Written by Thapanapong Rukkanchanunt

Distributed System

Outline

Distributed System คืออะไร

- Distributed System แปลว่า ระบบกระจาย โดยในระบบนี้คอมพิวเตอร์หลายเครื่องจะทำงานร่วมกัน (คอมพิวเตอร์จับกลุ่มกันทำงานขนาดใหญ่)
- Domain Name Server, Skype, ฐานข้อมูลขนาดใหญ่ หรือแม้กระทั่ง AlphaGo ต่างก็ใช้ Distributed System ในการทำงานทั้งสิ้น

ทำไมต้องใช้ Distributed System

- เพื่อเชื่อมต่อคอมพิวเตอร์ที่ไม่ได้อยู่ติดกันเข้าด้วยกัน
- เพื่อเพิ่มความปลอดภัยด้วยการแยกส่วนออกจากกัน
- เพื่อป้องกันความผิดพลาดด้วยการทำซ้ำ
- เพื่อยกระดับขีดความสามารถของคอมพิวเตอร์ด้วยการประมวลผลแบบคู่ขนาน

สิ่งที่ท้าทายใน Distributed System

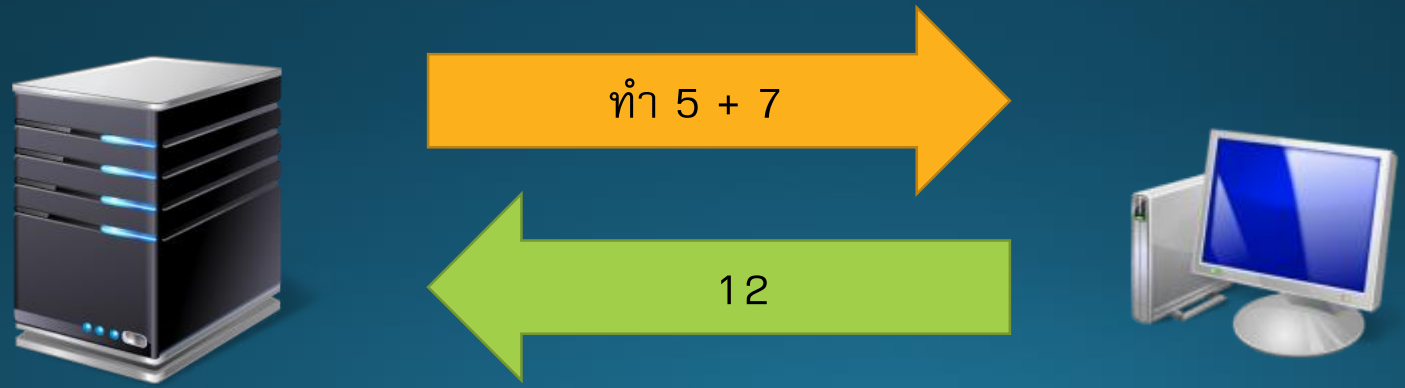
- เป็นระบบที่ซับซ้อน ทุกส่วนทำงานร่วมกันพร้อมกัน
- ต้องมีการจัดการหากส่วนใดส่วนหนึ่งเกิดความผิดพลาดอย่างเป็นระบบ
- อธิบายขีดความสามารถที่เพิ่มขึ้นได้ยาก (ในเชิงตัวเลข)

รูปแบบของ Distributed System

- Distributed System ได้รับการพัฒนาขึ้นมาในหลากหลายรูปแบบในช่วงเวลา 10 ปีที่ผ่านมา ถือได้ว่าเป็นหัวข้อที่ได้รับความนิยมอย่างมาก ในคลาสนี้เราจะเรียนรู้ Distributed System 3 รูปแบบคือ
  - Clients and Servers
  - Peer-to-Peer
  - Cloud Computing

Clients and Servers

- รูปแบบ Clients and Servers เป็นรูปแบบที่ระบบจะแบ่งการทำงานของคอมพิวเตอร์ออกเป็นสองกลุ่มคือ Clients และ Servers โดย Servers จะทำหน้าที่สั่งงานให้กับ Clients ที่เชื่อมต่ออยู่ โดยหลังจาก Clients ส่งผลลัพธ์กลับมาแล้ว Servers จะรวมผลลัพธ์ทั้งหมดเข้าด้วยกัน การสื่อสารในลักษณะนี้จะเรียกว่า Remote Procedure Call



ตัวอย่างระบบ Clients and Servers

- ต้องการบวกเลข 10,000 จำนวน โดยมี Server 1 เครื่องและ Clients 4 เครื่อง

ตัวอย่างระบบ Clients and Servers (2)

- ต้องการบวกเลข 10,000 จำนวน โดยมี Server 1 เครื่องและ Clients 4 เครื่อง

ตัวอย่างระบบ Clients and Servers (3)

- ต้องการบวกเลข 10,000 จำนวน โดยมี Server 1 เครื่องและ Clients 4 เครื่อง

คำตอบคือ SUM1 + SUM2 + SUM3 + SUM4

ปัญหาที่อาจเกิดขึ้น

- คอมพิวเตอร์เกิดความผิดพลาด (เช่นไฟดับ เนตหลุด เครื่องค้าง)

วิธีแก้ไขแบบที่ 1 Redistribution

- พยายามเชื่อมต่อกับเครื่องที่ขัดข้องอีกครั้ง

วิธีแก้ไขแบบที่ 1 (2)

- เมื่อเวลาผ่านไประยะหนึ่งแล้วไม่มีการตอบกลับ ให้เปลี่ยนเครื่อง

วิธีแก้ไขแบบที่ 2 Duplication

- แบ่งจำนวนออกเป็น 4 กลุ่มกลุ่มละ 2500 จำนวน

วิธีแก้ไขแบบที่ 2 (2)

- ถึงแม้เครื่องแรกจะขัดข้อง เราสามารถหาคำตอบได้ นั่นคือ SUM2 + SUM4

เปรียบเทียบทั้งสองวิธี

- ถ้าสมมติว่าการบวกตัวเลขหนึ่งครั้งใช้ระยะเวลา 1 nanosecond เราสามารถคำนวณระยะเวลาในการประมวลผลตามตารางด้านล่าง

	Redistribution	Duplication
เวลารวมเมื่อเกิดปัญหา	2499 + wait + 866 + 5 ns	4999 + 2 ns
เวลารวมเมื่อไม่เกิดปัญหา	2499 + 3 ns	4999 + 2 ns

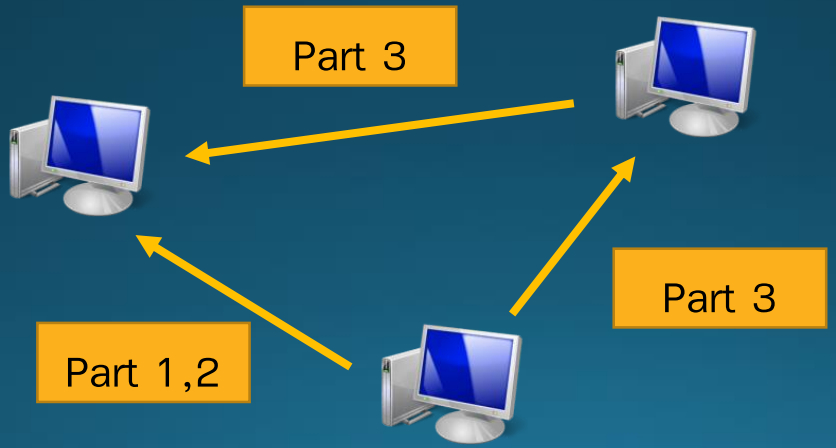
- Wait คือเวลาที่รอการตอบกลับของเครื่องที่ขัดข้องในครั้งที่ 2 ซึ่งมีค่าอย่างน้อย 2499

ปัญหาอื่น ๆ

- Clients แต่ละเครื่องมีประสิทธิภาพในการทำงานไม่เท่ากัน
- Clients อาจจะต้องทำงานให้กับ Servers หลายเครื่อง
- ข้อมูลที่ต้องการคำนวณอาจจะไม่สามารถคำนวณแยกเป็นส่วน ๆ ได้

Peer–to–Peer

- รูปแบบ Peer-to-Peer (P2P) เป็นรูปแบบที่คอมพิวเตอร์แต่ละเครื่องในระบบเชื่อมต่อกันเอง นั่นคือทุกเครื่องทำหน้าที่เป็นทั้ง Server และ Client เหมาะสำหรับการแชร์ไฟล์ขนาดใหญ่โดยทุกเครื่องจะทำหน้าที่เป็นผู้ดาวน์โหลดและผู้อัพโหลด



ตัวอย่าง P2P (BitTorrent)

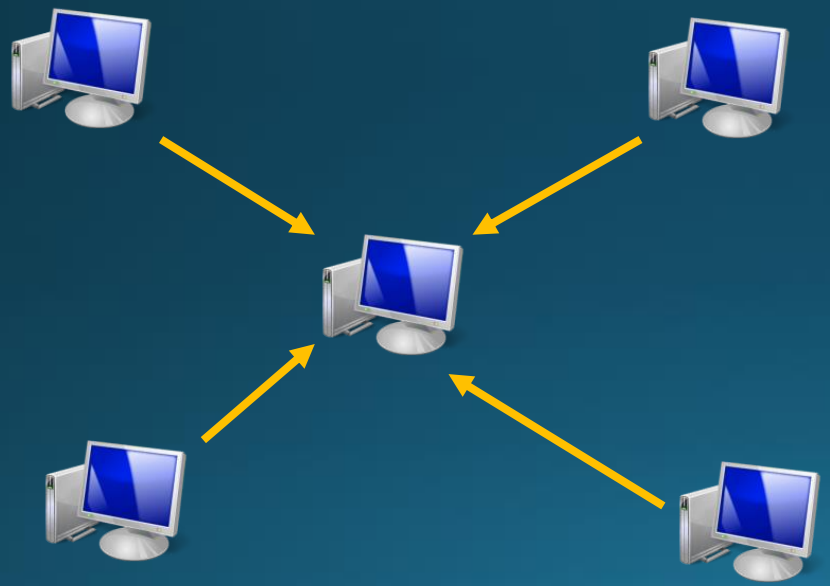
- เมื่อต้องการดาวน์โหลดไฟล์หนึ่งไฟล์ เราจะต้องดาวน์โหลดไฟล์ Torrent ก่อน ซึ่งในไฟล์ Torrent จะมี IP Address ของ Tracker

BitTorrent (2)

- จากนั้นเราจะเชื่อมต่อกับ Tracker เพื่อขอรายชื่อคอมพิวเตอร์ที่มีไฟล์นี้อยู่

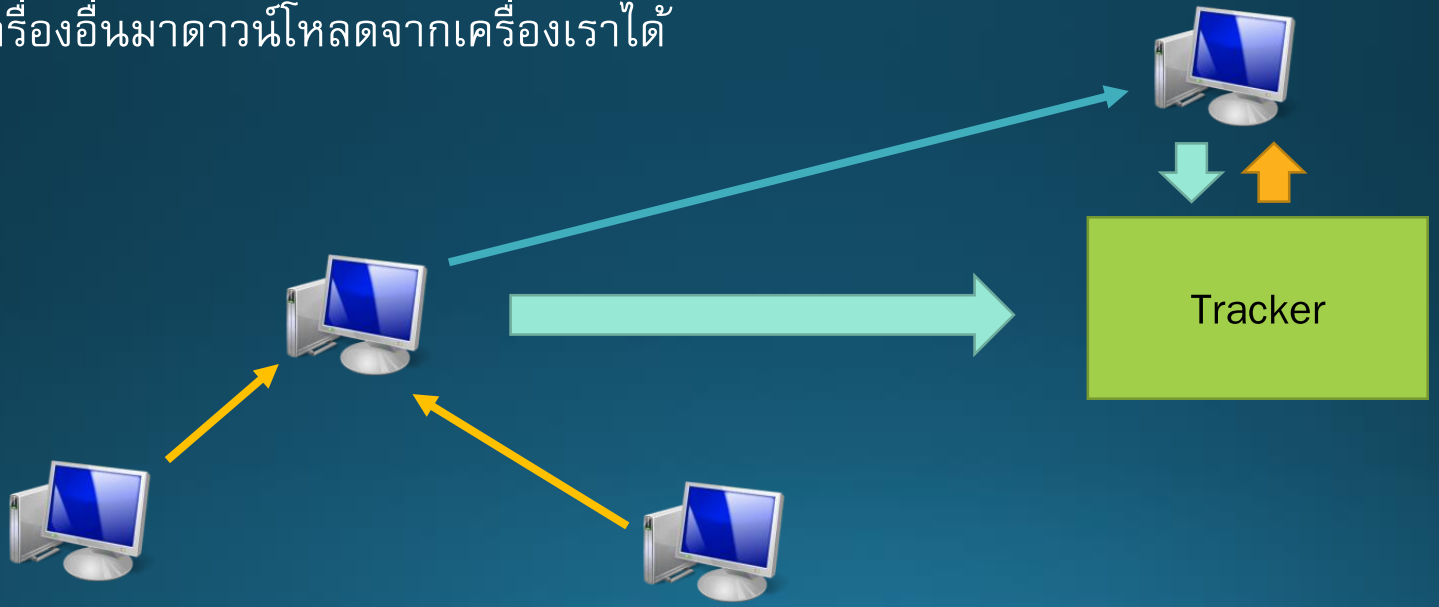
BitTorrent (3)

- เมื่อได้รายชื่อมาแล้ว เราก็เชื่อมต่อกับคอมพิวเตอร์เหล่านั้นเพื่อดาวน์โหลดไฟล์



BitTorrent (4)

- เมื่อการดาวน์โหลดเริ่มต้นขึ้น เครื่องเราก็จะบอก Tracker ว่ามีไฟล์ดังกล่าว เพื่อให้เครื่องอื่นมาดาวน์โหลดจากเครื่องเราได้



ปัญหาที่อาจเกิดขึ้น

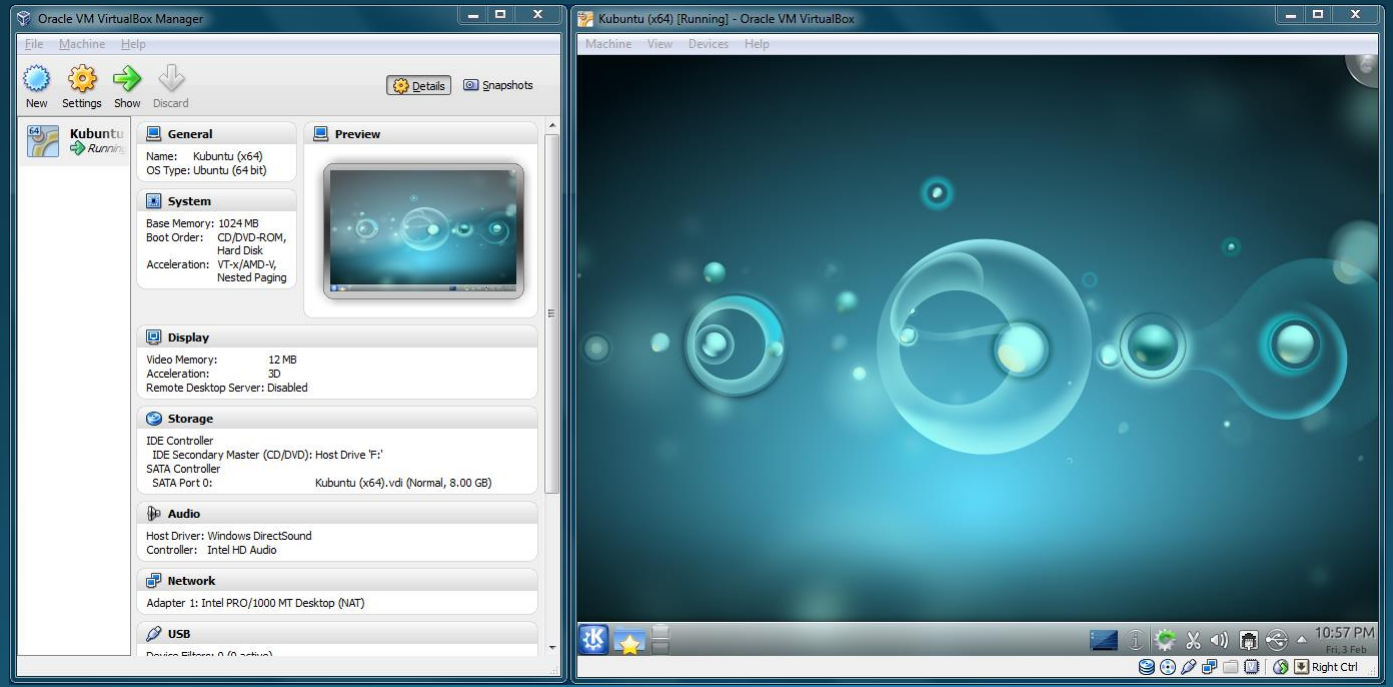
- คอมพิวเตอร์ขัดข้อง (เช่นปิดเครื่อง หยุดรับการเชื่อมต่อ)
  - เปลี่ยนเครื่องง่าย ไฟล์ทุกแบ่งออกเป็นส่วนย่อย ๆ อยู่แล้ว
- ถ้ารายชื่อมีจำนวนมาก ควรจะเชื่อมต่อเครื่องใด
  - เครื่องที่ใกล้ที่สุด หรือ เครื่องที่มีไฟล์สมบูรณ์ที่สุด
- ถ้ามีเครื่องเดียวที่ไฟล์สมบูรณ์ ควรจัดการอย่างไรให้กระจายไฟล์ได้ทั่วถึงและรวดเร็ว
  - อย่าลืมว่าไม่มี Server หลัก ดังนั้นวิธีการแก้ไขต้องเป็นในมุมของเครื่องหนึ่งเครื่องในระบบ
- ความปลอดภัย

Cloud Computing

- Cloud Computing เป็นรูปแบบระบบที่ใช้คอมพิวเตอร์จำนวนมากเป็นทรัพยากร เพื่อให้บริการต่าง ๆ แก่ผู้ใช้งานหรือลูกค้า โดยรูปแบบที่ให้บริการมีดังนี้
  - Cloud Infrastructure ให้บริการคอมพิวเตอร์ประสิทธิภาพสูงผ่านทางเครือข่ายอินเตอร์เน็ต โดยผู้ใช้บริการลงโปรแกรมระบบปฏิบัติการพิเศษ โปรแกรมต่าง ๆ ภายในระบบปฏิบัติการพึ่งพาการประมวลผลจากคอมพิวเตอร์ในระบบ Cloud
  - Cloud Platform ให้ผู้ใช้ทดสอบโปรแกรมที่พัฒนาขึ้นเองในสิ่งแวดล้อมที่ประกอบไปด้วยคอมพิวเตอร์จำนวนมาก ผู้ใช้ไม่จำเป็นต้องเขียนโปรแกรมที่เกี่ยวข้องกับการแบ่งงาน
  - On-demand Software ให้ผู้ใช้ใช้งานโปรแกรมบนระบบ Cloud ผ่านโปรแกรมหน้ากาก ซึ่งการประมวลผลทั้งหมดตลอดจนโครงสร้างของ Cloud จะถูกจัดการโดยผู้ให้บริการทั้งหมด

ตัวอย่าง Cloud Infrastructure

- Oracle VM VirtualBox



ตัวอย่าง Cloud Platform

- Google App Engine อนุญาตให้ผู้ใช้ที่พัฒนา Web Application และ Mobile Application ฝากไฟล์เพื่อใช้งานจริง โดยข้อมูลจะถูกจัดเก็บใน Data Center (มีอยู่หลายที่ทั่วโลก) ซึ่งดูแลโดย Google
- Microsoft Azure เป็นทั้ง Cloud Platform และ Cloud Infrastructure ซึ่งมีบริการในหลากหลายรูปแบบเช่น การสร้าง Web Application การสร้าง Mobile Application การสร้างภาพจากข้อมูล การจัดเก็บข้อมูลสำรอง

ตัวอย่าง On-demand Software

- Google Apps ให้บริการโปรแกรมในหลากหลายรูปแบบผ่านเว็บ เช่น Chat Word Excel PowerPoint Storage