



Тема 9

Віртуалізація

План

1. Поняття про віртуалізацію
2. З історії віртуалізації
3. Огляд технологій віртуалізації

1. Поняття про віртуалізацію

? Що таке віртуальний?

Вигаданий, уявний, зімітований.

Віртуалізувати - перейти від фактичної реальності до реальності потенційної через певну діяльність людини.

? Що буває віртуальним?

Віртуальна реальність, віртуальний клас, віртуальна економіка, віртуальна мережна спільнота, віртуальні машини тощо.

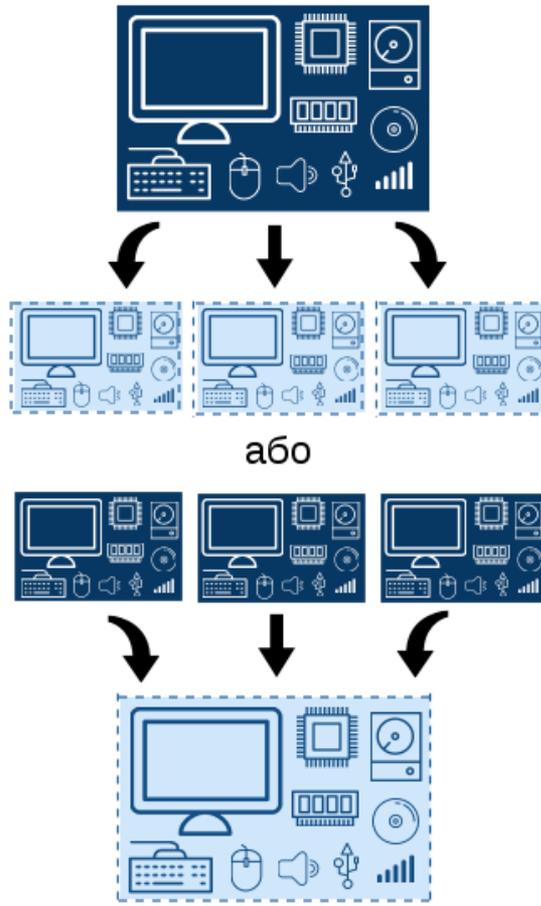
? Віртуальне - несправжнє?

Дайте свою відповідь :)

1. Поняття про віртуалізацію

Віртуалізація – це поняття, що об'єднує технології, засоби, методи тощо, котрим притаманні три головні *рис*и:

1)



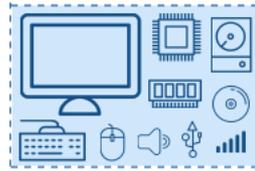
поділ ресурсів
одного фізичного комп'ютера
на декілька взаємно
незалежних віртуальних
середовищ

або

об'єднання ресурсів
кількох фізичних комп'ютерів
в одне віртуальне середовище

1. Поняття про віртуалізацію

2)



оперативність переходу
з одного віртуального
середовища
в інше

3)



приховування реальних
фізичних ресурсів та заміна їх
абстракціями

1. Поняття про віртуалізацію

? Які програмні продукти для віртуалізації ви знаєте?

Різні програмні засоби віртуалізації належать до різних технологій віртуалізації. Одна з найпопулярніших - віртуальні машини.

Віртуальна машина / VM (*virtual machine / VM*) - зімітований у межах реального комп'ютера віртуальний комп'ютер, на який може бути встановлено окрему ОС.

Гіпервізор (*hypervisor*) - спеціальне програмне забезпечення, яке

розподіляє ресурси між VM,
організовує взаємодію VM з фізичним обладнанням,
надає користувачу інтерфейс для керування VM.

Не плутайте віртуальні машини (результат) з гіпервізорами (засіб)!

2. 3 історії віртуалізації

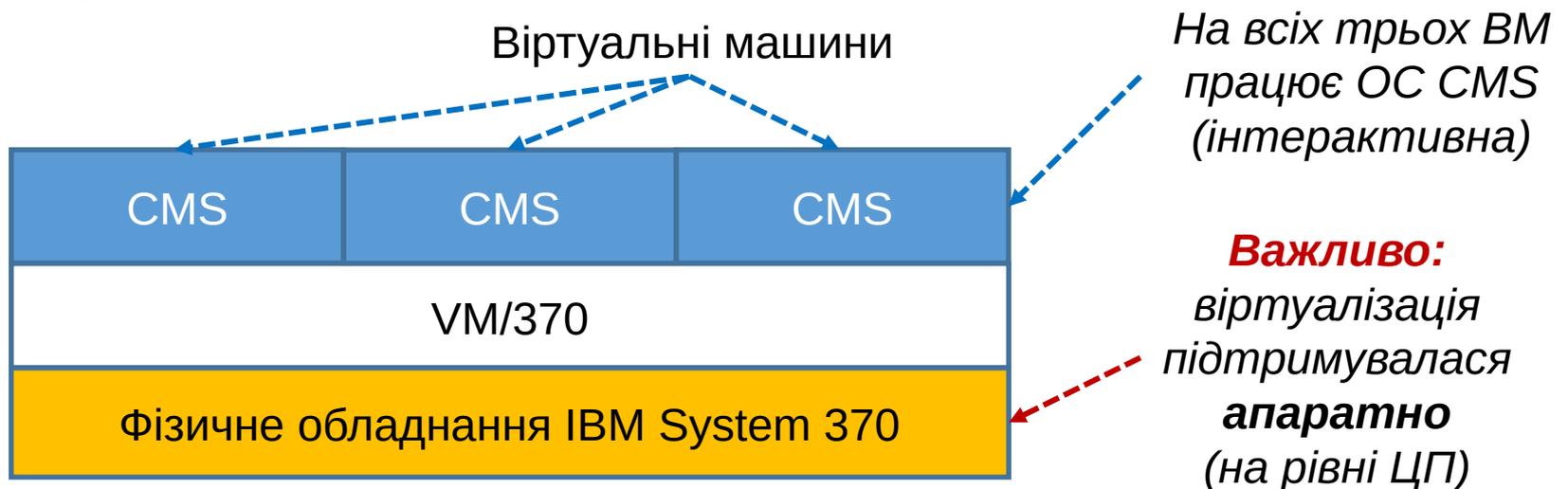
60-ті роки XX ст.: віртуалізація задля багатозадачності

У **VM/370** (для IBM System 370) кожний користувач працював з окремим образом всього апаратного забезпечення фізичної машини - віртуальною машиною.

Користувачі могли працювати кожний зі своєю віртуальною машиною - паралельно (*режим поділу часу*).

CP 67 - окрема програма у складі VM/370, яка керувала віртуальними машинами (*монітор віртуальних машин*).

На віртуальних машинах працювала окрема ОС (інтерактивна чи пакетної обробки).



2. 3 історії віртуалізації

x86: кому потрібна ця віртуалізація на ПК?

З появою ПК потреба в режимі поділу часу відпала (принаймні для звичайних користувачів). Тому у процесорах Intel x86 **апаратну підтримку віртуалізації реалізовано не було**.

Втім, ідея віртуальних машин лишилася: кілька віртуальних комп'ютерів всередині одного, тепер уже персонального. Але без апаратної віртуалізації для цього була серйозна перепона.

Службова інструкція (службова команда, *sensitive instruction*) - команда ЦП, необхідна для роботи ОС й розрахована на виконання лише на найвищому рівні привілегій.

Якщо службова інструкція надходить від програми, що працює на нижчому рівні привілеїв (від ОС, запущеної не безпосередньо, а на VM), то таку інструкцію буде виконано інакше або не буде виконано зовсім.

Приклади службових команд: команди введення-виведення, команди зміни налаштувань MMU тощо.

2. 3 історії віртуалізації

x86 (продовження)

У VM/370 службові інструкції перехоплював ЦП. Він же робив відповідні виклики монітора віртуальних машин (це **апаратна віртуалізація** - *hardware-assisted virtualization*). Але в x86 цього передбачено не було.

System/360, System/390, z-серії: тим часом у світі мейнфреймів

Важливо (та іронічно): в ОС System/360, а згодом в System/390 та z-серіях (теж IBM) апаратна підтримка віртуалізації лишалася. І лишається дотепер.

DISCO, VMware, паравіртуалізація, віртуальні контейнери, WINE: герої йдуть в обхід

Були спроби обійти апаратну неспроможність і реалізувати віртуалізацію повністю програмно. Перелічимо найуспішніші.

Проект DISCO (90-ті рр. XX ст., Стенфордський університет). З цього проекту виросла компанія **VMware** та технологія динамічної трансляції (dynamic translation, перший продукт - 1999 рік).

2. 3 історії віртуалізації

DISCO, VMware, паравіртуалізація, віртуальні контейнери, WINE (продовження)

Основна ідея **методу динамічної трансляції**: перехоплювати проблемні команди й замінювати їх безпечними послідовностями коду. Змін до джерельних кодів гостьової ОС не вноситься.

Метод **паравіртуалізації** (paravirtualization) передбачає заміну службових інструкцій безпосередньо у коді гостьової ОС (потрібний доступ до джерельних кодів гостьової ОС).

Повні емулятори (pure emulators) імітують усе апаратне забезпечення комп'ютера, включно з процесором. Має нижчу продуктивність, але дозволяє зімітувати машину однієї архітектури на машині іншої архітектури.

Емуляція системних бібліотек (system library emulation) - низка дуже різних за реалізацією технологій, які віртуалізують не всю ОС, а її системні бібліотеки. Наприклад, коли програму для Windows запущено у Linux за допомогою WINE, то кожний API-виклик цієї програми перехоплюється, й імітується таке його опрацювання, ніби програму й справді запущено у Windows.

2. 3 історії віртуалізації

DISCO, VMware, паравіртуалізація та віртуальні контейнери (завершення)

Віртуальні контейнери (*virtual containers*), або віртуалізація рівня ОС (*OS level virtualization*) - технологія, що передбачає створення і паралельну роботу в межах одного фізичного комп'ютера окремих віртуальних середовищ (контейнерів). Контейнери відносно незалежні, але використовують одне ядро. Головна перевага - швидкі та потребують менше ресурсів, ніж повноцінні віртуальні машини.

Віртуальні контейнери розраховані на Unix-подібні ОС. У деяких випадках можуть запускатися на Windows як на основній ОС (наприклад, Docker), але для цього основна ОС має надати контейнерам віртуальну машину.

Цей перелік неповний. Більше - у п. 3 цієї лекції.

2005: повернення апаратної віртуалізації для ПК

У 2005 році компанії Intel та AMD представили свої перші процесори з підтримкою віртуалізації (Intel VT-x та AMD-V).

Обоє спираються на досвід IBM VM/370.

3. Огляд технологій віртуалізації

Основні тези

- Технології віртуалізації можна систематизувати по-різному.
- Конкретні засоби віртуалізації часто використовують поєднання кількох технологій.
- Весь час з'являється щось нове.

Систематизація технологій віртуалізації

за напрямом

Віртуалізація серверів

server virtualization

Віртуальні контейнери virtual containers

Docker, LXC, LXD, OpenVZ,
Parallels Virtuozzo, FreeBSD-jail

Віртуальні машини virtual machines

Гіпервізори I типу type 1 hypervisors

VMware ESXi, Citrix Xen Server,
MS Hyper-V

Гіпервізори II типу type 2 hypervisors

VMware Workstation, VMware Fusion,
Oracle VirtualBox, KVM

Віртуалізація робочих столів

desktop virtualization

Сервіси віддаленого робочого столу remote desktop services

MS RDS, xrdp

Інфраструктура віртуального робочого столу virtual desktop infrastructure (VDI)

MS RDS (роль Remote Desktop Virtualization Host),
Citrix Virtual Apps and Desktops

Віртуалізація прикладних програм

application virtualization

Віртуальні контейнери virtual containers

Docker, LXC, LXD, OpenVZ,
Parallels Virtuozzo, FreeBSD-jail

Інші технології

Citrix Virtual Apps and Desktops,
MS App Virtualization (MS App-V),
Snap, Flatpak

Віртуалізація мереж

network virtualization

технологія VLAN;
оверлейні мережі VXLAN, GENEVE,
віртуалізаційне ПЗ Cisco PacketTracer, GNS

Віртуалізація накопичувачів

storage virtualization

технологія RAID;
менеджер томів LVM;
файлові системи ZFS, Btrfs

Віртуалізація інфраструктури

infrastructure virtualization

VMware vSphere, Proxmox VE, XCP-ng,
OpenInfra (OpenStack), OpenNebula, CloudStack

3. Огляд технологій віртуалізації

Напрями віртуалізації

Віртуальні машини (означення - на початку лекції)

Гіпервізори I типу (*type 1 hypervisors*), або гіпервізори, "виконувані на голому залізі" (bare-metal hypervisors). Такий гіпервізор не має під собою жодної ОС і, власне, сам є мінімальною ОС.

Приклади: VMware ESXi, Microsoft Hyper-V, Citrix Xen Server.

Гіпервізори II типу (*type 2 hypervisors*) працюють як ПЗ у межах уже встановленої на комп'ютері ОС (основної ОС). ОС на віртуальних машинах - гостьові ОС.

Приклади: Oracle VirtualBox, VMware Workstation, VMware Fusion, KVM.

3. Огляд технологій віртуалізації

Напрями віртуалізації (продовження)

Віртуальні контейнери (пояснення - у п. 2)

Приклади: OpenVZ, Parallels Virtuozzo, Linux-VServer, FreeBSD-jail, LXC, Canonical LXD, Docker.

Віртуалізація прикладних програм (*application virtualization*)

Вся ОС не віртуалізується. Реалізація — віртуальні контейнери або інше віртуальне середовище для кожного екземпляра програми.

Приклади: Citrix Virtual Apps and Desktops, Microsoft App-V, Snap, Flatpak + віртуальні контейнери.

Віртуалізація робочого столу (*desktop virtualization*)

Передбачає віртуалізацію середовища робочого столу, включаючи програми користувача. Це часто (але не обов'язково) передбачає, що користувач взаємодіє з обчислювальними потужностями віддалено, і його процеси виконуються не на пристрої користувача (ПК, ноутбучі, тонкому клієнті, планшеті, смартфоні тощо), а на сервері.

3. Огляд технологій віртуалізації

Напрями віртуалізації (продовження)

Віртуалізацію робочого столу може бути реалізовано на основі двох базових підходів: сервіси віддаленого робочого столу та інфраструктура віддаленого робочого столу.

Сервіси віддаленого робочого столу (*remote desktop services, RDS*)

ґрунтуються на наданні спільного доступу до ОС серверу (одного екземпляру цієї ОС для всіх користувачів).

Приклади: Microsoft Remote Desktop Services, xrdp.

Інфраструктура віддаленого робочого столу (*virtual desktop infrastructure, VDI*) забезпечує кожному користувачу доступ до окремого екземпляру ОС.

Приклади: Microsoft Remote Desktop Services (роль Remote Desktop Virtualization Host), Citrix Virtual Apps and Desktops.

3. Огляд технологій віртуалізації

Напрями віртуалізації (продовження)

Віртуалізація мереж (*network virtualization*)

Робота апаратних мережних пристроїв імітується програмно.

Приклади: технологія VLAN; оверлейні мережі (VXLAN, GENEVE); віртуалізаційне ПЗ (Cisco PacketTracer, GNS).

Віртуалізація накопичувачів (*storage virtualization*)

Об'єднання кількох фізичних накопичувачів в один віртуальний накопичувач.

Віртуальні накопичувачі можуть використовуватися на фізичних комп'ютерах або у віртуалізованих середовищах (наприклад, на віртуальних машинах).

Приклади: технологія RAID; менеджер томів LVM; файлові системи ZFS, Btrfs.

Віртуалізація інфраструктури (*infrastructure virtualization*)

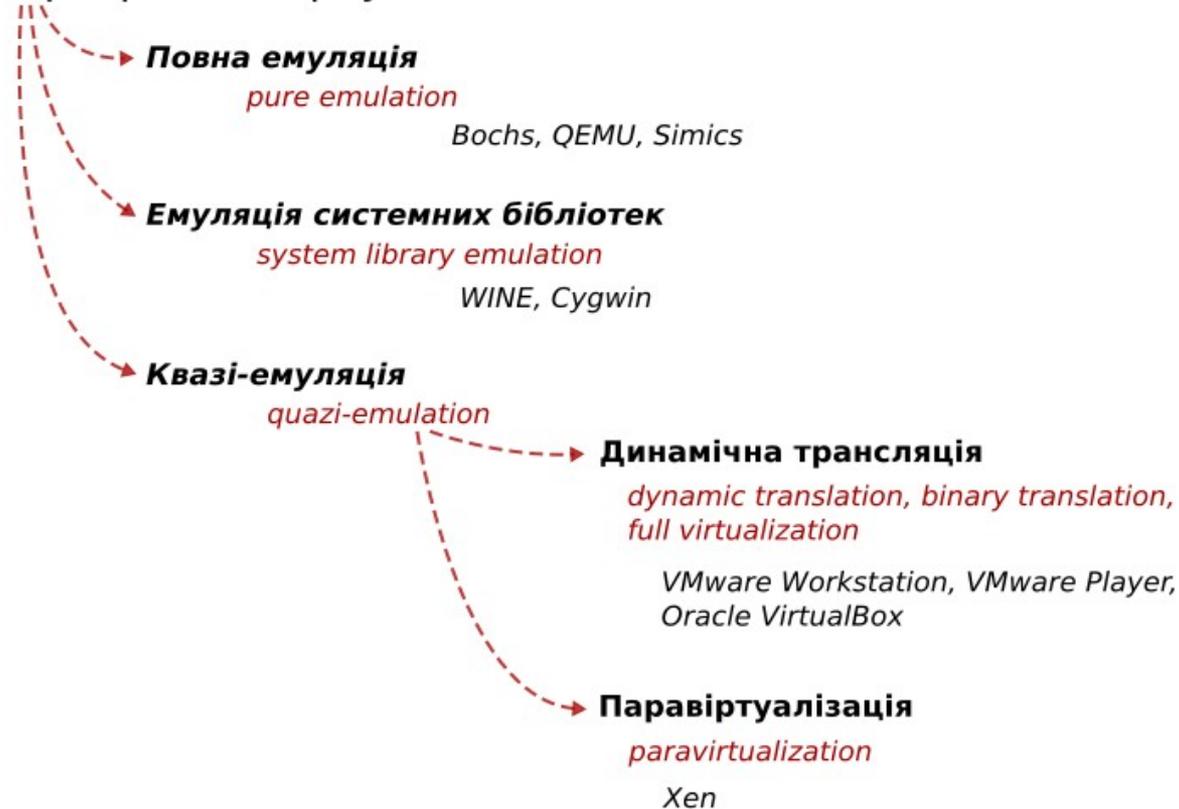
Віртуалізується вся IT-інфраструктура або її частина. Часто використовує одну з моделей хмари, але не обов'язково. Спирається на решту технологій.

Приклади: VMware Sphere, Proxmox VE, XCP-ng, OpenStack, OpenNebula, CloudStack + готові рішення від хмарних провайдерів.

Систематизація технологій віртуалізації

за методом

Програмна віртуалізація



Апаратна віртуалізація

hardware-assisted virtualization

Intel VT-x, AMD-V

3. Огляд технологій віртуалізації

З новенького

- *Хто швидший?*

На практиці технології апаратної віртуалізації за продуктивністю не обов'язково переважають технології програмної віртуалізації.

- *Взаємопроникнення*

Різні напрями і методи віртуалізації поєднуються. Наприклад:

KVM: динамічна трансляція для ОС із закритим кодом + паравіртуалізація для ОС з відкритим кодом

Oracle VM Server для x86: паравіртуалізація для ОС з відкритим кодом + апаратна віртуалізація для ОС із закритим кодом

Virtuozzo: початково - віртуальні контейнери, тепер - контейнери + адаптований гіпервізор KVM

Proxmox VE: віртуальні машини на базі KVM (динамічна трансляція та паравіртуалізація) + віртуальні контейнери на базі Open VZ.

- *Вкладена віртуалізація (nested virtualization)*

Термін, що використовується на позначення можливості запуску одного віртуалізованого середовища всередині іншого.

Приклади: гіпервізори KVM, Xen, VMware ESXi, Microsoft Hyper-V, хмарна платформа Oracle Ravello; VirtualBox (раніше - лише для AMD, тепер і для Intel).

3. Огляд технологій віртуалізації

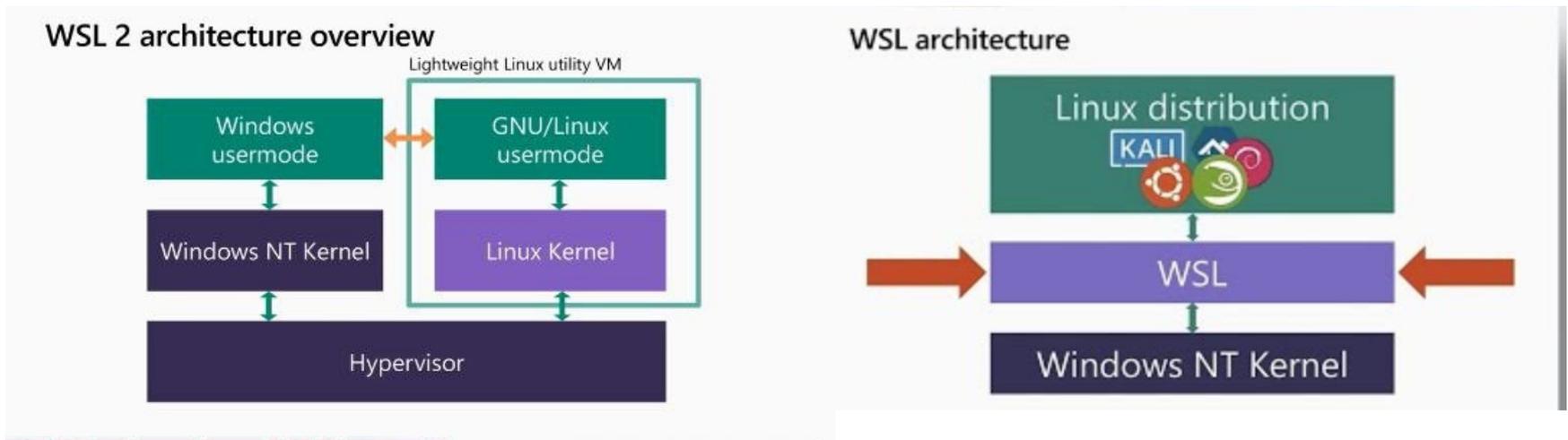
3 новенького

- *Не те, й не те*

Apache Guacamole — безсерверний шлюз для віддаленого доступу до робочих столів.

Windows Subsystem for Linux 2* (WSL 2) — віртуальне Linux-середовище, у якому видозмінене ядро Linux працює на «полегшеній віртуальній машині» (за термінологією Microsoft).

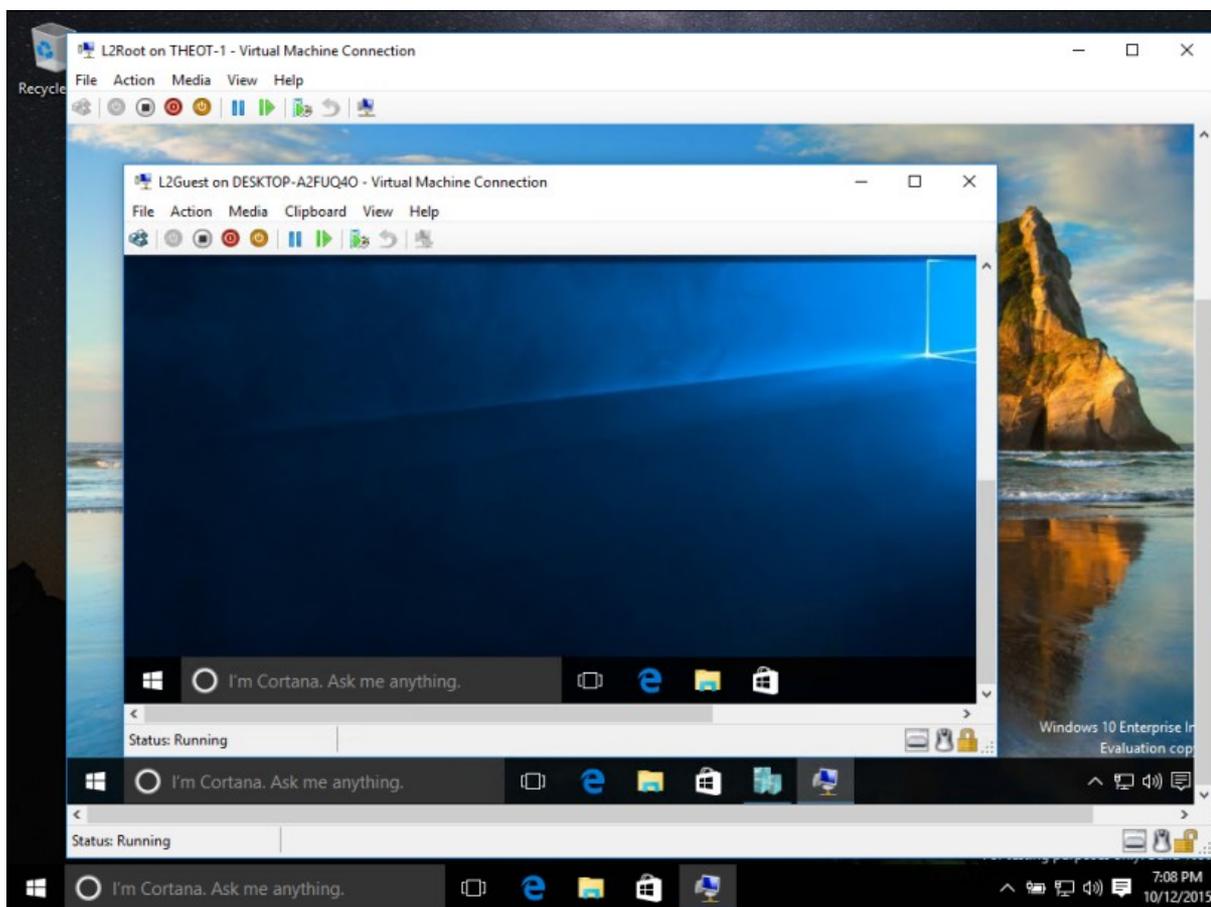
* WSL 1 використовувала іншу технологію, близьку до технологій емуляції системних бібліотек.



Джерело картинок: Microsoft Developer. The new Windows subsystem for Linux architecture: a deep dive — BRK3068. URL: https://youtu.be/lwhMThePdIo?si=cxsBbxh_GO76F9Iq

3. Огляд технологій віртуалізації

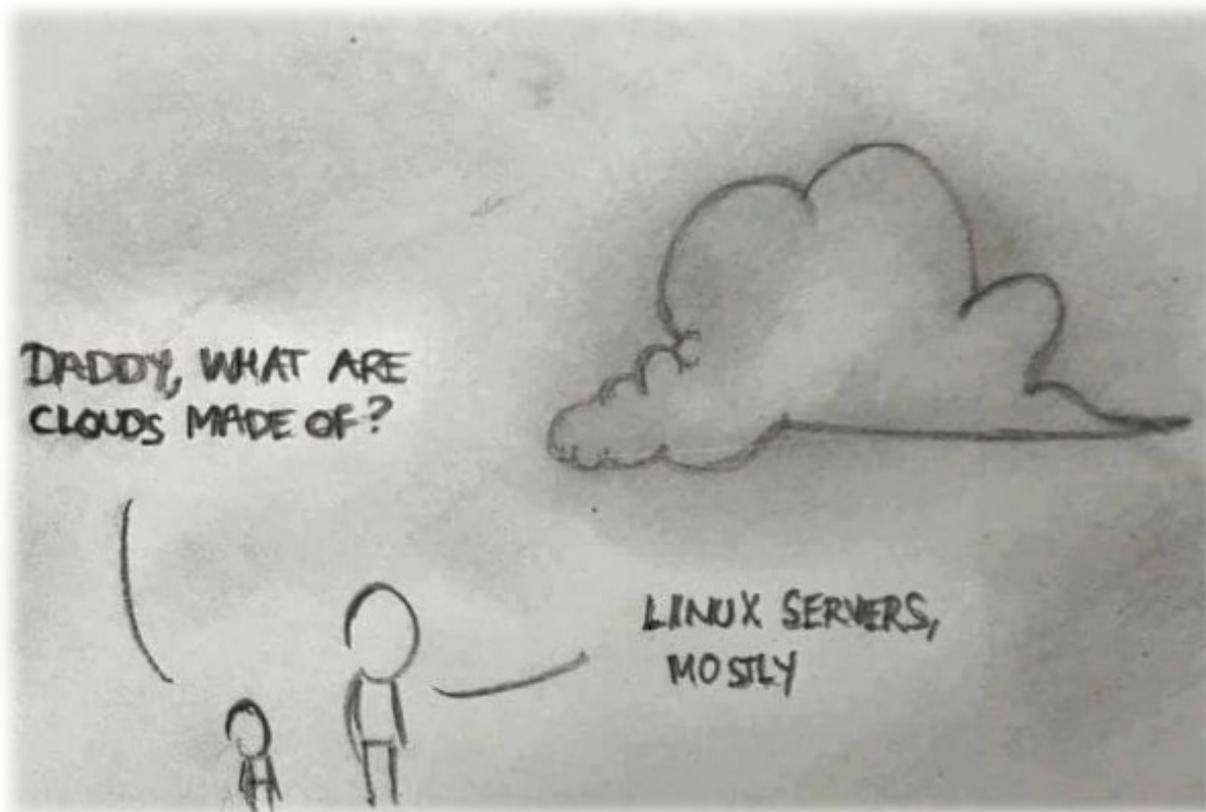
Вкладена віртуалізація (*nested virtualization*) - можливість запуску одного віртуалізованого середовища всередині іншого (наприклад, одна віртуальна машина всередині іншої віртуальної машини).



3. Огляд технологій віртуалізації

До чого тут хмари?

Хмарні технології спираються на різноманітні технології віртуалізації. Часто - на поєднання технологій.



3. Огляд технологій віртуалізації

З означення Національного інституту стандартів та технологій, США (National Institute of Standards and Technology, NIST)

Хмарні технології (хмарні обчислення, cloud computing) - це модель, яка дозволяє повсюдний, зручний мережний доступ на вимогу до області спільно використовуваних налаштовуваних обчислювальних ресурсів (мереж, серверів, сховищ, прикладних програм, сервісів тощо), які може бути швидко надано та швидко вивільнено з мінімальними управлінськими затратами та мінімальною взаємодією з постачальником сервісу.

3. Огляд технологій віртуалізації

Основні характеристики хмарних технологій (за NIST):

- самообслуговування на вимогу (через автоматичні засоби)
- універсальний мережний доступ (доступ з різних пристроїв)
- об'єднання ресурсів (тобто ресурсів постачальника)
- еластичність (обсяг виділених ресурсів динамічно зростає/зменшується)
- вимірюваність обсягів використання сервісу (контроль, моніторинг, оплата pay-per-use).

3. Огляд технологій віртуалізації

Основні моделі надання послуг за допомогою хмари

- **SaaS** (Software as a Service - програмне забезпечення як послуга)
Споживач використовує програмні засоби постачальника.
Споживач не має контролю над інфраструктурою (серверами, операційними системами, сховищами тощо).
Найчастіше споживачами є кінцеві користувачі.
- **PaaS** (Platform as a Service - платформа як послуга)
Споживач розміщує у хмарній інфраструктурі власні програмні засоби.
Споживач не має контролю над інфраструктурою загалом, але може керувати самим програмним засобом, налаштовувати середовище.
Найчастіше споживачами є розробники програм.
- **IaaS** (Infrastructure as a Service - інфраструктура як послуга)
Споживачу надається хмарна інфраструктура з доступом до операційних систем, сховищ, програмних засобів тощо.
Найчастіше споживачами є організації, яким потрібно розмістити у хмарі всю інфраструктуру, та мережні розробники.

3. Огляд технологій віртуалізації

Чотири моделі розгортання хмарної інфраструктури

- **Публічна хмара** (public cloud)
Послуги доступні для більшості населення.
- **Приватна хмара** (private cloud)
Послуги доступні для певної організації.
- **Громадська хмара** (спільна хмара, community cloud)
може використовуватися кількома організаціями, якщо ці організації мають спільні потреби.
- **Гібридна хмара** (hybrid cloud)
Дві або більше хмар, поєднані єдиною архітектурою.

3. Огляд технологій віртуалізації

? А як ми знатимемо, де ми:
на реальній ми машині
чи на віртуальній?



У Linux (один зі способів) - команда `systemd-detect-virt`:

```
ubuntu@ip-172-31-5-206:~$ systemd-detect-virt  
xen
```

Гіпервізор Xen
(хмарний сервіс
Amazon EC2)

```
olena@ubuntu:~$ systemd-detect-virt  
none
```

без віртуалізації

```
root@tryit-curious:~# systemd-detect-virt  
lxc
```

Віртуальний контейнер
LXD (на базі LXC)

```
sysadmin@localhost:~$ systemd-detect-virt  
docker
```

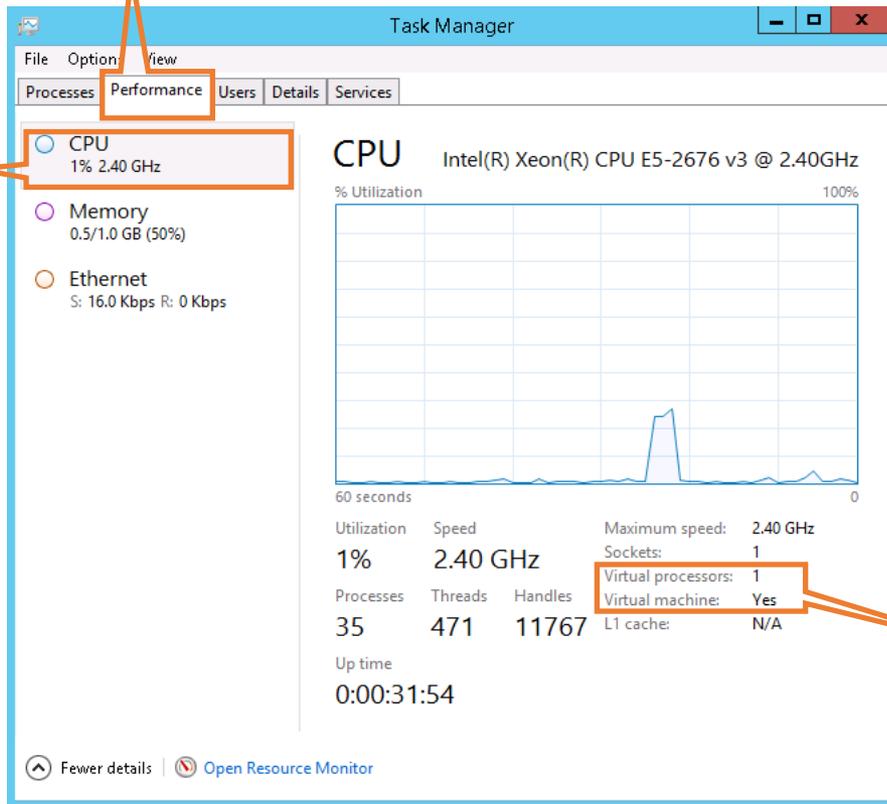
Віртуальний контейнер
Docker

3. Огляд технологій віртуалізації

У Windows - через Диспетчер завдань:

Вкладка Performance / Продуктивність / Производительность

Розділ CPU / Процесор



Базовая скорость:	2,40 ГГц
Сокетов:	1
Ядра:	2
Логических процессоров:	4
Виртуализация:	Включено
Кэш L1:	128 КБ
Кэш L2:	512 КБ
Кэш L3:	3,0 МБ

Апаратну віртуалізацію ввімкнено (але ми на реальній машині)



Maximum speed:	2.40 GHz
Sockets:	1
Virtual processors:	1
Virtual machine:	Yes
L1 cache:	N/A

Ми на віртуальній машині. Віртуальний процесор один.

Поля *Virtual processors*, *Virtual machine*, *Virtualization* - перелік може відрізнитися

3. Огляд технологій віртуалізації

У Windows - через *PowerShell*:

```
PS C:\Users\Administrator> get-wmiobject win32_computersystem | fl model  
  
model : VirtualBox
```

Для самоїтїного читання

1. [**Silberschatz, Galvin, Gagne, 2018**] Chapter 18.
2. [**Stollings, 2017**] Chapter 14.
3. [**Tanenbaum, Bos, 2014**] Chapter 7.
4. *Popek G. J.* Formal Requirements for Virtualization Third Generation Architectures / Gerald J. Popek, Robert P. Goldberg // Communications of the ACM. – 1974. – №7(17). – P. 412-421