

Topic: Documentation



User Story

Background

- Scope: 全球台積電員工 大約 70K+ 人
- Files: 100K+ Files
- Location: All over the world

User Story – 1

- As a 設備工程師
- I want to 寫設備文件, 供公司人員看
 - UI 設計部分用 EUV 部門為例
- Then 我可以與同部門共編設備文件

User Story – 2

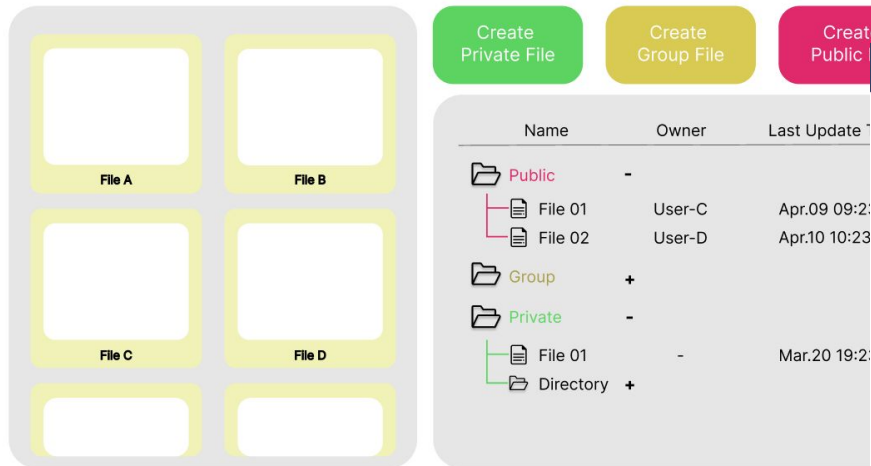
- As a 製造工程師
- I want to 看設備文件
- Then 我可以上系統, 根據權限去看設備說明文件

User Story – 3

- Given 台積電教育訓練
- When 有新員工進入台積電/所有員工
- Then 可以看到公開的公司教育訓練文件, 只有 View 的權限

UI & UX

- 這裡我採用 Figma 設計，細節說明可以到下方 Figma 連結查看
- 關於文件，我採用 Markdown 格式，可以將程式碼漂亮顯示出來，未來也可以根據使用者的偏好設定加上插件做使用。



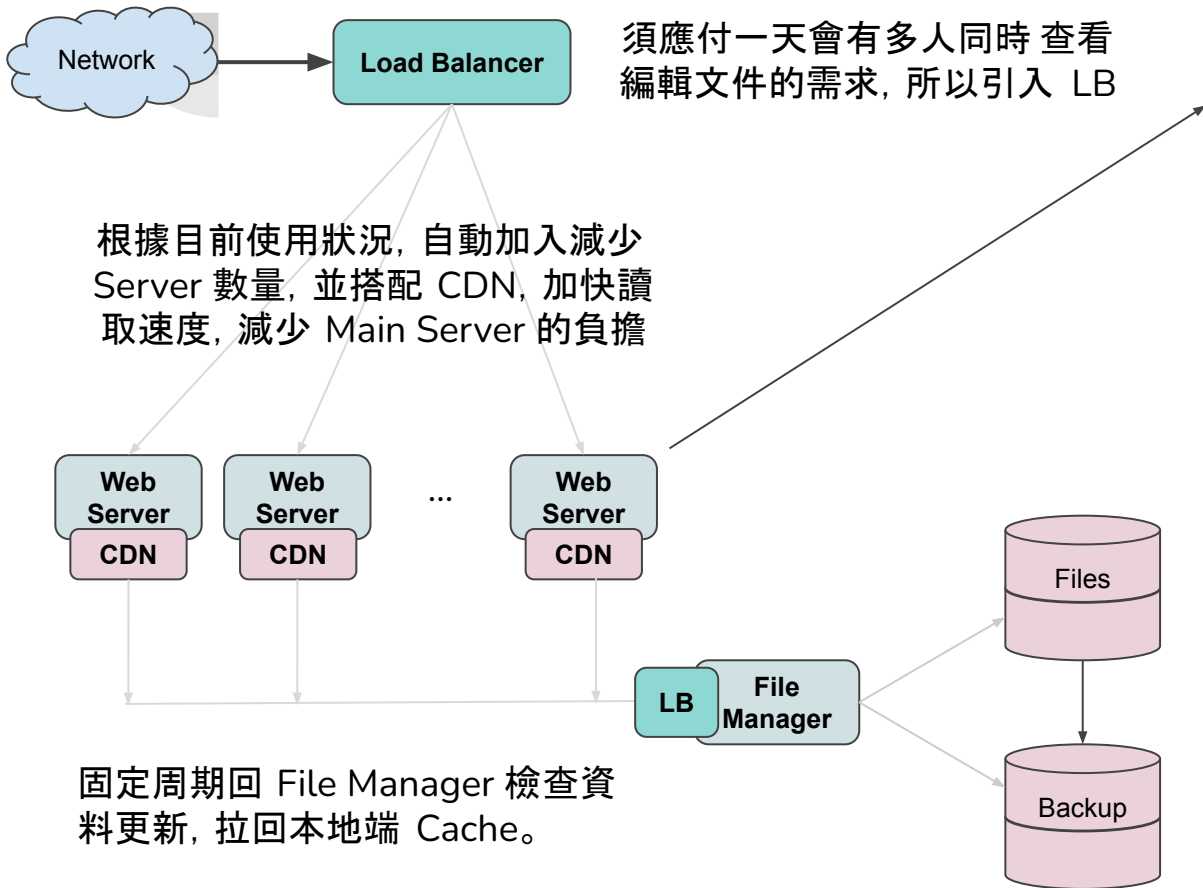
Figma: <https://shorturl.at/jpST1>

<https://www.figma.com/file/CPWiG5l9UjB1BdpYkrA9lL/TSMC-Documentation?node-id=0%3A1&t=EvxZJmM6UO6mOvXQ-1>

Application Architecture – 設計思考

- 因為 Files 為機台的規格為件與操作手冊
 - 可以假設: Read >>>> Write
 - 所以規格需以 **讀取速度** 為考量!
 - 引入 CDN
- 版本控制、查看文件歷史
 - 採用類似 Git 的版本控制
 - 可以還原, 查看版本
- 同時大量的流量請求
 - Load Balancer
- 檔案完整性
 - 自動備份
- Markdown 文件編輯
 - 可以輕量化
 - 使用者後續可自行增加主題
 - 版本比較簡單
 - 可以在前端處理, 不用開新的服務
- 文件寫回 DB 需特別處理
 - 因採用類似 Git 的方式, 沿用 Git 當版本出現碰狀的處理方式
 - 與 Git 相同機制工程師可以直接上手

Application Architecture



須應付一天會有多人同時 查看
編輯文件的需求, 所以引入 LB

根據目前使用狀況, 自動加入減少
Server 數量, 並搭配 CDN, 加快讀
取速度, 減少 Main Server 的負擔

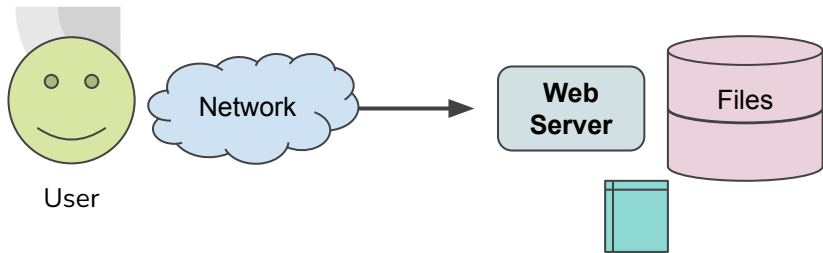
固定周期回 File Manager 檢查資
料更新, 拉回本地端 Cache。

Web Server 不僅部屬於主機端, 也
可以根據地域性, 在各個地區或是國
家部署 CDN & Web Server。

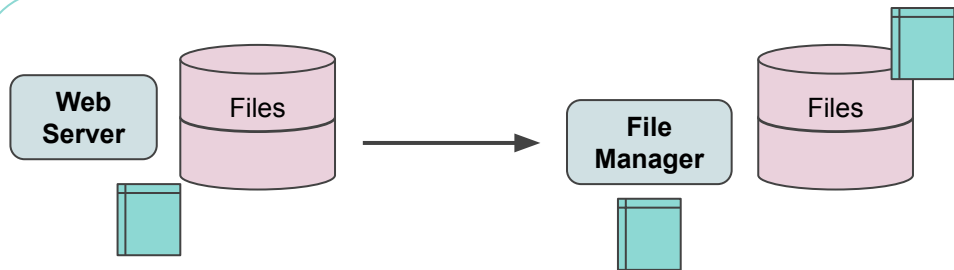
文件編輯版本問題之系統架構
將於下頁說明。

設定好備份週期與時間, 自動備份以
確保檔案能妥善儲存, 避免遺失或損
毀, 亦可搭配冷備份與異地備份, 達
到更好的資料保護。

Application Architecture



- User 編輯文件, Web Server 開啟本地端儲存
- 要推送儲存時, 將版本推送給 File Manager



- File Manager 收到版本後比較, 並收入資料庫
- 類似 Git 版本控制



- File Manager inform every WS file had changed
- 當 WS 有讀取需求時, 從 FM 拉取版本
- WS 更新 Cache

