

# Virtual Machine dan Parallel Processing in R

## Sebuah Learning Forum

Ikang Fadhli

Market Research @nutrifood

18 January 2022

- 1 MUKADIMAH
- 2 *VIRTUAL MACHINE*
- 3 *PARALLEL PROCESSING*

## Section 1

# MUKADIMAH

# Learning Forum Hari Ini

Pada hari ini, kita akan membahas dan berdiskusi tentang beberapa topik, yakni:

## ① *Virtual Machine*

- *Linux OS*
- *Google Cloud Virtual Machine*

## ② *Parallel Processing*

- *Serial Processing*
- *Embarassingly Parallel*
- *Parallel yang Sebenarnya*

## Section 2

# ***VIRTUAL MACHINE***

## Apa itu *Virtual Machine*?

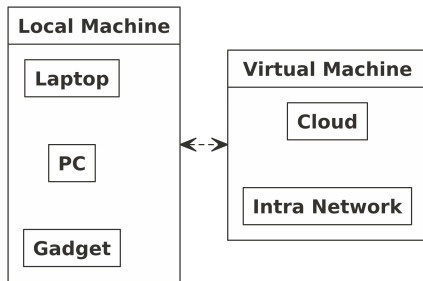
Pada konsep *computing*, *virtual machine* adalah *virtualization* atau *emulation* dari suatu sistem komputer.

*Virtual machine* berbasis suatu arsitektur komputer tertentu dan berfungsi sama halnya dengan komputer pada biasanya.

Salah satu *operating system* yang paling sering digunakan pada *virtual machine* adalah LINUX OS.

**Android** merupakan salah satu *mobile OS* yang berbasis Linux.

# Apa itu *Virtual Machine*?



Pembahasan pada **LEFO** kali ini dibatasi untuk *cloud VM*. Istilah lain yang sering digunakan untuk menyebut *VM* di *cloud* adalah *virtual private server* (VPS).

## Apa itu LINUX OS?

Linux adalah *operating system* berbasis UNIX (sama halnya dengan MacOS) yang bersifat *open source*. Dibuat oleh seorang *programmer* bernama **Linus Torvalds** pada 1991. Sampai saat ini, OS Linux merupakan OS yang populer digunakan di berbagai *server* dan *high performance computer* di dunia.

Ada banyak *distro* Linux yang berkembang (dengan basis komunitas yang kuat) antara lain:

- Ubuntu,
- Debian,
- Kali,
- Arch,
- Fedora.

Setiap distro memiliki spesialisasi tertentu sesuai dengan komunitas pengembang dan tujuannya.

Distro yang paling sering digunakan pada *server* dan *HPC* adalah Debian atau Ubuntu.



## *Fun Fact*

**Linus Torvalds** juga salah seorang *creator* dari Git.

# Apa itu LINUX OS?

Linux memiliki dua versi, yakni:

- 1 *Command line interface.*
- 2 *Graphical user interface.*

Kita bisa memilih versi mana yang hendak kita gunakan di *virtual machine*. Tapi versi **CLI** lebih disukai karena lebih ringan dan bisa memaksimalkan kemampuan *processors* yang dimiliki.

# Apa itu LINUX OS?

Linux sangat ringan, bahkan bisa dijalankan dengan **USB drive** tanpa harus di-*install* terlebih dahulu di laptop/komputer.

Selain itu, Linux bisa di-*install* di berbagai komputer dan *gadget* apapun. Misal: HP / tablet Android dan laptop Windows.

Di *gadget* Android, kita bisa mencicipi Linux CLI dan GUI dengan meng-*install apps* bernama userLand. Pada laptop Windows, kita bisa meng-*install* berbagai *virtual box* (misalkan MobaXterm).

## Google Virtual Machine

Berbekal *credit* yang kita dapatkan saat mengaktifkan layanan *Google Cloud Service*, kita bisa menyewa *virtual machine* sesuai dengan kebutuhan.

**Bagaimana caranya?**

# *Google Virtual Machine*

DEMO

# UserLand di Android

DEMO

# Diskusi

*Any comments?*

## Section 3

# *PARALLEL PROCESSING*



## Apa Tujuannya *Parallel Processing*?

Manusia itu terobsesi dengan hal-hal yang bisa dilakukan dengan cepat. Apalagi jika sumber daya komputasi yang dimiliki sangat mumpuni.

***Jika suatu komputasi bisa dilakukan lebih cepat, kenapa tidak?***

# Lebih Cepat, Lebih Baik!



# *Serial Processing vs Parallel Processing*

## *Serial Processing*

Adalah proses komputasi yang bersifat sekuensial. Misal suatu proses komputasi memiliki  $n$  proses. Proses ke  $i$  tidak bisa dilakukan hingga proses ke  $i - 1$  selesai ( $i \leq n$ ).

## *Parallel Processing*

Adalah proses komputasi dimana semua proses yang terlibat dijalankan secara bersamaan. Misalkan suatu proses komputasi memiliki  $n$  proses dijalankan bersamaan. Proses ke  $i$  bisa dijalankan tanpa menunggu proses yang lain.

# Analogi



**Menurut kalian proses perakitan satu mobil ini berlangsung secara serial atau parallel?**

# Analogi

Suatu ketika, saya diminta tolong untuk memberikan informasi kepada 8 orang via *messaging*.

Apa harus saya lakukan jika saya menggunakan:

- 1 *Serial Processing*
- 2 *Parallel Processing*

# Analogi

Dalam *parallel processing* ada yang disebut dengan *embarassingly parallel*. Apakah se-memalukannya itu?

## Contoh I

Saya membuat grup WA berisi 8 orang tersebut. Lalu menginformasikannya melalui grup tersebut.

## Contoh II

Saya meminta tolong orang lain sehingga menggunakan 8 buah *gadgets* untuk mengirimkan informasi tersebut kepada 8 orang secara bersamaan.

## Pertanyaannya

Antara contoh I dan II, mana yang memalukan?

## Ide Dasar *Parallel Processing*

Jika suatu proses komputasi bisa dilakukan selama  $X$  detik pada *single processor*, maka jika kita memiliki  $n$  buah *processors* prosesnya seharusnya bisa dihemat menjadi  $\frac{X}{n}$  detik.

**Setuju? Logis gak?**

# Tapi...

Percepatan *runtime* ini tidak berlaku *linear* seperti di atas. Tapi yang jelas memang lebih cepat dibanding *serial processing*.

Lantas seberapa cepat?



## Berbicara Mengenai *Parallel Processing*

Dahulu kala saat berbicara mengenal *parallel processing*, istilah ini hanya digunakan saat seseorang melakukan komputasi menggunakan *high performance computer* (HPC). HPC merupakan “super komputer” yang dibuat dari beberapa komputer (multi *processors*) yang dibuat sedemikian rupa (menggunakan arsitektur jaringan tertentu).

Namun demikian, istilah ini sekarang bisa untuk segala jenis komputer. Bahkan *gadget* Android yang kita pakai sekarang sudah *multicores* sehingga memungkinkan untuk melakukan *parallel processing*.

# Contoh

Deteksi berapa banyak *cores* di Huawei T10s.

```
ix@localhost:~/ikanx101.github.io/_posts/matematika ITB/server/post3$ lscpu
Architecture:      aarch64
Byte Order:        Little Endian
CPU(s):            8
On-line CPU(s) list: 0-7
Thread(s) per core: 1
Core(s) per socket: 4
Socket(s):         2
Vendor ID:         ARM
Model:             4
Model name:        Cortex-A53
Stepping:          r0p4
CPU max MHz:       1997.0000
CPU min MHz:       480.0000
BogoMIPS:          3.84
Flags:             fp asimd evtstrm aes pmull sha1 sha2 crc32 cpuid
```