台灣積體電路製造股份有限公司
Taiwan Semiconductor Manufacturing Company, Ltd.

TSMC IT ✕ NCTU CS　課號 5270

CLOUD NATIVE
Development Best Practice

雲原生 (CLOUD NATIVE) 概念與架構

資料及平台部｜吳聲葆 經理
February 18, 2022

AGENDA

Cloud Native Definition

Cloud-native Technology

- Microservices
- Containerization
- DevOps & GitOps
- Cloud-native Platform

Cloud Native Trail Map

References

TSMC IT X NCTU CS 課號 5270

CLOUD NATIVE
Development Best Practice

CNCF Definition

雲原生技術有利於各組織在公有雲、私有雲和混合雲等現代動態環境中，建構和運行可彈性擴展的應用程式。雲原生的代表技術包括容器、服務網格、微服務、不可變基礎設施和宣告式API。

這些技術能建構出容錯性好、易於管理和便於觀察的低耦合系統。結合可靠的自動化手段，雲原生技術使工程師能夠輕鬆地對系統作出頻繁和可預測的重大變更。

雲原生計算基金會（CNCF）致力於培育和維護一個廠商中立的開源生態系統，來推廣雲原生技術。我們經由最先進的民主化的模式，讓這些創新為大眾所用。

*https://github.com/cncf/foundation/blob/main/charter.md
**https://github.com/cncf/toc/blob/main/DEFINITION.md

Benefits of Adopting Cloud-native Computing

Speed, agility and productivity:

- ☐ Organizations can harness the innovation and shorten the time to market by embracing DevOps & GitOps practices via CI/CD (continuous integration / continuous delivery) pipelines.

Scalability and resilience:

- ☐ Cloud-native applications can be automatically scaled to meet surge requests instantly by leveraging the scalability, elasticity, resiliency, and flexibility the cloud service provides.

Cost saving:

- ☐ Resources can be automatically scaled out / in fast as needed serving the dynamic demands without having to allocate fixed size of capacity so as to optimize the usage with lower TCO.

Vendor lock-in reduction:

- ☐ The methodology of building a cloud-native application allows an enterprise to run it on multiple clouds (public or on-premises) using an array of open-source software projects.

Attract and retain top talent, like You!:

- ☐ The cloud-native technology enables developers to focus on value-added business and unleash people's innovations while repetitive operations are automated.

Cloud-native Technology

Cloud native is a methodology of building and running applications which utilize the power of cloud computing. The technology includes:
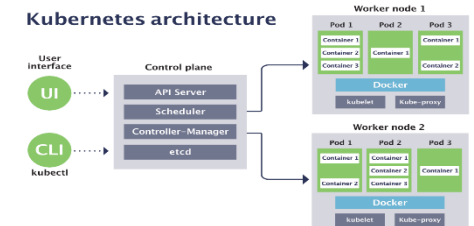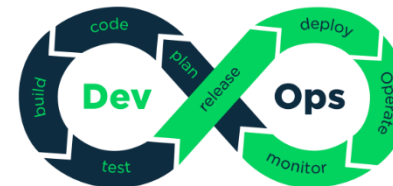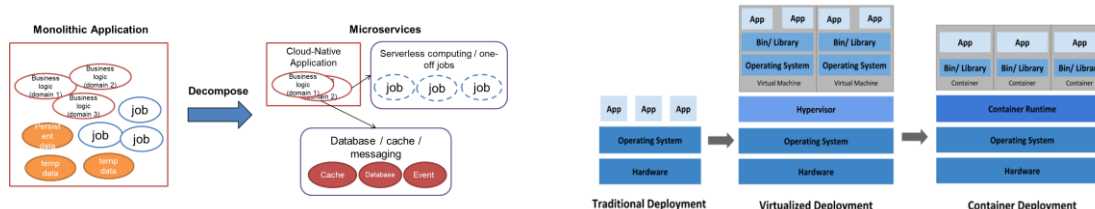
- ☐ **Microservices architecture.**
- ☐ **Containerization.**
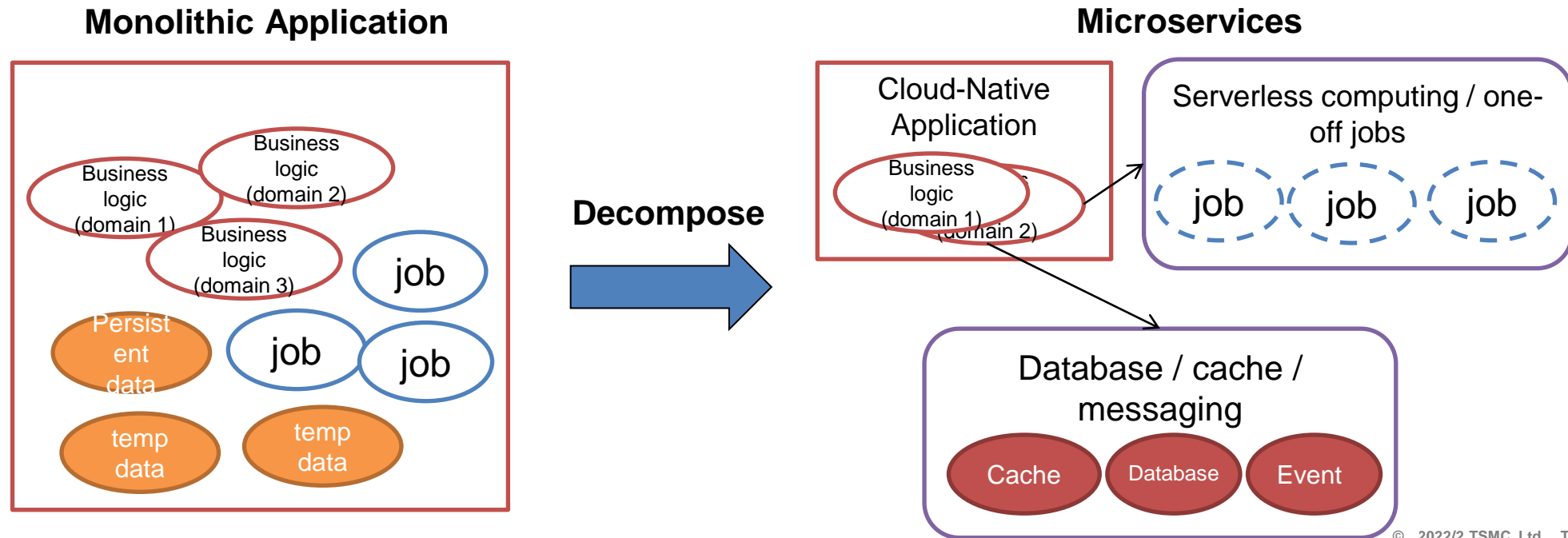- ☐ **DevOps & GitOps.**
- ☐ **Cloud-native Platform (Kubernetes).**

Microservices

A monolithic application contains a lot of business logic (domains), internal job and temp data to result high memory consumption.

Decompose the monolithic application to microservices by domain driven design, and utilize cloud services to auto scale and heal.

Container

A lightweight virtualization technology for developing, shipping, and running applications.

It provides:

- ☐ Fast, consistent delivery of your applications.
- ☐ Responsive deployment and scaling.
- ☐ Running more workloads on the same hardware.

* https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/
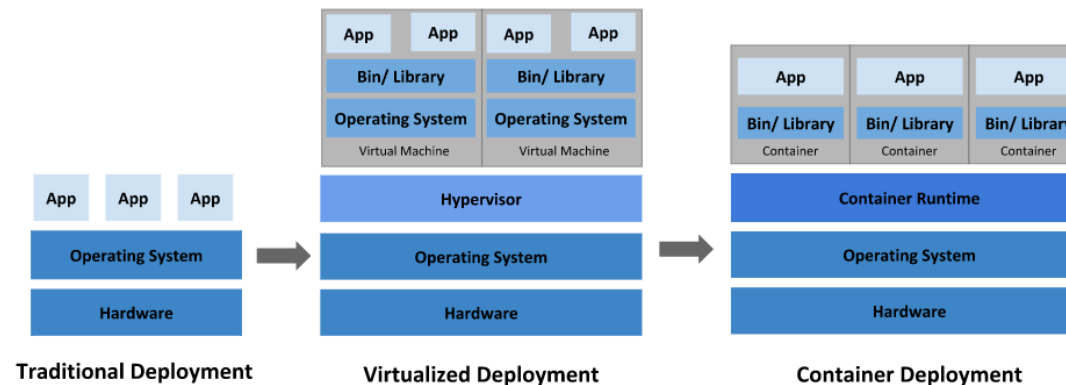
DevOps

DevOps is culture that encourages faster, better application development and faster deliver new features to customers.

- ☐ **Faster, better product delivery.**
- ☐ **Greater scalability and availability.**
- ☐ **More stable operating environments.**
- ☐ **Better resource utilization.**
- ☐ **Greater automation.**

* https://hsissokosio.wordpress.com/devops/

Kubernetes

A tool for deploying & orchestrating the containerized applications

It provides:

- ☐ **Declarative deployment pattern**
- ☐ **Automatic bin packing**
- ☐ **Self-healing**
- ☐ **Service discovery and load balancing**
- ☐ **Secret and configuration management**

* https://www.cncf.io/blog/2019/08/19/how-kubernetes-works/
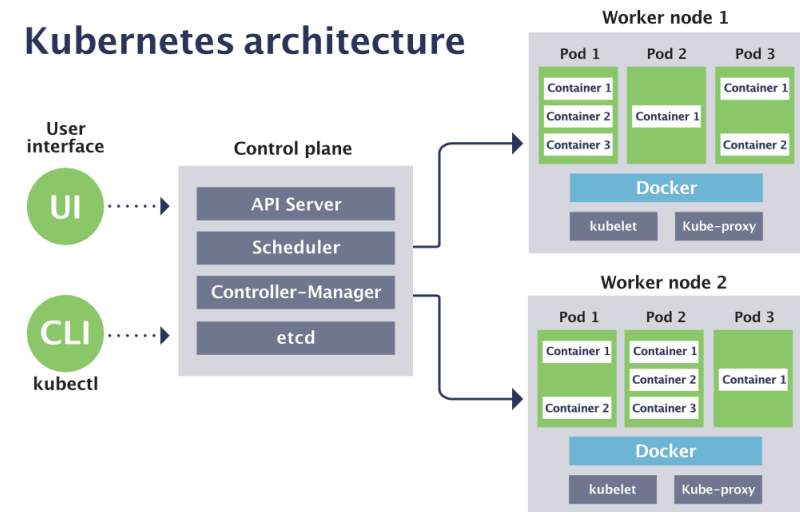
GitOps

A declarative continuous delivery & operation concept that rely on Git as a source control system.

- ☐ **Git is the SSOT(single source of truth) for the desired state of a system. (in yaml)**
- ☐ **Deployment changes become traceable with Git version control.**
- ☐ **Continuously Diff & Sync Git to live system.**

* https://github.com/weaveworks/awesome-gitops
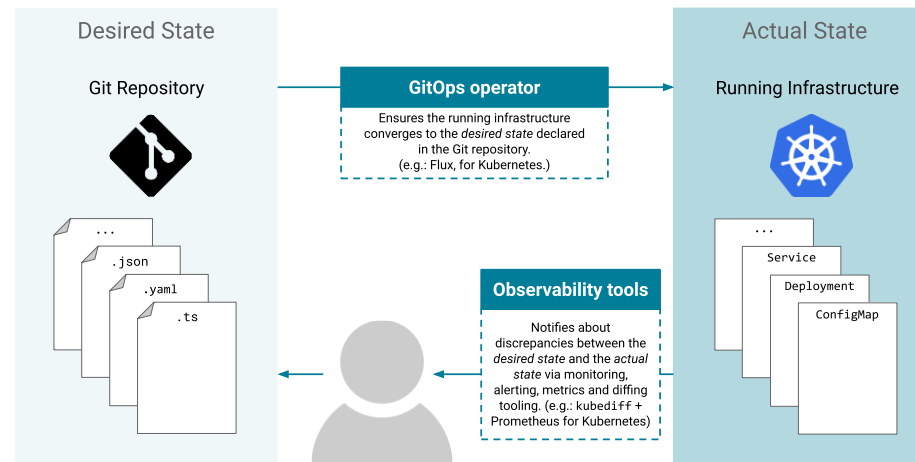
Cloud Native Trail Map

- **Recommended process for leveraging open source software:**

  1. Containerization
  2. CI/CD
  3. Orchestration & application definition
  4. Observability & analysis
  5. Service proxy, discovery & mesh
  6. Networking, policy & security
  7. Distributed database & storage
  8. Streaming & messaging
  9. Container registry & runtime
  10. Software distribution

https://github.com/cncf/trailmap

CNCF Cloud Native Landscape

https://landscape.cncf.io/
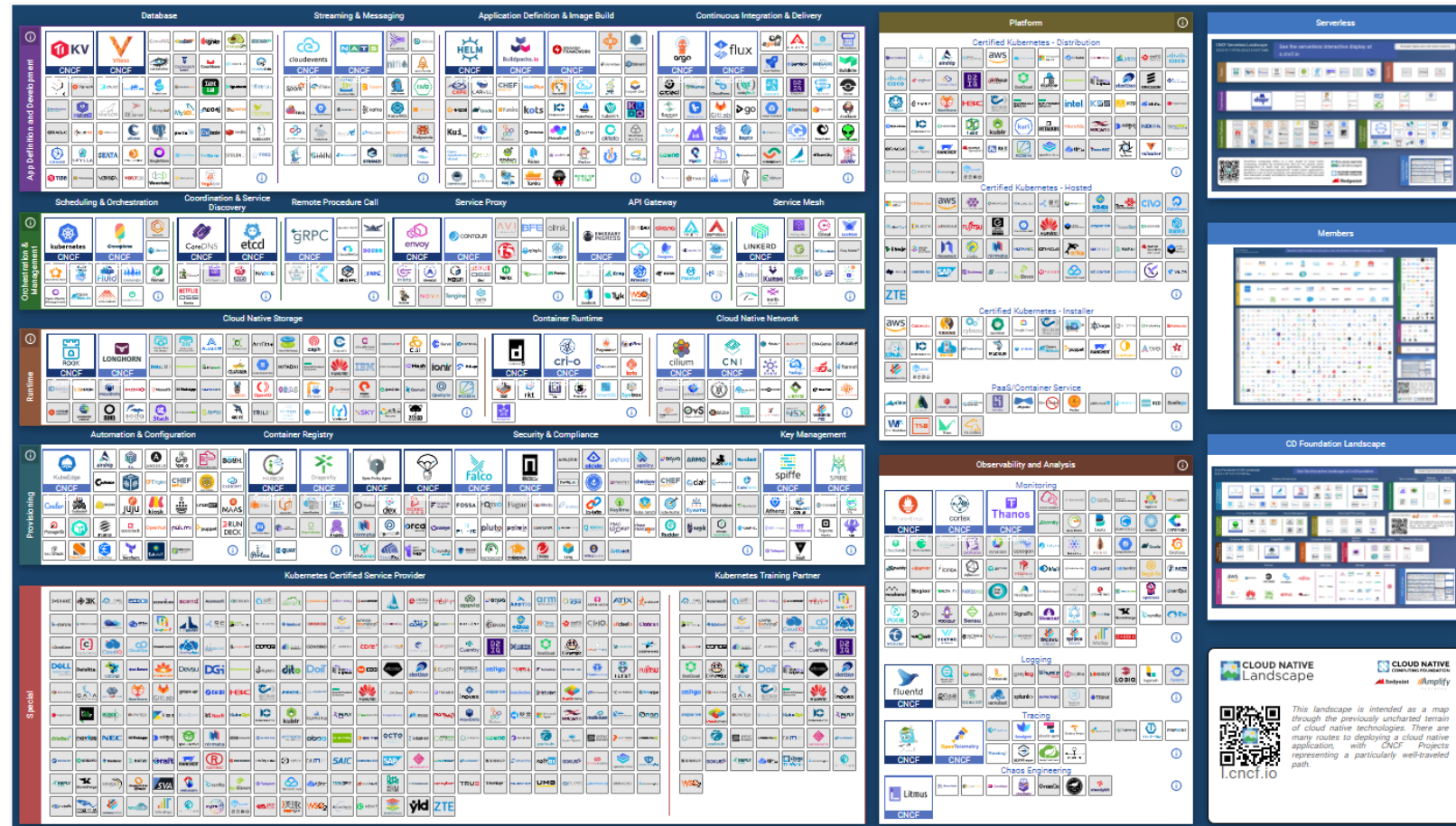
References

The Phoenix Project (鳳凰專案)

https://sre.google/books/

https://www.cncf.io/free-ebook-ci-cd-with-kubernetes/

https://kubereadme.com/
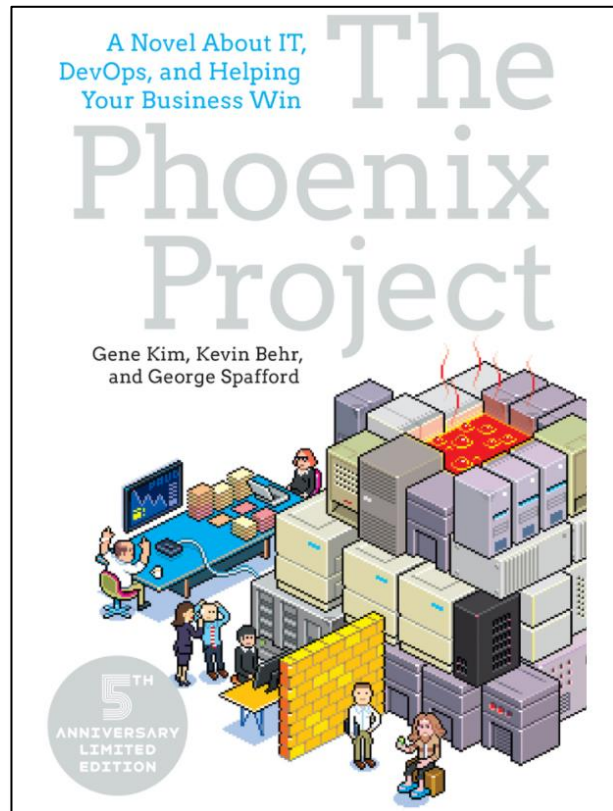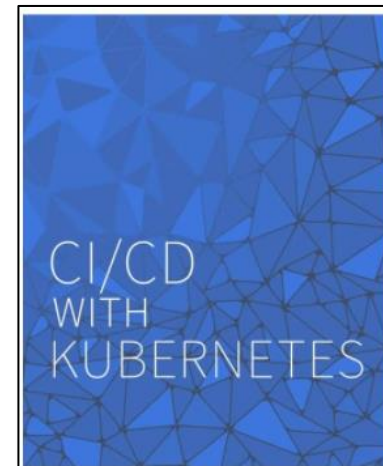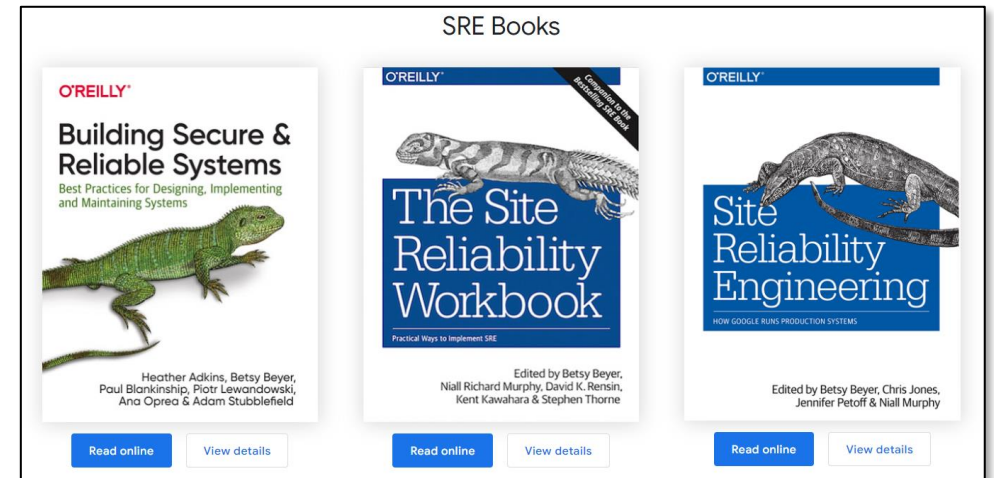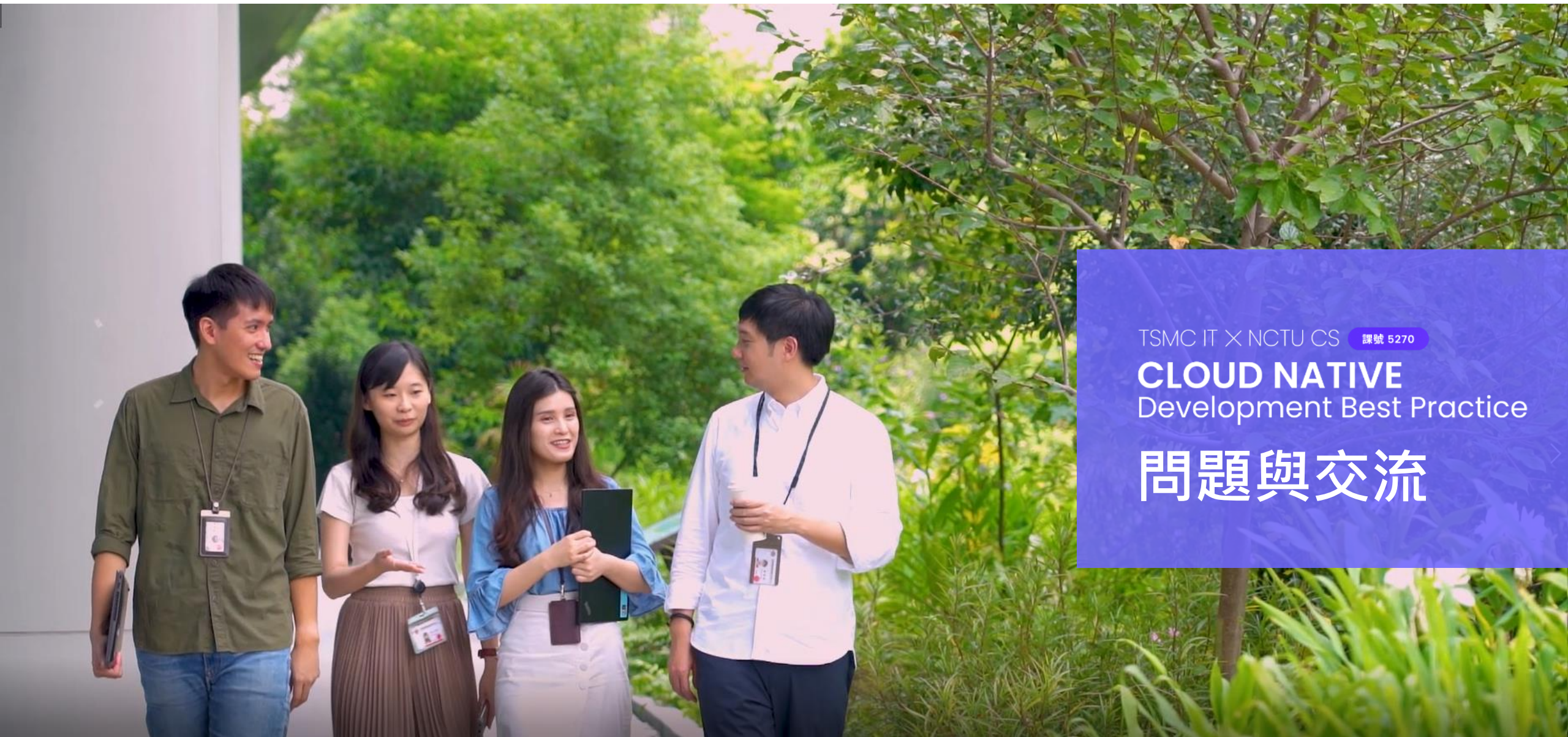
TSMC IT × NCTU CS　課號 5270

CLOUD NATIVE
Development Best Practice

問題與交流

TSMC IT ✕ NCTU CS 課號 5270

CLOUD NATIVE
Development Best Practice

THANK YOU FOR YOUR ATTENTION