

# SD與PG人員作業管理規範

- 本次說明內容是依據公司公佈於 EIP 上的表單/規範文件
- 文件參考時間:2022/8/31

目錄資訊

目錄安全

新增目錄

刪除目錄

重新命名

新增檔案

新增本文

訂閱

目錄連結

收藏目錄

文書管理

法務

財會

基金

專案管理

採購

軟體技術

- SD與PG人員作業管理規管
  - 表單文件
  - 規範文件
  - 需求表單

業務

經營企劃

資安

總務

職權會

知識共享

- EIP系統使用手冊

電子簽核流程

搜尋目錄下文件(名稱、編號、關鍵字):

搜尋

顯示其他狀態文件(說明)

顯示方式: 文件列表

排列方式: 文件名稱

狀態: 全部

筆數:1 到 8 共 8 筆

1

共 1 頁

文件編號	文件名稱	作者	版本	狀態	公佈日	關鍵字	到期日	操作
☆	AP_Log_作業管理規範_20170630 [19]	平台服務 鄭蓓雯	1.0	已公佈	2020/12/16 23:20			資訊 進階
☆	AP程式撰寫規範V1 1 [25]	平台服務 鄭蓓雯	1.0	已公佈	2020/12/16 23:22			資訊 進階
☆	foritfy產出說明 [13]	平台服務 鄭蓓雯	1.0	已公佈	2020/12/16 23:24			資訊 進階
☆	JAVA程式撰寫相關文件 [17]	平台服務 鄭蓓雯	1.0	已公佈	2020/12/16 23:25			資訊 進階
☆	Log 規範 [27]	平台服務 鄭蓓雯	1.0	已公佈	2020/12/16 23:26			資訊 進階
☆	UI開發規範文件 [18]	平台服務 鄭蓓雯	1.0	已公佈	2020/12/16 23:27			資訊 進階
☆	webinspect產出說明 [17]	平台服務 鄭蓓雯	1.0	已公佈	2020/12/16 23:28			資訊 進階
☆	新增dotNet相關文件 [12]	平台服務 鄭蓓雯	2.0	已公佈	2020/12/16 23:23			資訊 進階

筆數:1 到 8 共 8 筆

1

共 1 頁

1. 前言

2. AP 作業環境管理規範

3. AP 程式撰寫規範

4. AP 版本控制規範

5. AP 測試規範

6. AP 建構佈署作業管理規範

《開發前》

1. 前言

2. AP 作業環境管理規範

《開發進行中》

3. AP 程式撰寫規範

4. AP 版本控制規範

《測試活動》

5. AP 測試規範

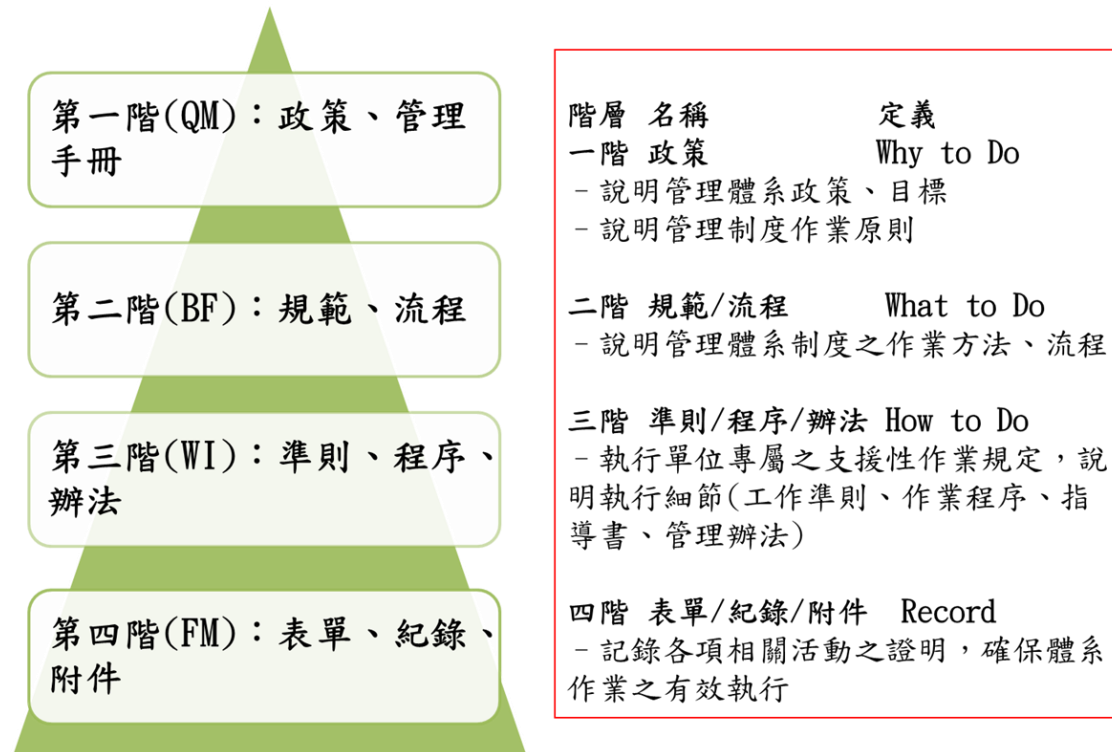
《上版作業》

6. AP 建構佈署作業管理規範

## 目的

- 明確訂定程式設計作業流程與管理規範，以利SD 與PG人員共同遵循，培養良好紀律、提升產能。

( 四階文件：政策、**規範**、程序、表單/記錄/附件 )



## 效益

- 減少時間成本。
- 提升技術水平。
- 降低專案風險。
- 確保程式品質。

## 對象

- 系統設計人員（SD）。
- 程式設計人員（PG）。
- 適用對象：公司全體同仁與委外廠商。

《開發前》

1. 前言

2. AP 作業環境管理規範

《開發進行中》

3. AP 程式撰寫規範

4. AP 版本控制規範

《測試活動》

5. AP 測試規範

《上版作業》

6. AP 建構佈署作業管理規範



### 目的

- 統一專案團隊針對各AP作業環境之稱呼、定義與用途。
- 協助專案團隊迅速掌握瞭解AP系統整體架構、AP系統相互影響之範圍與AP系統的資料流向。
- 確保環境一致，減少AP 因環境差異所造成之執行結果不一致或偵錯困難。

## 2. AP 作業環境管理規範

**定義**      類型：包含（但不限於）下列**五**種環境：

#	環境類型	用途	實體位置
1	個人開發環境	專案相關成員個人使用的程式開發環境	2022年 依據公司“資料不落地”政策，開發人員環境應於 VDI 內操作
2	專案開發環境	專案相關成員共同使用的程式開發環境	通常位於內湖陽光機房、是方機房、或 公有雲服務
3	測試環境	專案相關成員個人或共同使用的程式測試環境。 1.單元測試 2.功能測試 3.整合測試 4.使用者驗收測試	通常位於內湖陽光機房、是方機房、或 公有雲服務
4	正式環境	日常業務正式運作之環境	通常位於板橋機房、或 公有雲服務
5	異地備援環境	或稱「災害復原環境」，於重大災難或意外發生時，用以替代正式環境之系統環境	通常位於桃園富國機房，或外部雲端機房。

## 2. AP 作業環境管理規範

### 政策規範

#### 一、作業環境管理

環境類型	管理
個人開發環境	<b>2022年</b> 依據公司”資料不落地”政策，開發人員環境應於 VDI 內操作。專案所需之相關軟體安裝設定，應由各AP專案單位自行負責。
專案開發環境	
測試環境	
正式環境	
異地備援環境	由AP專案單位向基盤服務部申請協助建置與網路連線，軟體(除OS及DB以外)安裝設定與日常維護工作則由AP專案單位自行負責。

## 政策規範 二、作業環境一致性

### 1. 環境間的一致性

AP專案單位應盡可能維持各環境之間的一致性，並製作「軟體構成表」，以減少因環境變數所造成的AP執行結果不一致或偵錯困難，若有差異之處亦應特別加強說明。

**1)版本**：軟體版本盡可能一致，如：Apache、JBoss、JDK、.NET Framework.....等等。

**2)路徑**：軟體安裝的路徑或檔案路徑必須盡可能地一致。

**3)使用者權限**：process (daemon或batch job) 執行身份、檔案或目錄的權限。

**4)通訊協定**：通訊協定一致，如：HTTP、HTTPS、FTP.....等。

**5)硬體參數設定**：如：OS、JBossServer、DB...等參數設定。

## 政策規範 二、作業環境一致性

### 2. 個人開發環境之一致性

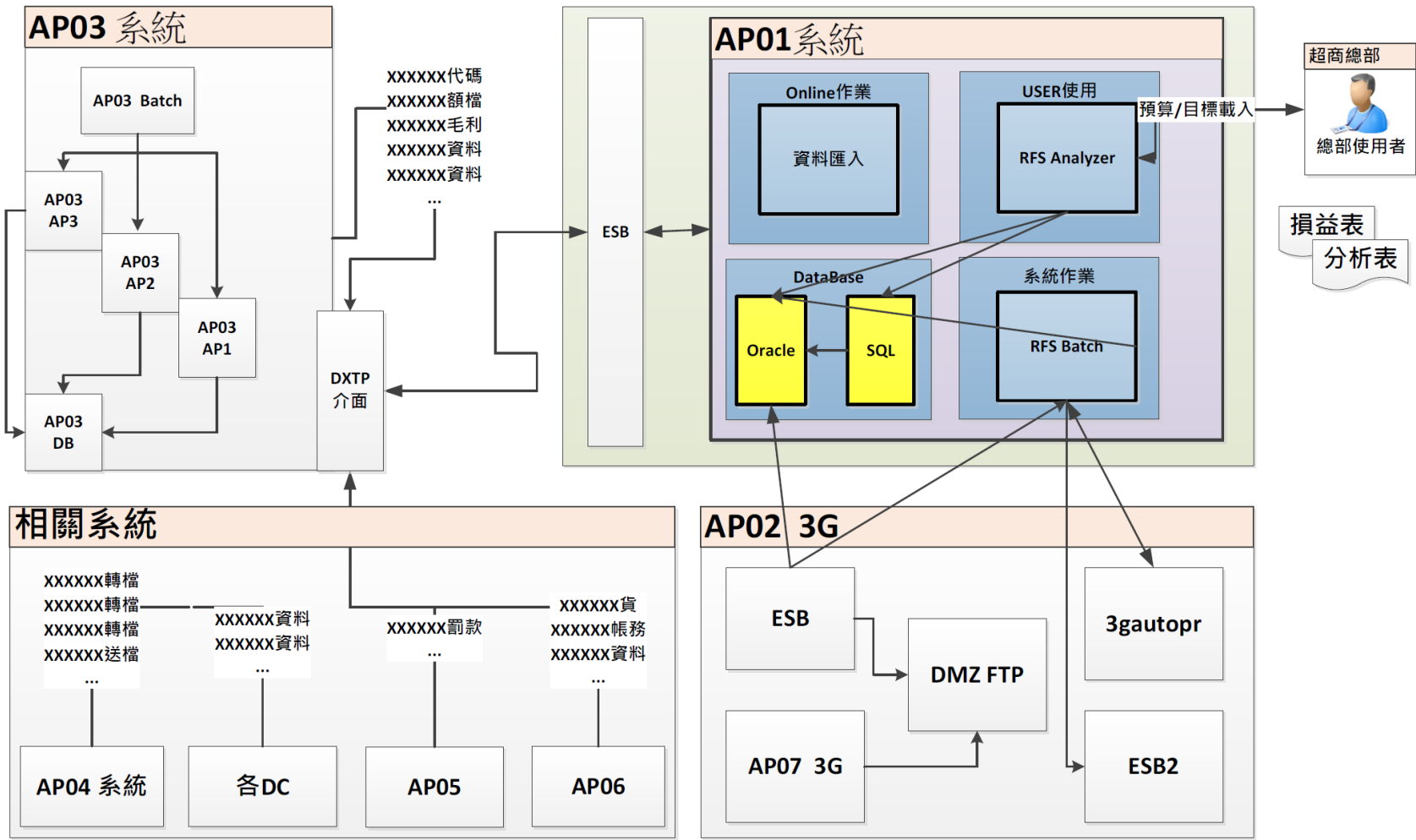
AP專案單位應盡可能維持「個人開發環境」之間的一致性，以縮短開發人員使用不同個人開發環境時的適應時間，並減少日後因環境不同所衍生的相容性問題或bug（如：路徑、JDK、瀏覽器版本不同，導致執行結果不一致），相關資訊須紀錄至「軟體構成表」內。

	A	B	C	D	E	F
1	範例(可自行增刪項目，如：用途等說明)					
2						
3	PIC Framework 2G					
4	軟體	目前版本	路徑/服務	權限	通訊協定	通訊埠
5	項目名稱					
6	軟體名稱	安裝版本	安裝路徑	安裝角色	FTP, HTTP等。null	port號。null代表無
7						
8	如以下範例：					
9						
10	作業系統					
11	Windows	Windows Server 2008 R2 Enterprise	系統槽 C:\	administrator	null	null
12	AP Server					
13	IIS	iis7	C:\inetpub\wwwroot	administrator	HTTP	80
14	Framework					
15	.Net Framework	4.5	服務	administrator	null	null
16						
17						
18						
19						
20						

### 政策規範 三、系統全體關聯圖

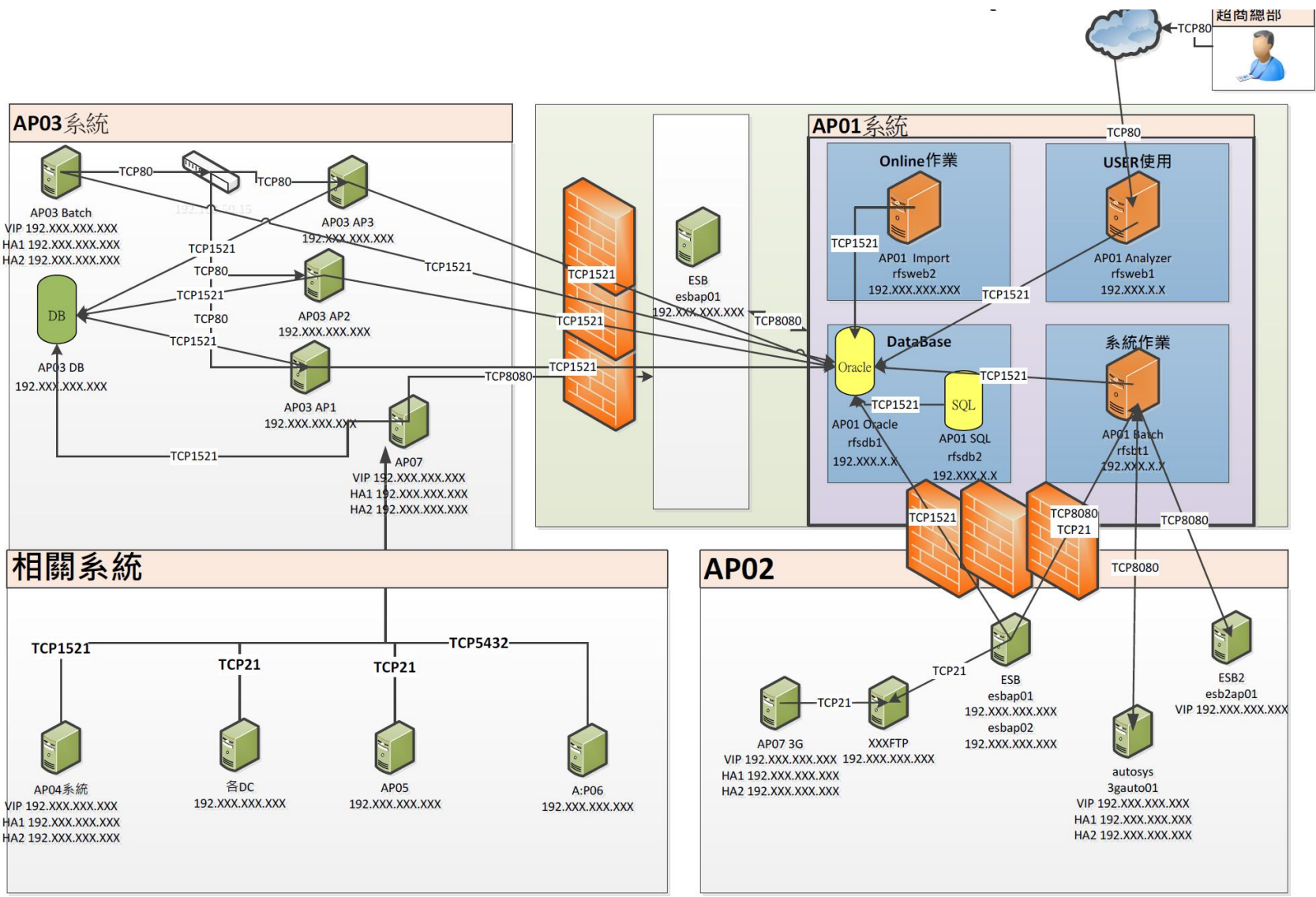
AP專案單位須維護「正式環境」之系統全體關聯圖（或稱架構圖），系統全體關聯圖包含「**系統關聯圖**」、「**網路架構圖**」以及「**軟體架構圖**」。

## 政策規範 三、系統全體關聯圖-系統關聯圖(範例)



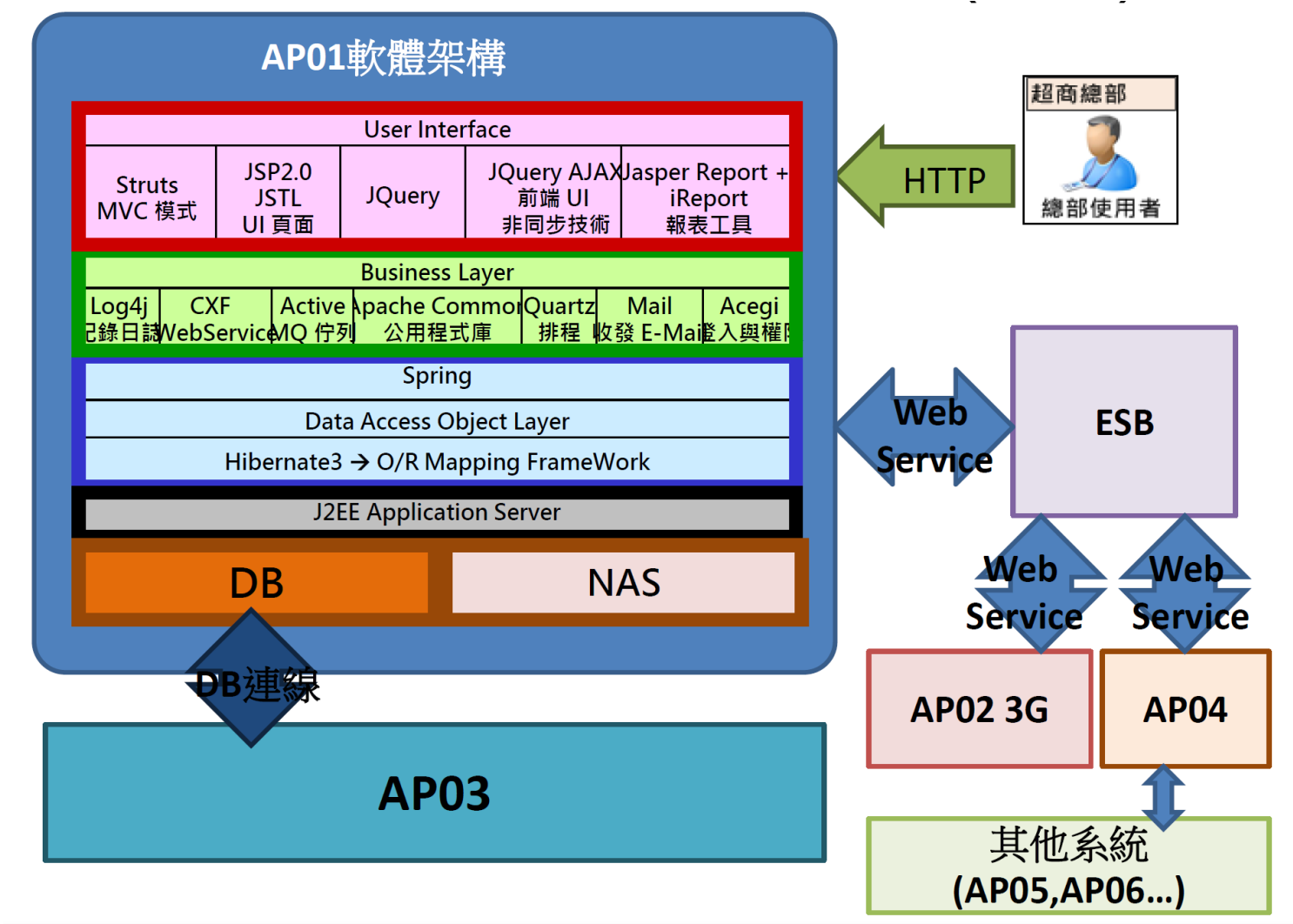
## 政策規範

### 三、系統全體關聯圖-網路架構圖(範例)





## 政策規範 三、系統全體關聯圖-軟體架構圖(範例)



### 政策規範

#### 四、專案團隊之職責

- 專案團隊應確實遵循本規範之規定，配合審查單位之審查。若對本規範有任何異議，亦得提出具體建議。
- 專案團隊採用任何未經公司認可之AP環境之前，均須提報技術支援單位評估。
- 專案團隊需維護一份「系統全體關聯圖」以及「軟體構成表」，並增編修訂紀錄。

《開發前》

1. 前言

2. AP 作業環境管理規範

《開發進行中》

3. AP 程式撰寫規範

4. AP 版本控制規範

《測試活動》

5. AP 測試規範

《上版作業》

6. AP 建構佈署作業管理規範

## 目的

- 統一專案團隊之程式撰寫的標準。
- 減少專案團隊因個人作業模式差異所造成之後續程式維護困難。
- 提升專案團隊之知識技能、工作效率與程式品質。
- 管理公司共同資產，降低AP專案開發與維護成本。

## 定義

### ➤ 程式碼撰寫規則(Coding Rule)：

內容須包含但不限於命名規則(Naming Rule)、程式撰寫風格(Coding Style)、Log處理方式、Exception 處理方式等。

### ➤ AP框架(AP Framework)：

指提供已設計好的API、SDK、Interface...等，讓程式設計者可以在更短時間內進行開發、測試，如UI、MVC、Batch、Testing、SOA、AOP等框架。

### ➤ 安全程式碼：

依據各程式開發語言之特性，撰寫程式時所需注意的資安議題，確保專案程式碼在具有最大安全性之後方可使用。

## 定義

➤ **AP資安弱掃：**

指使用公司建置之AP資安弱掃平台所進行的源碼檢測與AP弱點掃描之動作與流程。

➤ **AP資安弱掃平台：**

指公司建置的源碼檢測工具與AP弱點掃描工具之統稱。

➤ **源碼檢測：**

指專案維運/開發過程中，使用源碼檢測工具(例如：Fortify、Checkmarx)進行程式碼檢視與程式碼資安檢測。

## 定義

### ➤ AP弱點掃描：

指專案維運/開發過程中，使用AP弱點掃描工具(例如：WebInspect)進行資訊安全弱點檢測之程序流程與動作。

### ➤ AP效能調校：

依據各程式開發語言之特性，撰寫程式時所需注意的效能優化，或調整程式執行環境之參數以期符合系統規格之要求。

### ➤ 第三方軟體(Third Party Software)：

指本公司不具備智慧財產權之軟體產品，如Library、API、SDK、Framework等。

## 政策規範

### 一、程式撰寫規範

#### 1.作業原則

- 依程式語言之特性，由專案團隊定義相對應之程式碼撰寫規則。
- 依程式語言之特性，由專案團隊定義AP框架、AOP與物件導向設計模式的相關事項。
- 上述兩項，專案團隊應優先參考技術支援單位所制定之通用規範；既存現有AP專案系統，可遵循既有的程式碼撰寫規則。
- 於開發/維運期間，需定期進行源碼檢測與AP弱點掃描。
- 使用第三方軟體，需先進行源碼檢測並持續進行資安相關的更新與補強。



## 政策規範

### 一、程式撰寫規範

#### 2.作業程序

- 程式撰寫須符合程式碼撰寫規則。
- 於程式撰寫完成後，應由其他成員(不可為原程式碼撰寫者)進程式碼檢視 ( Code Review )，以確認內容是否符合程式碼撰寫規則之要求。
- 程式碼檢視 ( Code Review ) 作業後，得將改善項目內容進行紀錄，以便後續改善進度、狀況追蹤以及原因檢討，檢視者可依檢視項目清單 ( Checklist ) 進行檢視，檢視結果須明確說明須改善項目內容，並進行修正，直至符合程式碼撰寫規則之要求。
- 開發期間需維護一份程式清單總表，並記錄程式碼檢視 ( Code Review ) 實施情況。
- 開發程式碼需透過公司制定的AP資安弱掃平台，進行AP資安弱掃。

## 政策規範

### 一、程式撰寫規範

#### 2.作業程序

- 各專案團隊需維護一份AP資安弱掃平台使用總表，並記錄每次源碼檢測與AP弱點掃描後 **Critical Level** 與 **High Level** 之原因，以及程式碼調整之範圍與作法。
- AP專案進入維護階段後，仍須一年至少一次進行AP資安弱掃。
- 程式碼檢視 ( Code Review ) 與AP資安弱掃結果須進行統計、分析與檢討，以利往後程式撰寫的改善。
- AP專案於開發期間，需使用公司提供之壓力測試平台進行壓力測試，以確保系統的效能穩定，並符合專案SLA需求。

## 政策規範

### 二、專案團隊之職責

- 專案團隊應確實遵循本規範之規定，配合審查單位之審查。若對本規範有任何異議，亦得提出具體建議。
- 專案團隊採用任何未經公司認可之AP框架之前，均須提報技術支援單位評估、審核。
- 專案團隊採用任何未經公司認可之第三方軟體之前，均須提報技術支援單位進行評估、審核，審核後須確認授權許可與智慧財產權等問題；如有必要，得請法務人員協助確認。為避免延宕專案時程，宜及早提出申請評估。

## 政策規範

### 二、專案團隊之職責

- 每次程式碼檢視(Code Review)後皆須產生檢視記錄，專案團隊須妥善保留此份文件，以供審查單位進行審查。
- 專案團隊需定期進行AP資安弱掃。
- 各專案團隊皆盡可能使用公司制定之系統資源監控工具，進行系統資源效能監控。
- 於AP專案系統開發與維運時期發生如效能瓶頸之問題，欲對AP伺服器進行相關參數調整時，需依技術支援單位所制訂之AP效能調校指導書進行調校。若進行之參數調校未載明於指導書之中，得依據開發之程式語言或採用之AP伺服器官方文件進行參數調整，並紀錄調整之依據來源提供技術支援單位進行指導書修訂。

《開發前》

1. 前言

2. AP 作業環境管理規範

《開發進行中》

3. AP 程式撰寫規範

4. AP 版本控制規範

《測試活動》

5. AP 測試規範

《上版作業》

6. AP 建構佈署作業管理規範

## 目的

- 規定專案團隊通用之版本控管標準及確保流程一致性。
- 減少團隊在進行專案開發時，因版本控管流程操作之差異，或系統當機而導致版本異常的情況，如：程式碼相關資料的相互覆蓋或衝突等情況。
- 確保發生版本異常時，可回溯至穩定的版本。並藉以提供有效的追蹤紀錄，以確保系統文件的完整性及安全性。

## 定義

### ➤ 系統文件：

與程式碼有關之專案團隊工作產出物，包含但可不限於程式原始碼、設定檔、組態檔、DB schema、shell scripts、圖檔、HTML、Javascript、CSS等。

### ➤ 版本庫(Repository)：

版本庫中儲存了系統文件與改版歷程，是版本控制系統的基本管理單位。每次版本庫的異動提交 ( commit ) 皆會對應至一組改版編號，其中記錄了該次異動的檔案清單、內容、日期時間、提交者、說明訊息等。

## 定義

### ➤ 版本控制系統 ( **Version Control System** ) :

用來儲存與管理多個版本庫的系統。在軟體開發或維護過程中，專案團隊可使用此類系統進行版本控管，並對不同版本庫設定不同的帳號或群組存取權限。此類系統如Subversion ( SVN )、Git、Team Foundation Server等。以下簡稱版控系統。



## 政策規範

### 一、系統文件版本控制

建置全公司統一的版本控制平台，區分開發測試的版本與正式環境的版本，並放置不同之版本庫進行控管，且進行異地備份。

#### ➤ 開發測試的版本：

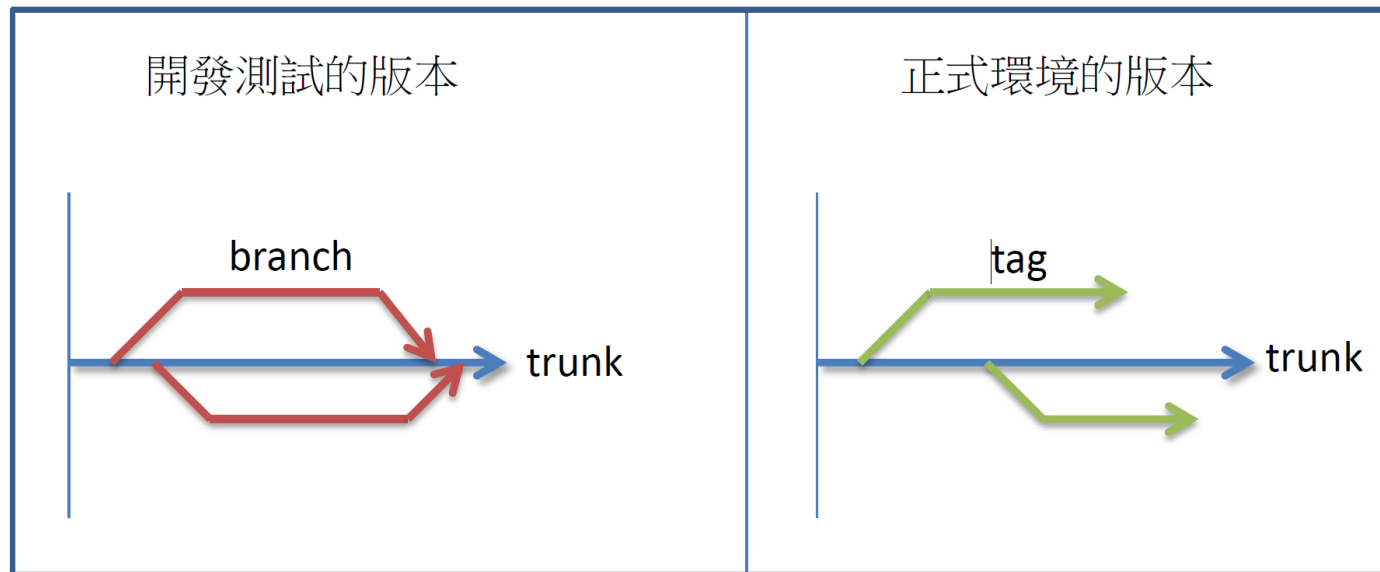
- 需確保系統文件的一致性，並記錄更新歷程。
- 盡量採行trunk/branch方式進行系統文件的版本控制。

#### ➤ 正式環境的版本：

- 需確保Release系統文件的一致性，並記錄更新歷程。
- 盡可能在上版完成後，進行Tag的分割，以利正式環境版本的追溯與還原。

## 政策規範

### 一、系統文件版本控制



## 政策規範

### 二、系統文件管理

- 保障系統文件完整性
- 系統文件的權限控管
- 保障系統文件安全性

## 政策規範

### 三、系統文件集中管理

- 專案團隊應主動將所負責的全部系統文件，提交至公司統一的版控系統平台之中。
- 全公司統一的版控系統平台應設置於Intranet網段，禁止未經授權的人員的存取，並來自Internet的存取需透過VPN機制進行，存取記錄（access log）依公司資安政策規定保存五年。
- 全公司統一的版控系統平台管理員須視人員職務調動（含離職）、業務移轉或組織調整，即時調整帳號權限或系統設定。

《開發前》

1. 前言

2. AP 作業環境管理規範

《開發進行中》

3. AP 程式撰寫規範

4. AP 版本控制規範

《測試活動》

5. AP 測試規範

《上版作業》

6. AP 建構佈署作業管理規範

## 目的

- 規定專案團隊通用之系統測試標準及流程。
- 減少專案團隊因個人測試差異所造成之程式障礙。
- 提升專案團隊之工作效率、程式品質。
- 降低因程式錯誤所產生之專案開發及維護成本。

## 定義

### ➤ 單元測試 ( Unit Test ) :

在開發期間針對軟體的最小單元進行程式正確性檢驗的測試工作，以確保程式品質。在程序中，一個單元意即單個程式、函式等；例如：物件導向設計模式中，最小單元就是方法或函式。

### ➤ 功能測試 ( Function Test ) :

針對單一USE CASE的作業功能，撰寫測試案例進行測試，以符合系統功能規格。

### ➤ 連結測試 ( Interface Test ) :

確認整合的過程中各部分功能是否達到或實現設計需求，由SD進行測試，SA協助共同進行測試，並對應障礙與問題。(內部與外部連結測試)

## 定義

### ➤ 整合測試 ( **System Integration Test** ) :

由SA進行測試，SD協助共同進行測試，並對應障礙與問題。(功能面與非功能面測試)

### ➤ 用戶驗收測試(**User Acceptance Test**) :

此活動是以使用者的觀點來確認系統妥當性的測試；SA偕同User共同進行測試，SD協助對應障礙與問題。

### ➤ 效能測試：

確立系統可正常運作的一種測試方法，通常在系統正常運作範圍之內進行，確保系統執行效能，符合SLA的效能需求。

### ➤ 壓力測試：

確立系統穩定性的一種測試方法，通常在系統正常運作範圍之外進行，以考察其功能極限和隱患。



## 政策規範

### 一、單元測試

#### 作業原則

- 1) SD應依據系統設計規格文件(如：SD001)，提供單元測試原則，PG據以進行單元原則，進行單元測試個案的開立，並進行測試工作。
- 2) PG在完成或修改一或數個單元測試碼撰寫後，即必須自行完成單元測試的工作，以檢驗是否達成詳細設計文件所訂立的規格，並將執行結果記錄於單元測試規格書(PG003)。
- 3) 單元測試程式盡可能遵循FIRST法則撰寫，可快速且重複被執行，且相互之間不互相依賴，並顯示執行結果是否成功，以避免人工比對執行結果造成誤差。
- 4) 盡可能以自動化工具協助進行測試，以加速完成測試活動進行，可透過技術支援單位核可之第三方工具軟體或框架如：Junit、Nunit、Selenium...等工具撰寫及執行單元測試。

## 政策規範

### 一、單元測試

#### 作業原則

- 5) PG應將單元測試之結果、報表或螢幕拷貝等測試佐證紀錄，產出單元測試規格書提交SD檢閱，形式如下：Html、PDF、Word、Excel...等及其他可明確呈現之樣式。
- 6) SD應檢視單元測試規格書，並檢視單元測試個案，確認測試完整性。

## 政策規範

### 二、功能測試

#### 作業原則

- 1) SD依據功能規格文件(如：RA006)開立功能測試規格書(SD008)，於程式開發完畢後，PG應盡可能自行進行功能測試，SD則進行最後的測試與紀錄。
- 2) SD確認功能測試規格書內之測試個案須全數通過測試後，將測試通過之程式碼，須依建構管理流程納管後進行上版，通知SA進行驗收測試。
- 3) 功能測試須遵循專案團隊制定之品質基準，如：retake次數低於3次...等標準。
- 4) 若SD及PG為同一人，須進行交叉測試。

## 政策規範

### 三、連結測試、整合測試

#### 作業原則

- 整體測試計畫由PM進行規劃制定。
  - 1) SD負責執行連結測試相關活動，並記錄測試結果。
  - 2) SA負責執行整合測試相關活動，並記錄測試結果，若發生系統障礙，則由SD協助處理與對應。

## 政策規範

### 四、用戶驗收測試

#### 作業原則

- 1) 使用者負責制定與執行用戶驗收測試相關活動，並紀錄測試結果，確認驗收。
- 2) PM與SA協助使用者制定與執行用戶驗收測試。

《開發前》

1. 前言

2. AP 作業環境管理規範

《開發進行中》

3. AP 程式撰寫規範

4. AP 版本控制規範

《測試活動》

5. AP 測試規範

《上版作業》

6. AP 建構佈署作業管理規範

## 目的

- 規定專案團隊通用之程式建構佈署作業管理標準。
- 減少專案團隊因個人的習慣差異，導致程式建構佈署未落實。
- 建置AP專案程式建構佈署標準，提升程式建構佈署之工作效率。
- 簡化並強化AP建構佈署流程，降低程式建構佈署錯誤率。

## 定義

### ➤ 靜態程式分析(Static Program Analysis)

指在不執行電腦程式的條件下，進程式分析的方法。本規範指的靜態程式分析是指配合靜態程式分析工具進行的分析(源碼檢測)以及人工進行的分析 (程式碼檢視Code Review)。

### ➤ 建構 ( Build )

使軟體成為可交付、可發行或可佈署形式的過程，包含但不限於原始碼編譯、靜態程式分析、執行單元測試、打包、產生報告或文件等步驟。最終產出物包含但不限於編譯或打包後之二進位檔案、設定檔、單元測試報告、測試覆蓋率報告、異動程式一覽表等。亦稱為「建置」。



### 定義

#### ➤ 交付 ( **Delivery** )

將建構產出物交給客戶，由客戶端自行安裝於目標執行環境 ( Target Runtime ) 。

#### ➤ 發行 ( **Release** )

將建構產出物放置於公開平台，由客戶自行下載並安裝於目標執行環境。亦稱為「釋出」。

#### ➤ 佈署 ( **Deployment** )

將建構產出物安裝至目標執行環境上，並使其能正常運作的過程，包含但不限於軟體安裝、組態設定、服務重啟等步驟。亦稱為「上版」。

#### ➤ 計畫性佈署 ( **Planned Deployment** )

指在工作計畫之中已預先安排的佈署作業。

## 定義

### ➤ 非計畫性佈署 ( **Unplanned Deployment** )

指在工作計畫之中未預先安排，但因突發事件必須儘速實施的佈署作業。

### ➤ 持續整合工具 ( **Continuous Integration Tool** )

指在軟體開發過程中，用以協助開發團隊將軟體產品維持於隨時可交付、發行與佈署狀態的自動化建構工具。設定排程與腳本後，此類工具將自動從版本控制系統簽出程式碼，執行編譯、單元測試、產生測試報告、佈署等作業，建構佈署完成或過程中發生錯誤，均可設定自動通知團隊成員。此類工具如Jenkins、Team Foundation Server (TFS)等。

## 政策規範

### 一、建構作業

#### 作業原則

- 1) 建構作業應以自動化為原則，專案團隊可於整合測試環境上使用持續整合工具，以進行每日例行性的自動建構佈署作業。
- 2) 用以建構軟體產品之來源檔案皆應納入版本控制系統中進行版本控管，以利問題追溯或版本回復。
- 3) 基於資安考量，執行建構作業之建構人員，應由專人負責，且不可為原程式開發人員。

## 政策規範

### 一、建構作業

#### 作業程序

- 以下作業程序可以由開發人員/建構人員手動執行，或使用持續整合工具自動執行：
  - 1) 自版本控制系統中取出原始碼、設定檔或其他相關檔案。
  - 2) 建構測試環境透過版本控制中的測試版本庫取出程式碼，建構正式環境則由正式版本庫取出程式碼。
  - 3) 對上述檔案進行程式編譯、測試、靜態程式分析、打包、產生文件或報告等程序。
  - 4) 在交付、發行或佈署前，開發人員/建構人員必須將完成建構的相關產出物提交至Release確認會議審核。

## 政策規範

### 二、佈署作業

#### 作業原則

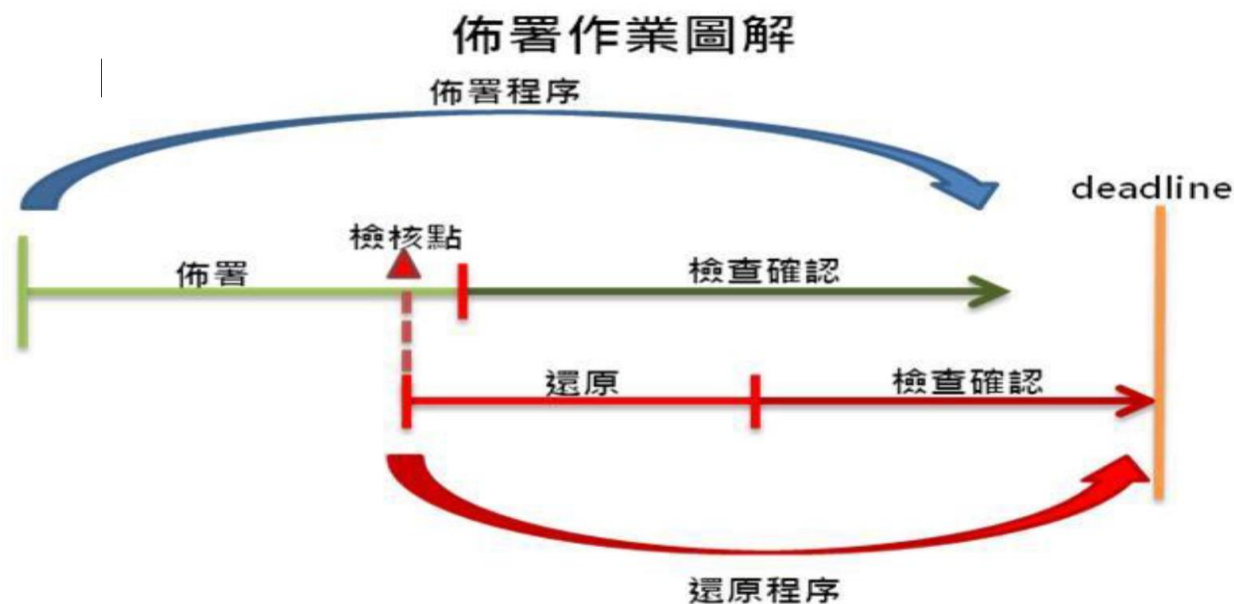
- 1) 於正式環境上進行之佈署作業時，應盡可能確保系統之高可用度、減少因為服務中斷或暫停所造成之客戶損失、最小化業務影響範圍與時間、符合SLA要求，並經客戶與Release確認會議核可之後，方可實施。
- 2) 為縮短佈署時間、減少人工失誤，應盡量利用相關軟體工具輔助佈署作業，建立常態性的佈署作業手順，並具備可重複利用性質。
- 3) 佈署逾時情形應盡量避免。佈署前應妥善規劃佈署與還原手順，整體作業時間應考慮還原及作業檢核點的作業時間。於佈署過程中，若佈署人員確認無法於原訂作業時間內完成，應立即回報上一級主管即進入還原程序，並於第一時間召開會議進行檢討。

## 政策規範

### 二、佈署作業

#### 作業原則

- 4) 非計畫性佈署應盡量避免。若因突發事件不得不為之，必須經PM與SA人員共同確認非計畫性佈署為障礙排除的最終手段，並經客戶核可後，方得實施。非計畫性佈署皆應列為部門障礙會議之追蹤檢討事項。



# 4. AP 建構佈署作業管理規範

## 政策規範

### 二、佈署作業

#### 作業程序

- 1) 區分計畫性與非計畫性佈署
- 2) 考量前置、中期與後期作業

# 簡報完畢 謝謝大家

Thanks for your participation!

