# 領航巡檢原則與實務

資通系統高可用、可維護領航巡檢

# 教材大綱

- 目的 (Why)
- 企圖 (What)
- 方法 (How)
- Level 2 輔導項目預覽

# 目的 (Why)

# 目的 (Why)

一起踏上旅程，探索各種可能的發展

**不要**當糾察隊

**不要**當消防隊

# 企圖 (What)

# 企圖 (What)

- 系統發展藍圖
- 缺陷的面對、記錄與解決
- 事故的應對與復原
- 自動化、自動化、自動化

# 系統發展藍圖

「外包不是讓廠商通包」
「便宜行事往往是災難的種子」

承辦要比廠商清楚狀況

包山包海就是等著出包

最小可行成果

可維護才能累積高可用

該求救就求救

# 缺陷的面對、記錄與解決

口頭交辦一定會遺忘

沒有持續追蹤就沒有奇蹟

版本記錄帶來的優勢

記錄可以輔助教育訓練

「別再相信有乖乖就足夠好運」

# 事故的應對與復原

養兵千日用在一時

演習要盡量真槍實彈

備份 3-2-1 原則

檢討報告要回饋到日常程序

「尖叫之後還是需要面對」

# 自動化、
# 自動化、
# 自動化

「人力別用來做傻事」

機器很笨，不用它的人更笨

適時告訴長官要跟上時代

細到每筆資料、每個欄位

自動產出也要自動稽核

# 方法 (How)

# 1. 工作事項及系統缺陷管理

公文往返與口頭交辦是公務部門普遍存在的習慣，但這些習慣不利於資訊系統的長期發展。

運用具有問題追蹤 (issue tracking) 功能的資訊系統去做工作與缺陷的管控，會更容易些；資訊領域環繞著扁平化的慣例發展至今，許多里程碑都奠基在這個基礎上，採用工具的同時也要有些文化上的改變發生。

# 外包沒辦法避免出包

檢視 / 輔導項目：

1. 簡述系統未來一年內之發展規劃、說明系統更新時間

[自評] 具備清晰的系統發展藍圖

[參考] 專案生命週期、專案里程碑規劃

關鍵技能：

- 專案管理
- 資訊系統架構
- 核心技術概念

# 面對問題才能解決問題

檢視 / 輔導項目：

1. 建立系統缺陷的嚴重程度分級 (severity) 及處理原則

外包廠商是否強勢主導技術選擇？

是否能夠建立鼓勵發掘問題的文化？

內部人員是否能夠區別問題嚴重程度？

處理原則 (SOP) 是否流於形式？

# 2. 災難復原站點策略

網路連線有其脆弱的一面，使用中的資訊機房很可能在事故發生當下完全無法連結。

事先備妥災難復原站點 (DR site, disaster recovery site) 並且確實演練，才能夠在災害發生當下確保資訊與服務的持續供應，避免資訊系統停擺造成的恐慌與連鎖效應。

# 異地或甚至跨國都該進入討論

檢視 / 輔導項目：

根據機構需求及風險評估選擇適當的災難復原站點 (DR site, disaster recovery site) 類型：local/remote 及 cold/warm/hot。

[自評] 災難復原站點策略

[參考] 災難復原策略文件、近期演練紀錄、災難復原測試方法。

建立災難復原站點啟動、維護及測試流程

確保災難復原站點與生產環境同步並保持更新

# 3. 人員的輪值、備援、當責

於任何時刻皆有至少一位以上之人員值勤或備勤，及時處理事故。

# 合理的班表才有合理的可靠度

[自評] 人員的輪值

[參考] 協力廠商系統維運合約、輪值表、近期事故處理紀錄

檢視 / 輔導項目：

人員的輪值、備援、當責

# 4. 事故處理、演練

事故的發生往往在預期以外，而平時的準備與演練會在當下見真章。

在事故過後，能否確實將「檢討」回饋到既有處理與演練非常重要！

# 沒有演練過的程序等於不存在

檢視 / 輔導項目：

事故處理、演練

[自評] 具備事故緊急處理標準程序

[參考] IR (Incident Response) SOP
規劃文件、通訊清單
協力廠商系統維運合約
近三年之中等以上事故記錄、處理方式（包含但不限系統失效、使用者無法存取、系統不可用、資料遺失或污染、未授權存取或修改等）

# 5. 監控及警示

自動化工具持續監控與警示，才能夠確保事前的準備能夠及時派上用場，別再安排無謂的人工檢查表。

你知、我知、獨眼龍也知道裡面打勾不代表真的發生過。

# 記得要查驗儀表板背後的真實性

[自評] 具備自動監控及警示

[參考] 協力廠商系統維運合約、系統儀表板、系統事件記錄

檢視 / 輔導項目：

監控及警示

# 6. 基礎設施可用性

基礎設施具備單點、單線路故障後持續運作能力，如電力設施、網路設施、實體主機設施、冷卻設施等。

# 99後面有幾個9？

檢視 / 輔導項目：

基礎設施可用性

[自評] 基礎設施高可用性

[參考] 基礎建設架構圖（主機、服務、網路、電力等）
基礎設施事故處理紀錄
雲端服務緊急處理連絡窗口清冊
設備之認證或佐證資料（包含但不限安全性、可靠性、可替代性、可維護性等）

# 7. 本地韌性程度

系統、元件、網路或基礎設施在面對故障、錯誤或外部干擾時，能夠在「本國範圍內」維持正常運行。

（「本國範圍內」目前意指「本國台澎金馬範圍內」。）

# 海纜斷線後 ？

[自評] 服務可於本國範圍內獨立運行

[參考] 系統架構文件、外部依賴清單、外部依賴之本地韌性 (local resilience) 承諾書

檢視 / 輔導項目：

本地韌性程度

# 8. 日誌分析工具與自動化系統

確保日誌系統具備資料保密性、完整性及可用性。

根據系統需求設定日誌收集、分析及告警策略。

# 也要注意警告疲勞問題

[自評] 日誌分析與自動化

[參考] 日誌系統、日誌分析策略、日誌告警策略

檢視 / 輔導項目：

日誌分析工具與自動化系統

# 9. 需求掌握程度

清楚呈現系統主要角色和關鍵使用路徑。

用簡單明確的方式記錄需求。

依重要性和價值排序、分類未來需求。

# 內部要有人清楚全貌

[自評] 需求掌握程度

[參考] 系統主要角色分類、關鍵使用路徑圖、需求說明書

檢視 / 輔導項目：

需求掌握程度

# 10. 外部依賴掌握程度

依賴管理：妥善管理外部依賴，對外部依賴進行版本控制使其易於鎖定版本、更新及追蹤。

授權和合規性：了解並遵守外部依賴的授權許可。

# 細節裡的魔鬼

[自評] 外部依賴掌握程度

[參考] 外部依賴清冊（載明用途及授權）

檢視 / 輔導項目：

外部依賴掌握程度

# 11. 資料掌握程度

資料可讀性：能清楚識別每個資料欄位的目的、特性、留存時間。

資料儲存結構變更管理：檢視資料表結構的異動版本記錄。

存取紀錄：對重要資料的讀取及變更，具備存取紀錄。

# 誰改了這筆資料？

[自評] 資料掌握程度

[參考] 資料表說明文件、資料儲存結構變更管理方式、近期重要資料變更存取紀錄

檢視 / 輔導項目：

資料掌握程度

# 12. 系統掌握程度

具備易於持續更新的系統文件，包括系統架構圖、API 文件等。

具備系統變更紀錄，載明目的、上線時間。

# 系統的全貌

[自評] 系統掌握程度

[參考] 系統架構圖、API 文件、系統變更紀錄

檢視 / 輔導項目：

系統掌握程度

先備：需求掌握程度、外部依賴掌握程度

# 13. 人員掌握程度

確保總是能快速找到「最熟悉需求的當責人員」及「最熟悉系統的當責人員」。

# 要知道找誰求救

[自評] 人員掌握程度

[參考] 相關人員（需求、系統管理）角色清冊

檢視 / 輔導項目：

人員掌握程度

先備：需求掌握程度、外部依賴掌握程度

# 14. 程式碼掌握程度

擁有完整的原始碼，並確認授權以及合約對於原始碼的限制。

清楚定義程式更新方式，具備更新日誌，並以版本管理系統進行程式異動管理。

# 程式碼是另一本技術手冊

[自評] 程式碼掌握程度

[參考] 程式碼庫、軟體更新日誌

檢視 / 輔導項目：

程式碼掌握程度

先備：需求掌握程度、資料掌握程度、系統掌握程度

# 15. 資料備份機制（3-2-1 原則）

保留至少 ***3*** 份資料副本，應在 ***2*** 種不同的存儲介質（儲存媒體）上，且至少有 ***1*** 份位於其他地點。

定期檢查備份資料的完整性和可用性。

建立資料恢復流程，以確保在數據丟失時可以迅速恢復。

# 沒備份神仙難救

[自評] 資料備份機制（3-2-1 原則）

[參考] 資料備份策略文件、資料恢復流程文件、資料備份檢查紀錄

檢視 / 輔導項目：

資料備份機制（3-2-1 原則）

# Level 2

輔導項目預覽

# Level 2 檢視 / 輔導項目

- 確保災難復原站點能在災難或故障發生時迅速恢復系統運行
- 定期驗證和測試災難復原站點的恢復能力
- 確保日誌擷取、分析和儲存的完整性與一致性
- 除了日誌收集外，還須選擇適合的「即時」日誌分析工具，包含網路流量、端點存取、端點行為、Web 存取、Web 行為等
- 服務效能、服務穩定性、資安事件三方面之即時分析預測與監控

# Level 2 檢視 / 輔導項目

- 主動分析及偵測以提升系統效率、縮短事件反應時間、偵測異常及入侵行為
- 定期檢查日誌分析系統運作狀況，並調校日誌分析策略以提高系統監控效率
- 透過日誌分析監控系統狀態、識別問題並調校性能
- 可追溯性：確保需求項目與程式元件和測試元件之間是可追朔的
- 變更管理：確立在軟體生命週期中，應對「需求變更」的方式，包括如何分析、溝通、進入開發及驗證

# Level 2 檢視 / 輔導項目

- 變更管理與控制：對於已上線持續運作中之系統，若須更動，從變更提案、評估、計劃、實施到後續監控的所有過程，確保系統變更的安全性和可靠性
- 模組化：系統妥善模組化並清楚定義交互介面，以提高可讀程度
- CI/CD：確立 CI/CD 流程，並確保它們在開發週期中減輕開發人員負擔
- 複雜度：讓系統足夠簡單，簡化或刪除意義模糊、流程不清的元件
- 系統配置管理：系統配置是否文件化、程式化並透過版本管理系統管理

# Level 2 檢視 / 輔導項目

- 可驗證：確保具備與正式環境配置相近的主機或服務，以便進行有效的系統驗證
- 知識共享：確保需求管理者、系統使用者、系統管理者等角色之間知識共享，保持開放交流和合作
- 技能發展：強調持續學習和技能發展的需要，以保持最佳技術及實踐
- 求才：確保機關或協力廠商能持續求才，能及時補足適任人員
- 程式可讀性：確立程式風格指引，檢視程式是否風格一致並且易懂

# Level 2 檢視 / 輔導項目

- 程式元件目的及可重用程度：清楚知道每一個程式元件的目的，並確保可重用性佳
- 技術更新：檢視程式元件、函式庫之版本升級、軟體技術升級方式及未來規劃
- 可測試程度：檢視測試方式、測試流程及測試涵蓋率