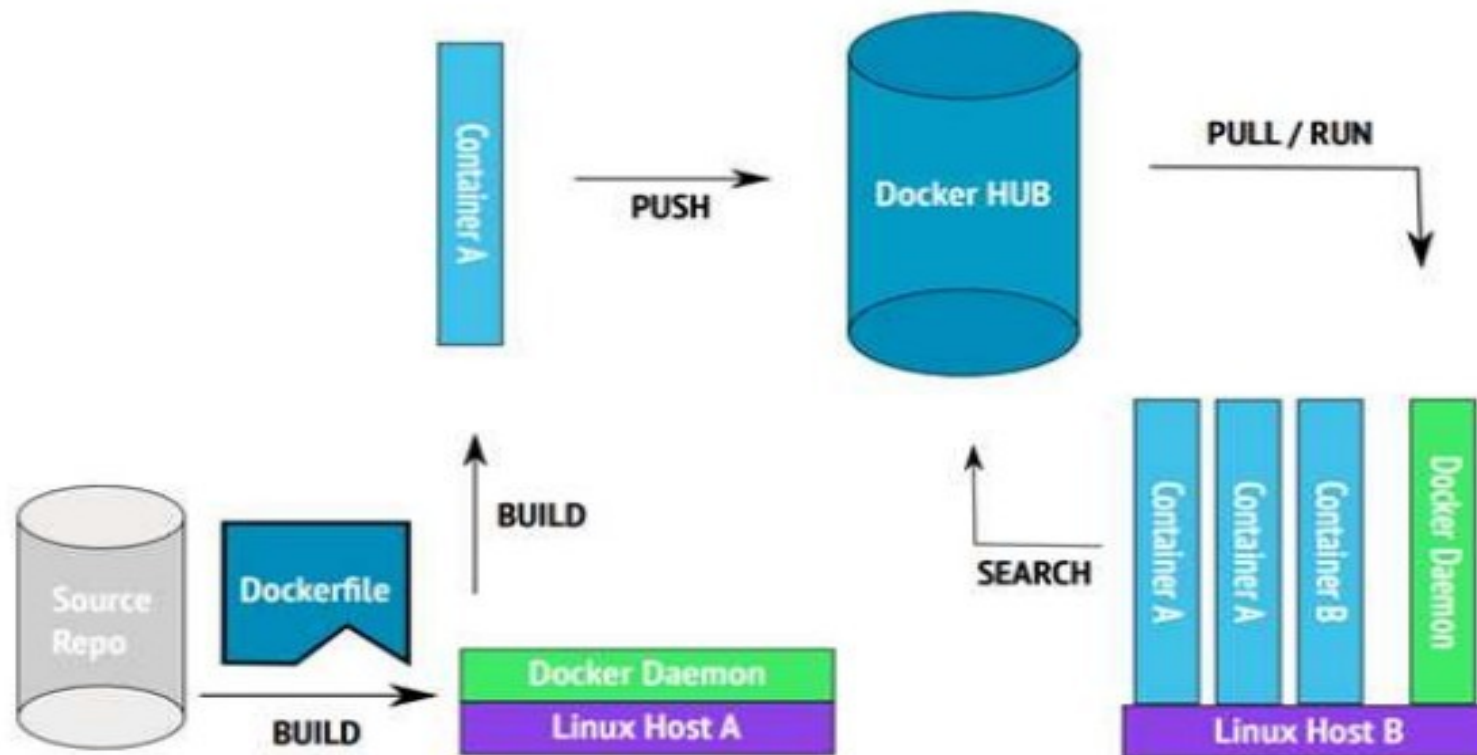
Docker on Development Process



- Developer create applications
- Build a container by deploying the applications with other components such as web server and database.
- Push the container to Docker Hub (Ready to be used).
- Download the container and run on production system (Cloud or On-premise).

Docker on Development Process

Docker: Workflow





Jalan Mandor Basar Nomor 54, RT. 01/001, Rangkapanjaya,
Pancoran Mas, Kota Depok 16435



www.petik.or.id



021 7788 6691



info@petik.or.id