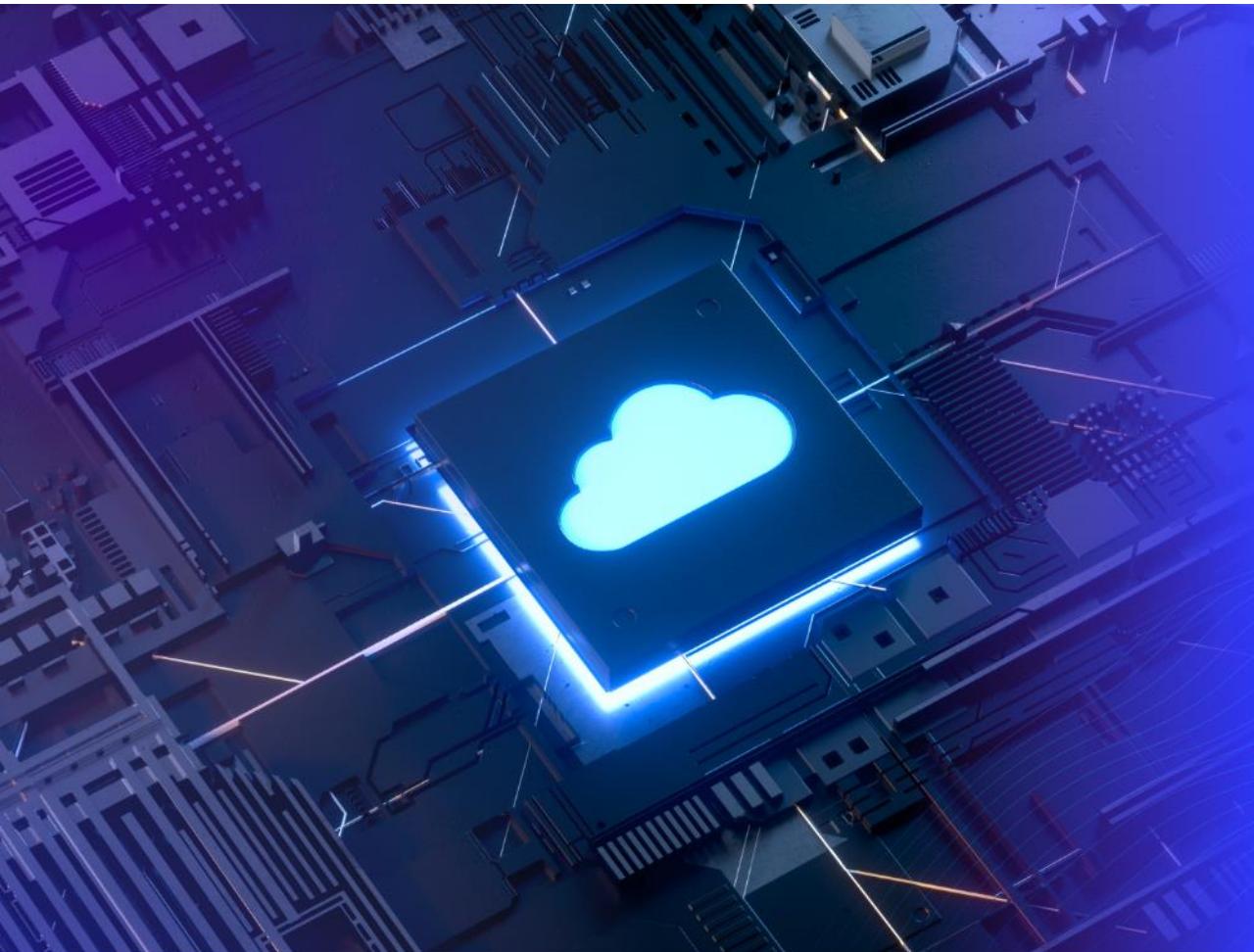




台灣積體電路製造股份有限公司
Taiwan Semiconductor Manufacturing Company, Ltd.



TSMC IT X NCTU CS 課號 5270

CLOUD NATIVE Development Best Practice

雲原生架構與現代軟體 發展趨勢介紹

廠區維運平台部 | 陳蕙猛 部經理
February 16, 2022



TSMC IT X NCTU CS

課號 5270

CLOUD NATIVE
Development Best Practice



AGENDA

台積電簡介

台積電IT介紹

- 智慧台積 數位轉型
- 五大職務
- 訓練課程

2021 Careerhack

台積電×微軟黑客松程式競賽

Our Trinity of Strengths



TSMC IT & NCTU CS Meet up

- Date/Time/Venue: 11/12(Fri) 14:30~15:30@F12 P1#924
- Attendees : TSMC IT x7, NCTU x7

TSMC IT

- IT 林宏達 CIO
- BSID 林均彥 處長
- TSID 陳儒寬 副處長, 胡君怡 部經理
- AAID 沈文冰 副處長
- ICSD 陳守文/鄺國民 副處長
- 謝冬青 HR 經理

NCTU

- 曾煜棋院長 交大AI學院/PAIR中心主任
- 陳添福 交大資訊學院副院長
- 黃敬群 國際學位學程副主任
- 彭文志 交大資工系主任
- 黃俊龍 數據工程與科學研究所長
- 范倫達 資工系/電子資訊中心副主任
- 范瑀真 資工系 宣傳

AGENDA

System introduction

Challenge

Continuous Architecture For DevOps

- Culture/Organization change
- Technology advance
- Continuous architecture principle
- System attributes
- Trade off and decision

Summary

- It's a long road



Who Am I



- A programmer. Start coding from Apple][basic. Implement project by using C, Delphi, VB, Java, C#, JavaScript, typescript, Perl, Python. Study F# now.
- An architect. Try to balance technology, business and resource.
- A Manager. Lead team to do IT DevOps transformation and support tsmc digital transformation.
- Work experience
 - Work in QA Department of International electronic company for 3 years.
 - Work in Startup Internet Company for 2 years.
 - Work as project team leader in SI company for 3 years
 - Join TSMC in 2005, years major work in system development and enterprise architecture.

12 million 12-inch equivalent wafers / year (2020)

2 ~ 5 thousands machines / Fab

1 ~ 2 TB data / day / advanced tool

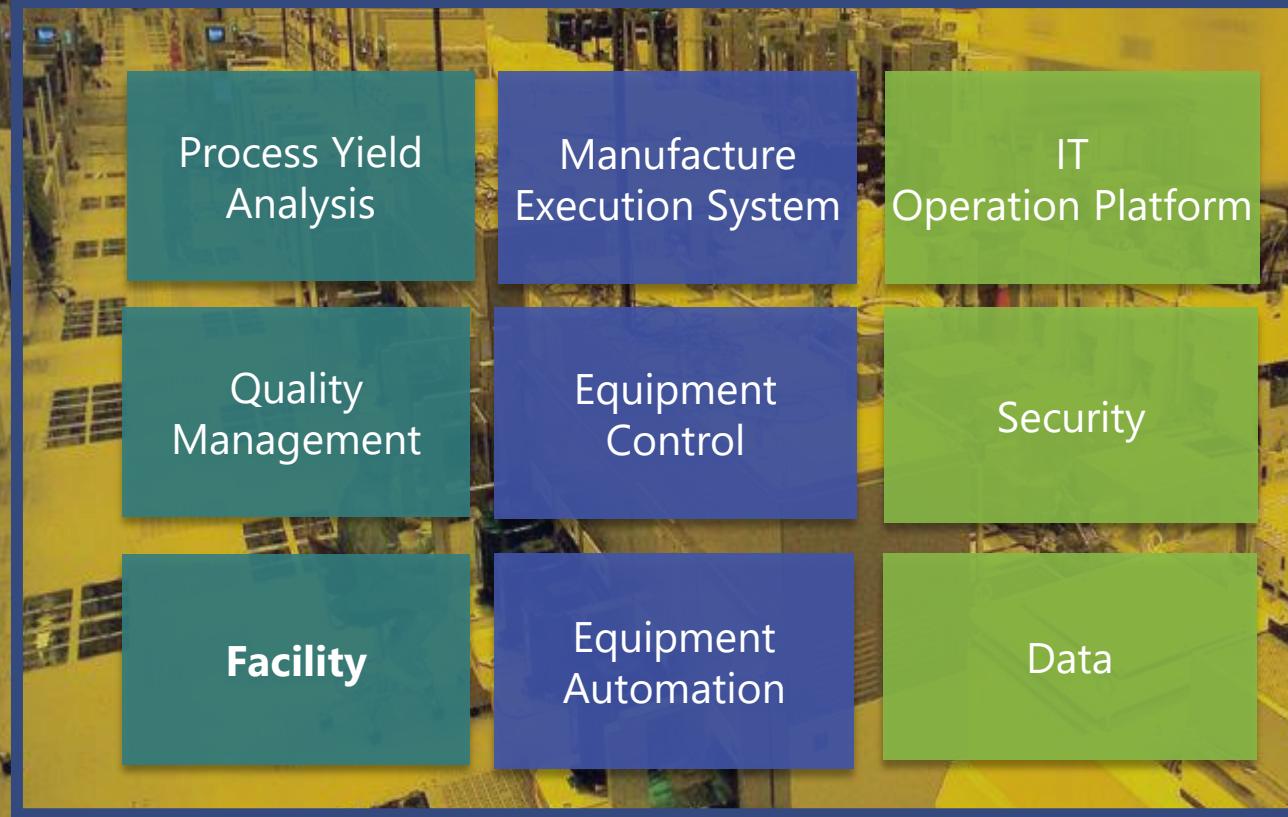
8 ~ 10 million manufacturing instructions / day / Fab

400,000 Km OHT travel distance / day / Fab

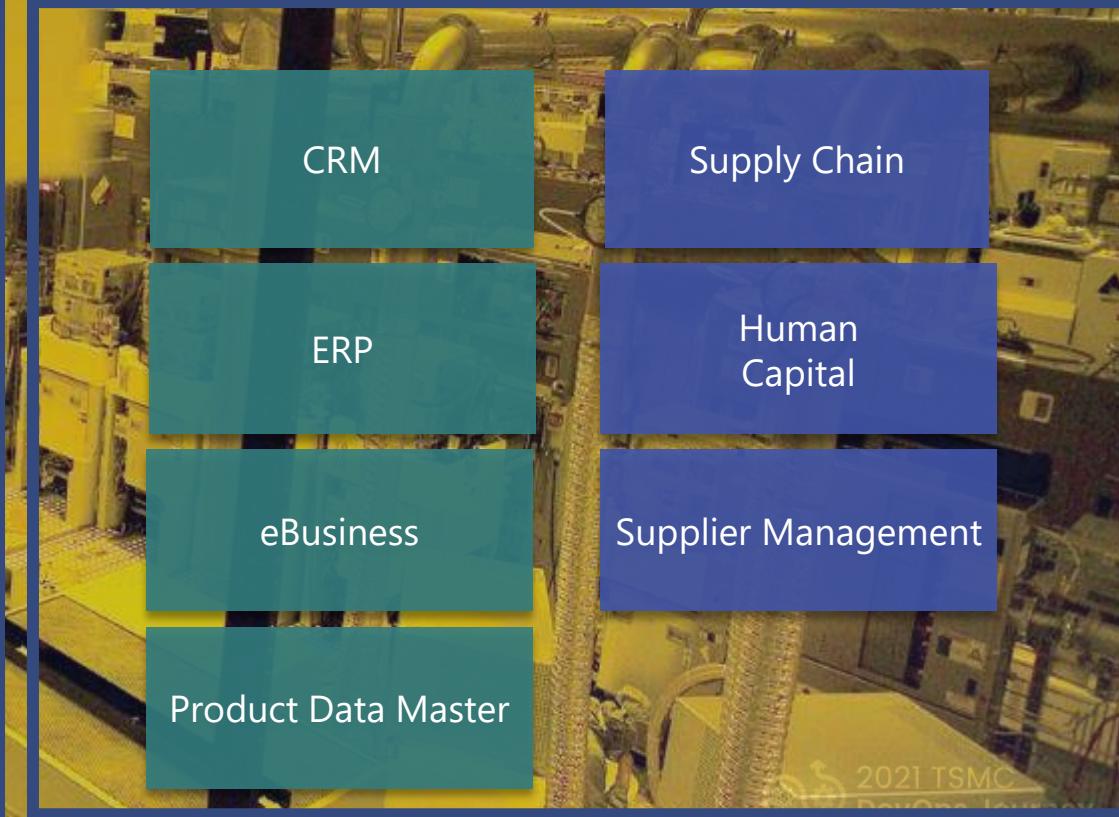
IT Products and DevOps Teaming

- ❑ Software defined Everything in tsmc. Our business runs on IT.
- ❑ IT Digital transformation for future growth.
- ❑ More start up spirit in TSMC IT. Learn new technologies, build innovation systems , and create business impact.

MES, Fab Automation, Yield Analysis



Business Management



Challenge From Global Manufacturing



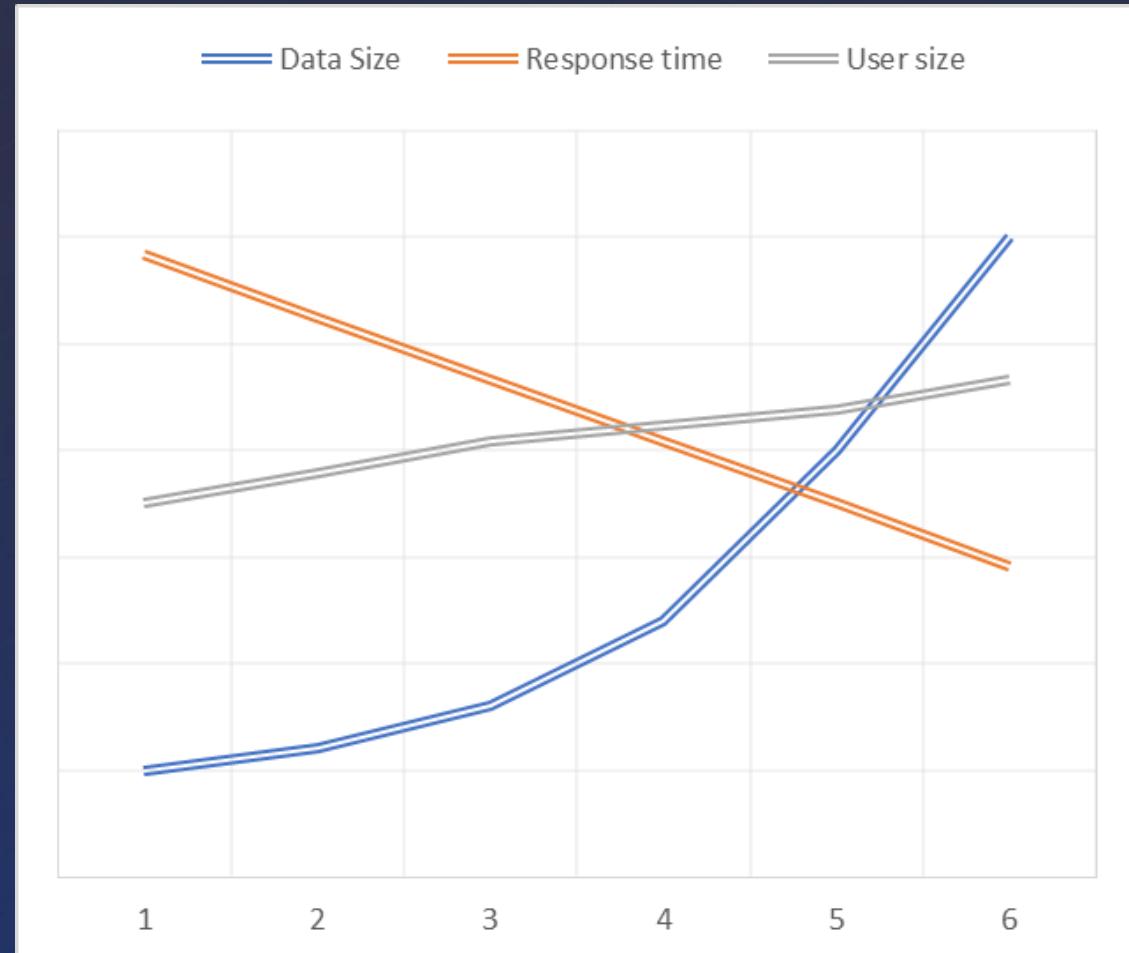
New Fab Construction

- ❑ Build new site operation organization and copy systems from mother fab.
- ❑ Diverse environments and configuration introduce complexity.

```
while (businessGrowth()) {  
    newFab = tsmc.createNewFab();  
    siteOrg = tsmcIT.OperationOrg.create(newFab);  
    newFab.system = siteOrg.copySystemsFrom(motherFab);  
    newFab.system.config = siteOrg.modifyFrom(motherFab);  
    try {  
        newFab.system.run();  
    } catch (exception e) {  
        siteOrg.recover(e);  
    }  
}
```

Challenge of TSMC IT

- ❑ New scalable infra is needed to handle more and more logistic and engineering data for analysis, AI and process control.
- ❑ Data size increase exponentially, user size increase linearly, Response time need to improve to fulfill demand.
- ❑ Complex process and logistic control is key for our yield and throughput. Realtime data process and event handle capability is necessary.



“

Here we have a very interesting challenge that we actually extend the compute model to something even deeper. How do you bring a cloud-native design to be on premise, to be able to scale that it has to run when there's no internet? That takes a lot of good engineering discipline, and you don't get to do that elsewhere. Right? And it's just a very simple example. And also the complexity that we're dealing with: We're applying a lot of machine learning software, we're bringing in the DevOps discipline and SRE discipline from the tech company, and bring it here within TSMC IT.

TSMC's Digital Transformation



林宏達 Dr. Chris Horng-Dar Lin

企業資訊技術副總經理暨資訊長

Vice President, Corporate Information Technology /
Chief Information Officer (CIO)

“

So I think the environment is both challenging but also will stretch your imagination. The demand of your quality of your engineering output is going to be on par with a top-tier tech company of the world. So I think it's really interesting projects that you'd get to work on. And I think more importantly, we're going through a digital transformation right now with a very clear plan and a vision, and I think you will love to be part of that. Because when TSMC go through this and come out on the other end, it will continue to grow, and grow in a way much more efficiently, and you'll be very proud you are part of that team that make it happen.

YOUR CAREER @ TSMC IT

林宏達 Dr. Chris Horng-Dar Lin

企業資訊技術副總經理暨資訊長

Vice President, Corporate Information Technology /
Chief Information Officer (CIO)



IT Engineer DevOps Change

Chaos Engineering

On-Prem
Cloud Native
Platform

OBSERVABILITY

SERVICE MASH

CODING

TO DO
EVERTHING

INFRA API

DATA CENTER
RETROFIT

NETWORKING

AUTOMATION

INFRA SERVICE



RECOVERY DRILL

METRICS

CI/CD



SLA/SLO

MICROSERVICE

CODE REVIEW

REALTIME
MESSAGE

AGILE PLAN

API FIRST

DELIVER
VALUE
BETTER
MANAGEMENT

SUSTAINABLE
EXPANSION

CHANGE
CONFIDENCE

BETTER
QUALITY

REVIEW & REVIEW

DEADLINE DRIVEN DEVELOPMENT

USER URGENT REQUEST

PUSH SCHEDULE

OKR

product
Design



SHORT-LIVE, UNSUSTAINABLE

Projects

PRIORITY

Technology Ch

Value Syst

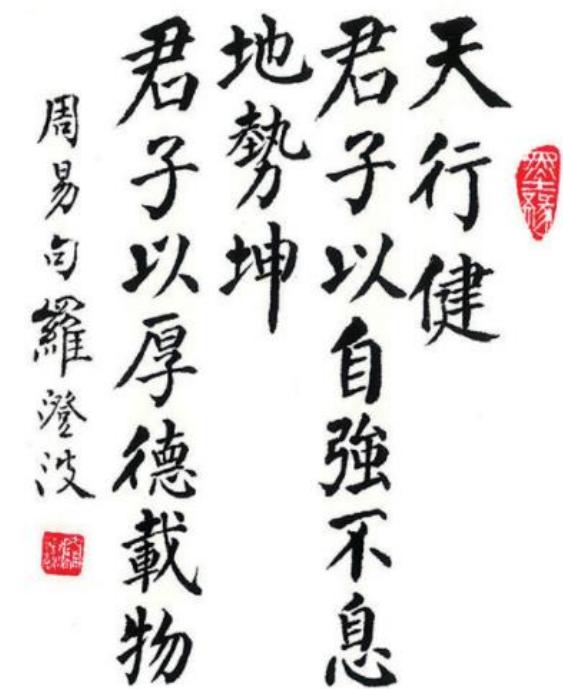
Culture Cha

Continuous Architecture Principle

- Business is continuous growth, IT systems need to growth along with business.
- Design systems for change, change for business growth.
- IT need continuous everything.
- The Six Principles of Continuous Architecture
 - Architect products, not projects
 - Focus on Quality Attributes, not on functional requirements.
 - Delay design decisions until they are absolutely necessary.
 - Architect for change—leverage “the power of small.”
 - Architect for build, test, and deploy.
 - Model the organization after the design of the system.
- These are six principles of Continuous Architecture

天行健
君子以自強不息
地勢坤
君子以厚德載物

周易句 羅澄波



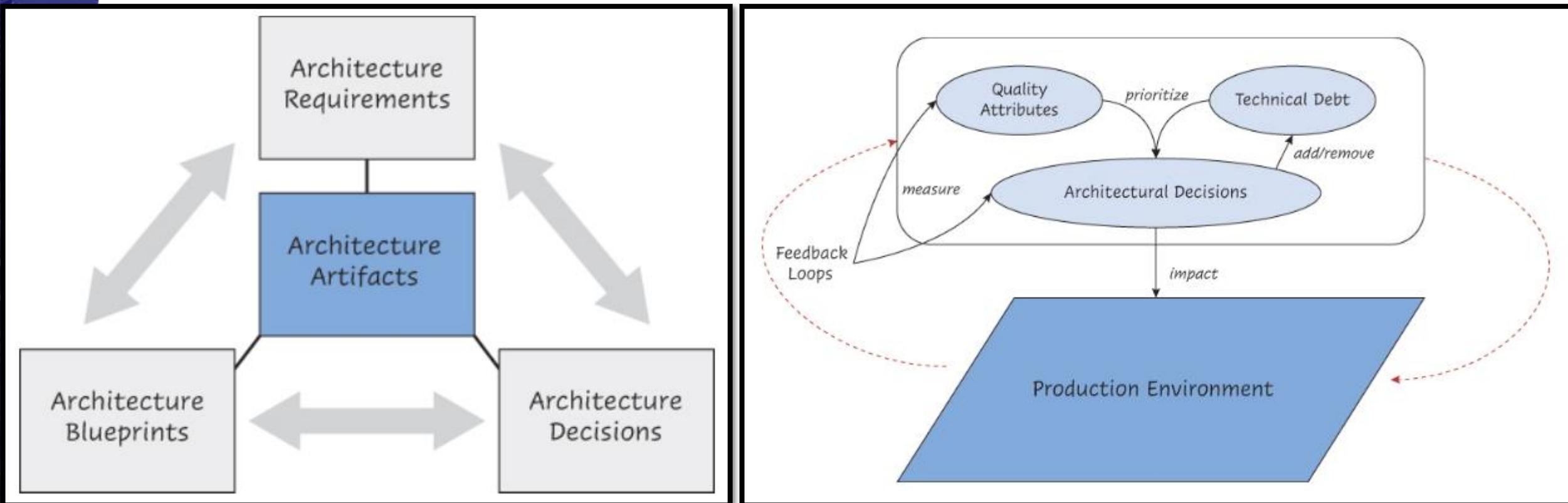
Continuous Architecture

Architecture design is not just about coding, it's about communication and collaboration.

Technical Debts, Architecture Decision, Quality Attributes.

Without proper architecture design and growth, the tech debts will hamper down the system and business growth.

Architecture is all about trade-off and decision making.

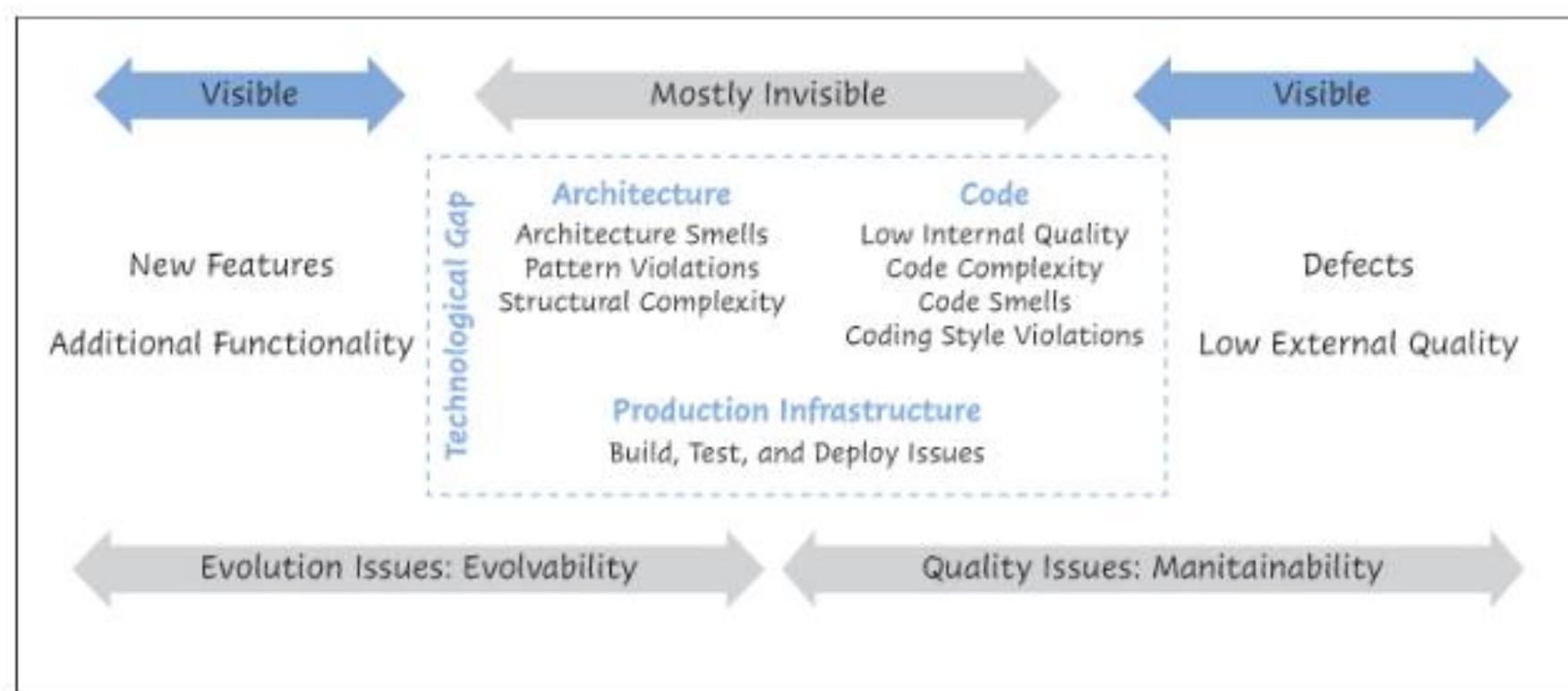


Quality Attributes Of IT Systems

- **Functional Suitability:** Completeness, Correctness, Appropriateness
- **Performance Efficiency:** Time behavior, Resource utilization, capacity.
- **Compatibility:** Coexistence, interoperability.
- **Usability:** Learnability, user error protection, accessibility...(a lot of stuff)
- **Reliability:** Availability, Fault tolerance, Recoverability.
- **Security:** Confidentiality, Integrity, Nonrepudiation, Authenticity, Accountability.
- **Maintainability:** Modularity, Reusability, Analyzability, Modifiability, Testability.
- **Portability:** Adaptability, installability, Replaceability.

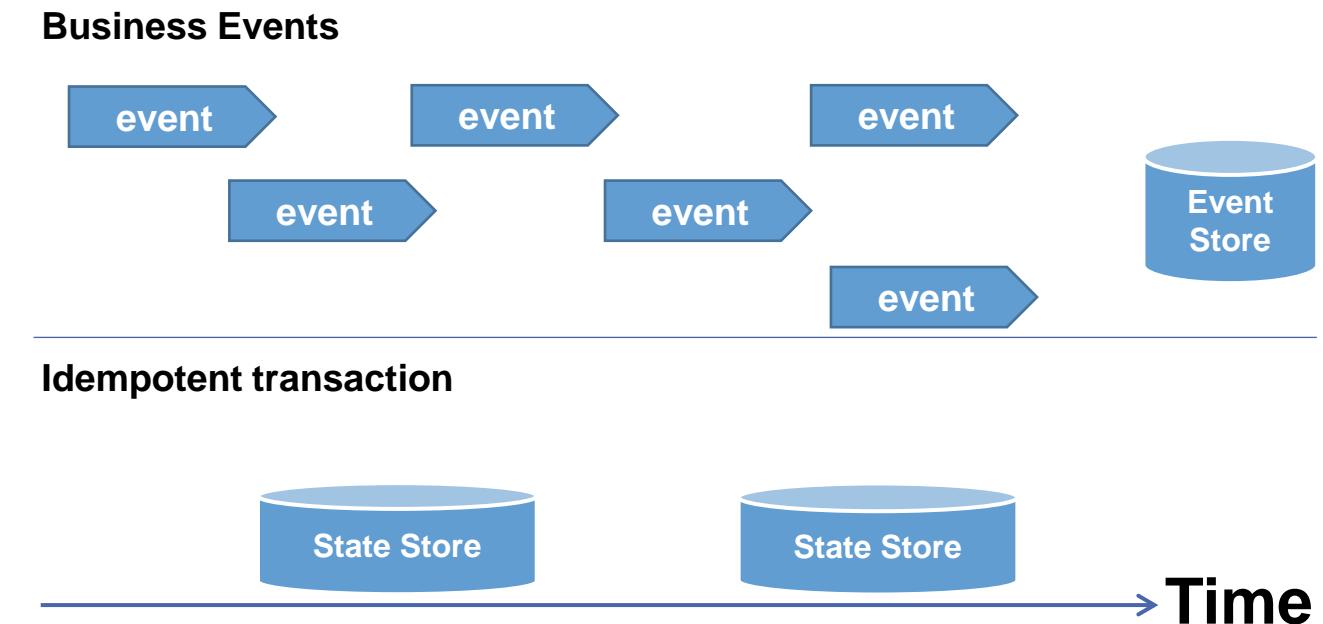
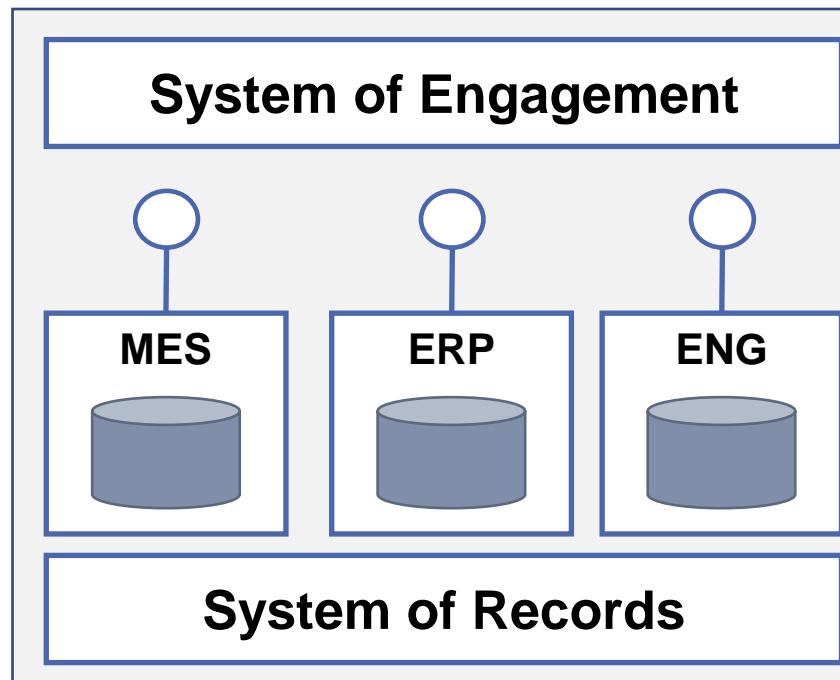
How Technical Debts Impact Growth

- So what happens if the interest we owe on our technical debt starts to exceed the number of man-hours in a working day? *Technical Debts singularity point. (Alex Yates 2015)*



Data Architecture

- Common Language / Ubiquitous Language / Domain Language
- NoSQL and Polyglot persistence
- Scale and availability, Eventual consistency
- Event vs State, Event sourcing
- Data Analytics, wisdom and knowledge from information



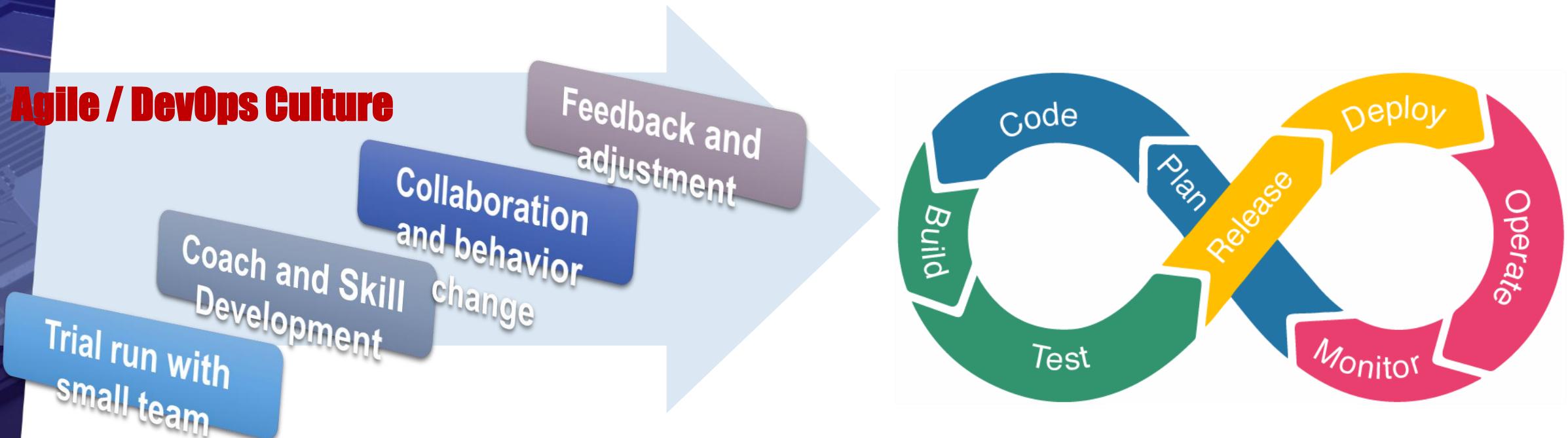
Technology Trends in TSMC

Semiconductor is a booming age, as well as software and IT. Why not join us to explore the possibilities.

- Cloud Native
- Distributed high volume transaction
- Tool Automation
- Microservices
- Big Data & AI
- Digital transformation

Continuous Architecture

- 2019 - Dev teams consolidated, Ops teams aligned
- 2020 - Start DevOps, Scrum practice on selected products and CI/CD adoption
- 2021 - Transform Ops team with SRE practice and fully adopt Product DevOps



On-Prem Cloud, Auto CI, Auto CD, EFK, Kiali, Jaeger, Message Systems

Infra Service Automation

Plan – Feature to Code

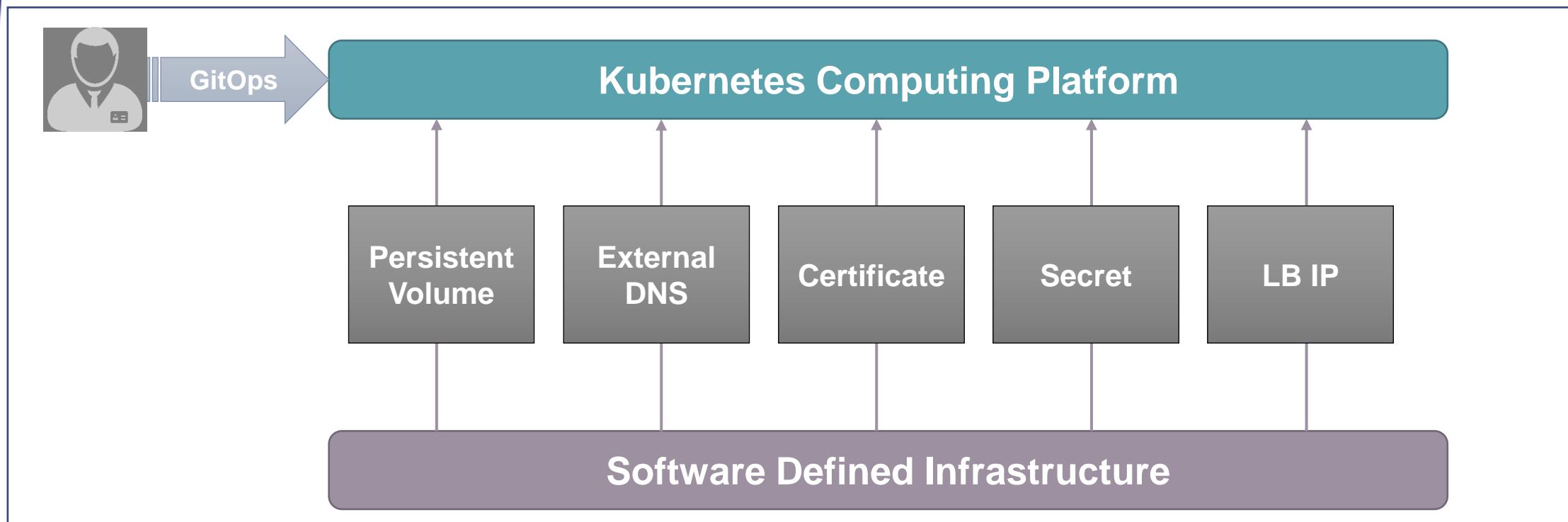
- Transform product roadmap and user feedbacks into backlogs
- Show backlogs relationship with other backlogs and due date can help team to understand what product is heading and what progress is.
- According to release plan, team members will decide what backlogs should go into each iteration.
- Development strategy – git flow enhance version
 - Create branch for each backlog/task. Merge back to dev branch with pull request
 - Decrease merge conflict
 - Continue testing code changes
 - release/main branch for pi-run & production release

Quality – PR, UT & Code Review

- When pull request is created, it will automatic assign reviewer(s) and trigger CI pipeline for testing/linting tasks.
- Reviewer will review pull request to make sure code is ready for merge. At the same time, reviewer will use PR to help junior to improve their coding skill.
- When PR is merged, it will trigger another CI pipeline for testing report 、 coverage report 、 fortify 、 white source and publish image.
- After CI is done, it will trigger CD pipeline for DEV staging deployment.

Platform Service Auto Provisioning

- In addition to K8s compute resources, engineers should be able to invoke IaaS providers declaratively to acquire services required to make applications runnable, discoverable and accessible.



Management – Release, Metrics, Feedback

Planning / Retrospective session



Daily Meeting



UT coverage / PR / Code Review



Release Process



New Fab Construction in DevOps

- Aligned on-prem cloud native platform to host all our applications.
- Consolidate IT product DevOPs team take responsibilities for new Fab construction & Operation.
- Global development, release, operation and monitoring.
- Continuous system growth to support business growth.

```
while (businessGrowth()) {  
    newFab = tsmc.createNewFab();  
    for (product in ITProducts) {  
        product.deploy(newFab);  
    }  
    newFab.run();  
}
```

Summary

- Consolidate Dev and OS team into one product DevOps team to take responsibility of product and customer success.
- Align Infrastructure into home built on-prem cloud as base of applications.
- Align CI/CD approach, adopt Git flow, test driven and code review discipline to improve code quality.
- IT engineer get the following
 - Learn new skills to solve their own business problems.
 - Change role from System admin/operator to software engineer.
 - Bootcamp for new comer and traditional site engineer.
- Business Benefit
 - Shorten Fab system deployment and alignment cycle time.
 - Faster feature delivery and feedback.
 - Aligned user experience by using single fab operation system.

Reference Material

- **Continuous Architecture in Practice: Software Architecture in the Age of Agility and DevOps.** Eoin Woods, Murat Erder, Pierre Pureur. (2021)
- **ISO/IEC 25010:2011 systems and software engineering – systems and software quality requirements and evaluation.** – SQuaRE Models (2011)
- **The Technical Debts Singularity Observations.** Alex Yates (2015)
- **TSMC IT DevOps Presentation** (2021)
- **TSMC IT Campus hiring presentation** (2021)



TSMC IT X NCTU CS

課號 5270

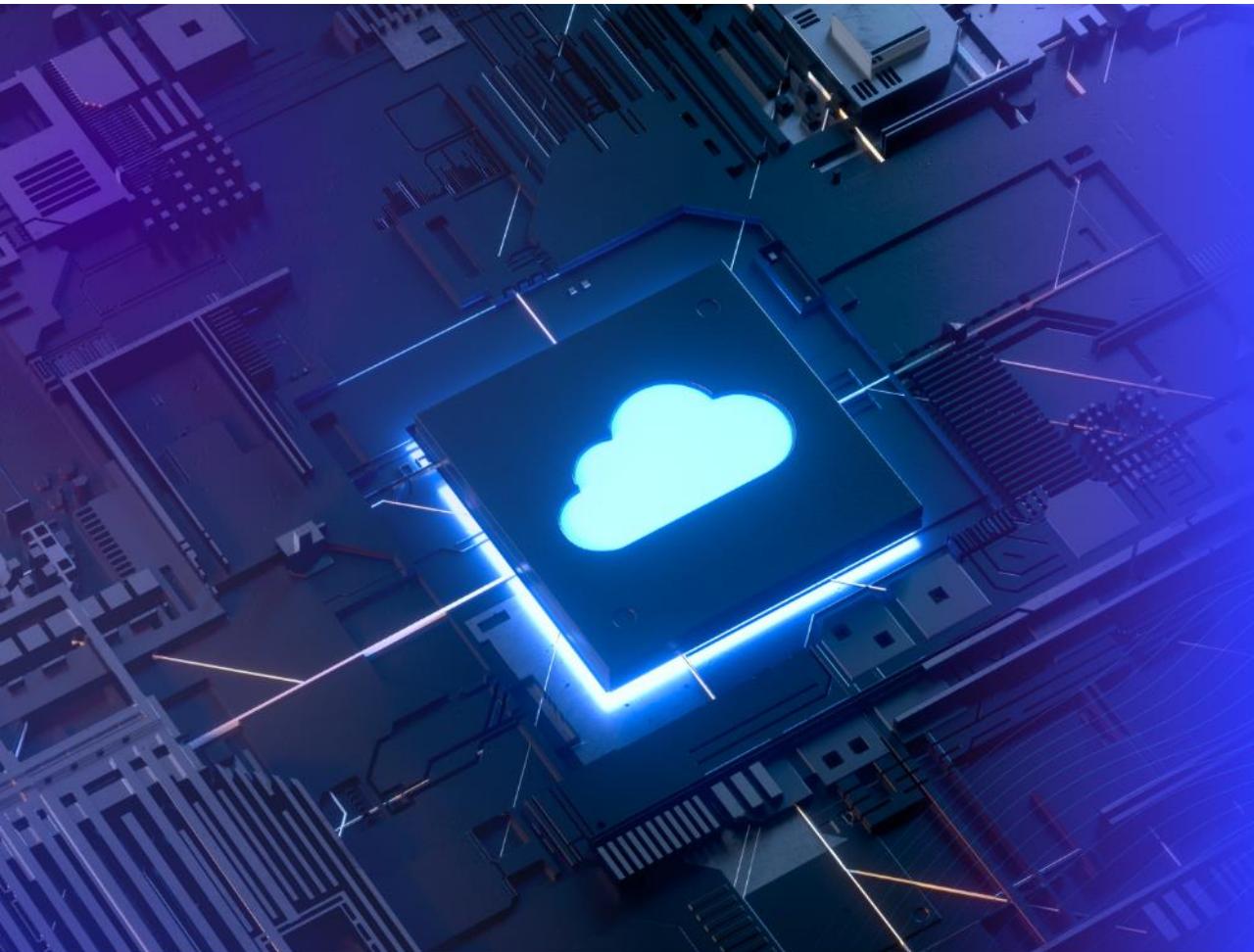
CLOUD NATIVE Development Best Practice

問題與交流





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



TSMC IT X 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

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.

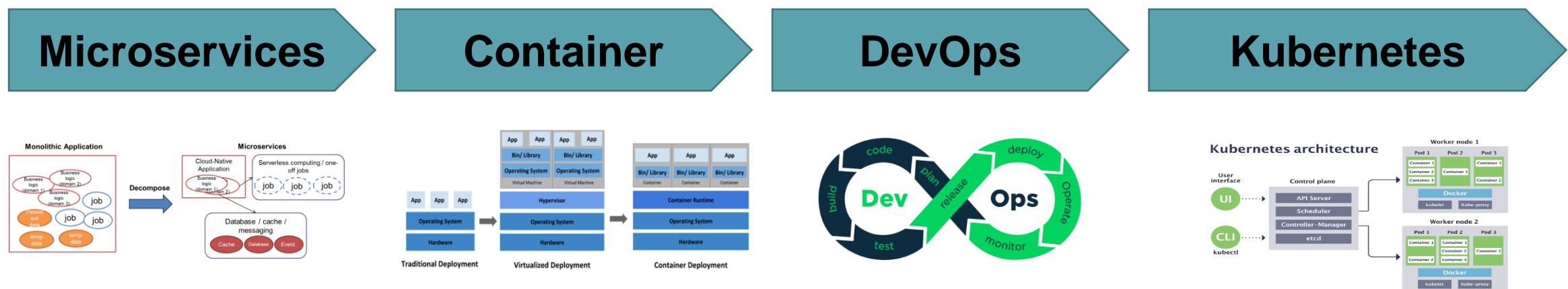
Attract and retain top talent, like You!:

- 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:

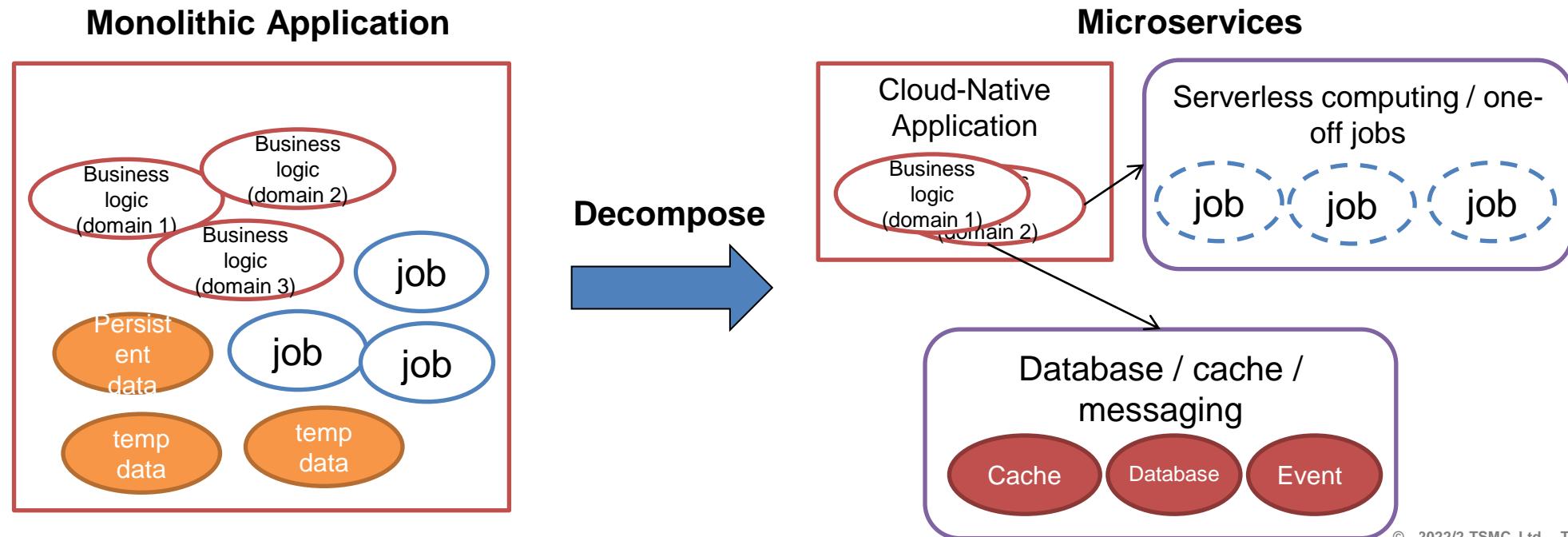
- Microservices architecture.
- Containerization.
- DevOps & GitOps.
- Cloud-native Platform (Kubernetes).



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.

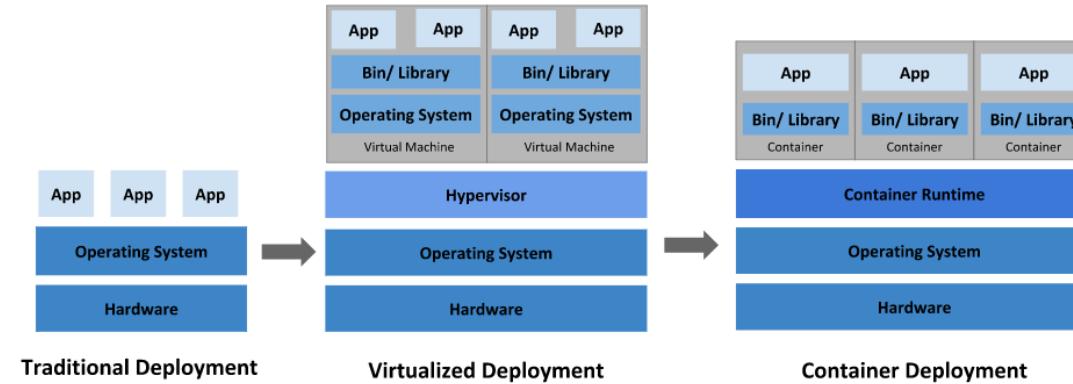


Container

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.

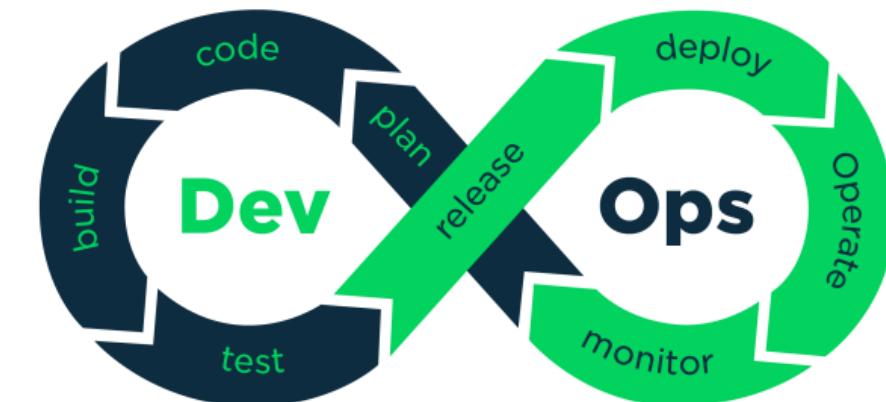


* <https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/>

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.



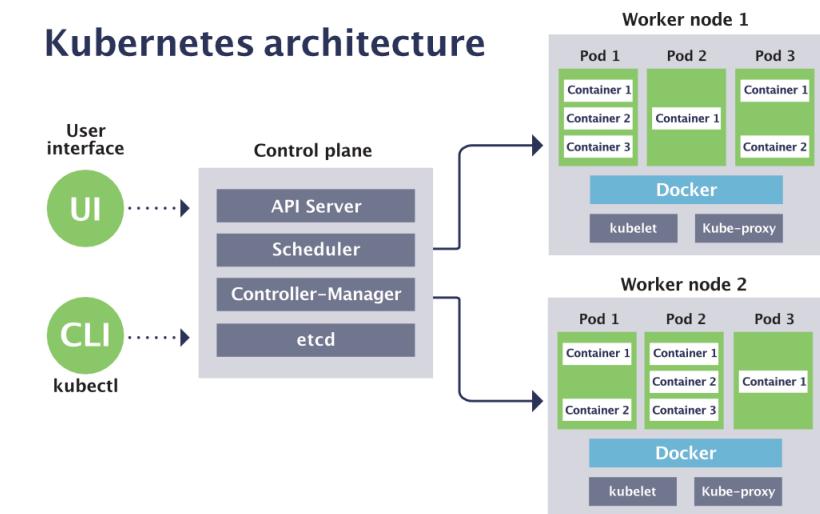
* <https://hsissokosio.wordpress.com/devops/>

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

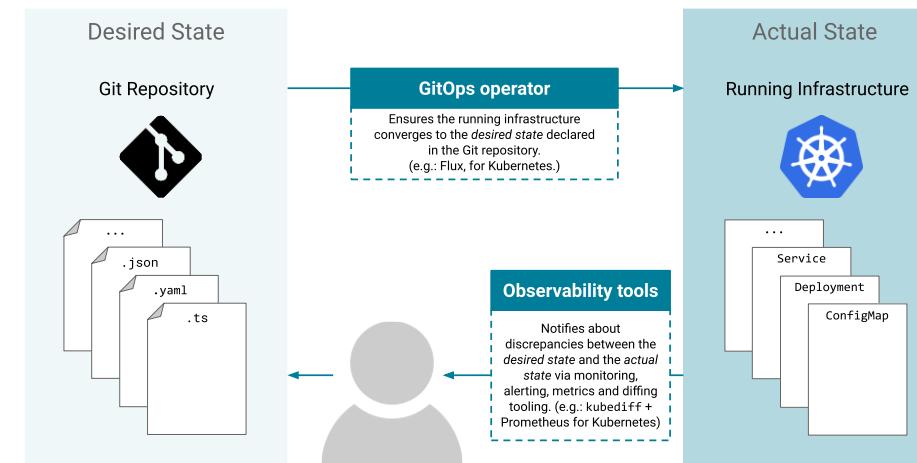


* <https://www.cncf.io/blog/2019/08/19/how-kubernetes-works/>

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.

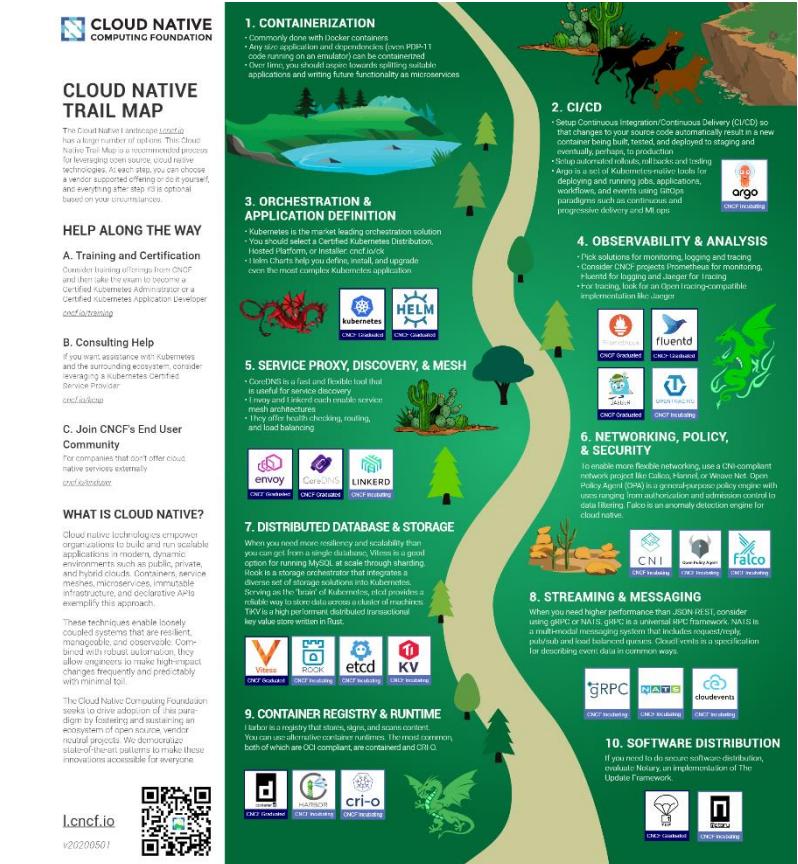


* <https://github.com/weaveworks/awesome-gitops>

Cloud Native Trail Map

● Recommended process for leveraging open source software:

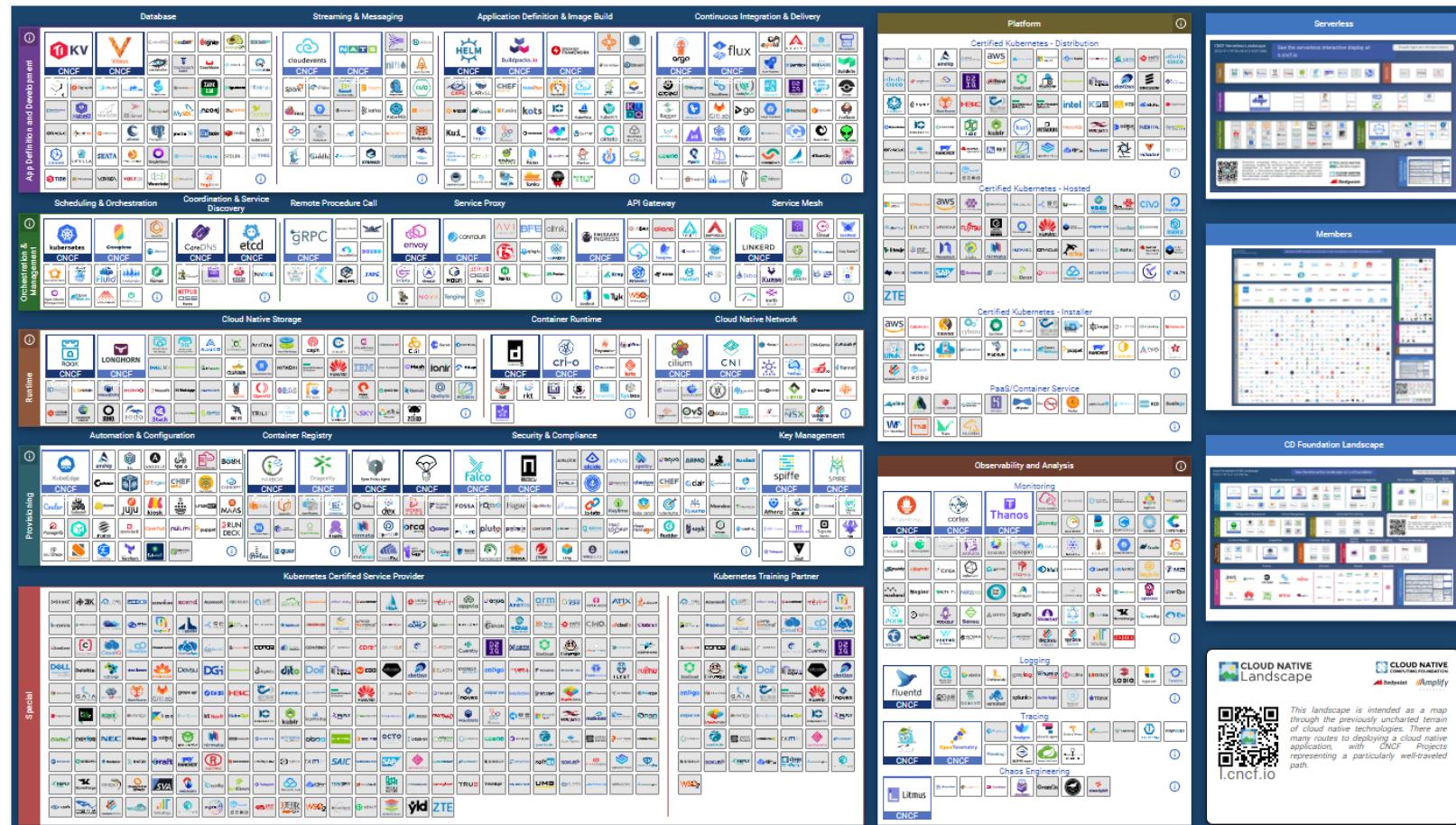
1. Containerization
2. CI/CD
3. Orchestration & application definition
4. Observability & analysis
5. Service proxy, discovery & mesh
6. Networking, policy & security
7. Distributed database & storage
8. Streaming & messaging
9. Container registry & runtime
10. Software distribution



<https://github.com/cncf/trailmap>



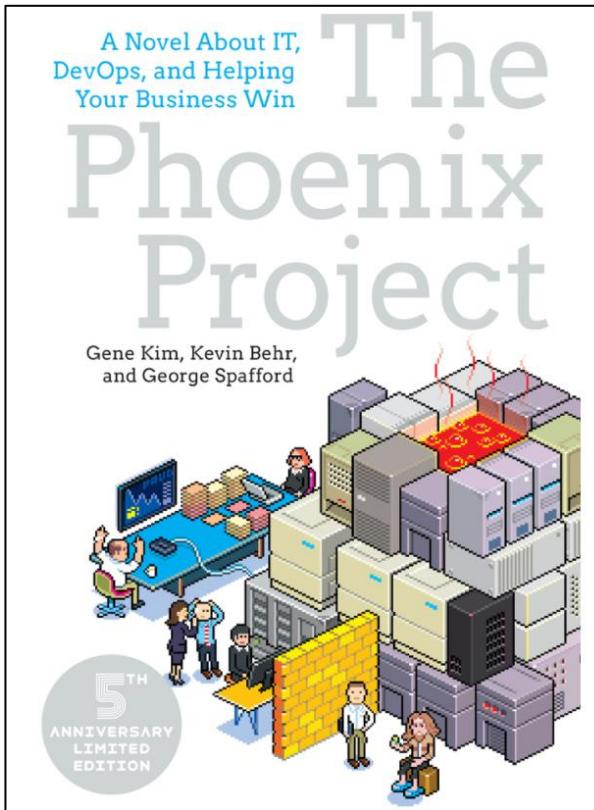
CNCF Cloud Native Landscape



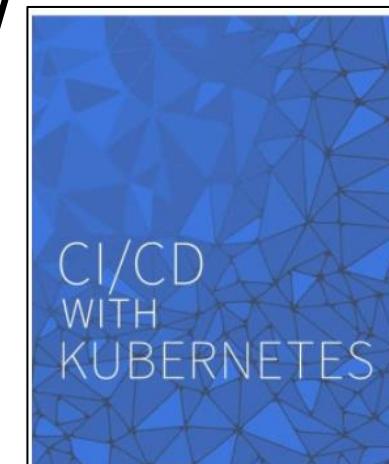
<https://landscape.cncf.io/>

References

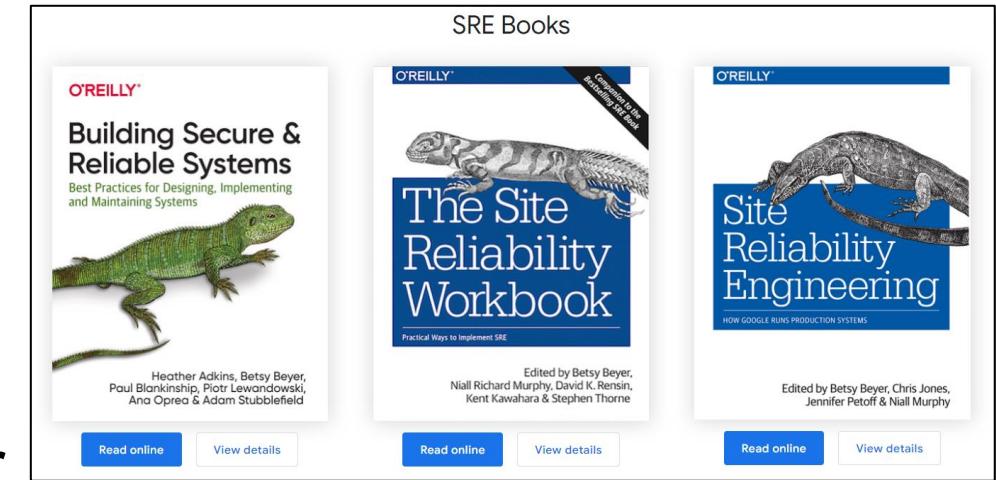
The Phoenix Project (鳳凰專案)



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



<https://sre.google/books/>



<https://kubereadme.com/>





TSMC IT X NCTU CS

課號 5270

CLOUD NATIVE Development Best Practice

問題與交流

