

Factory Designer - 連結設計與執行的橋樑

文件編號: IDTF-V3.3-04

版本: 1.0

日期: 2025-10-12

作者: C.C. Lin

目錄

- [1. 簡介](#)
- [2. 專案定位](#)
- [3. 核心功能](#)
- [4. 技術架構](#)
- [5. 使用場景與工作流程](#)
- [6. 與生態系統整合](#)
- [7. 實施建議](#)

簡介

Factory Designer 是 IDTF V3.3 的核心新增功能,它填補了從「資產建模」到「工廠部署」之間的關鍵缺口。Factory Designer 是一個專注於**廠級佈局設計**的應用程式,使用 **FDL (Factory Design Language)** 作為其核心語言。

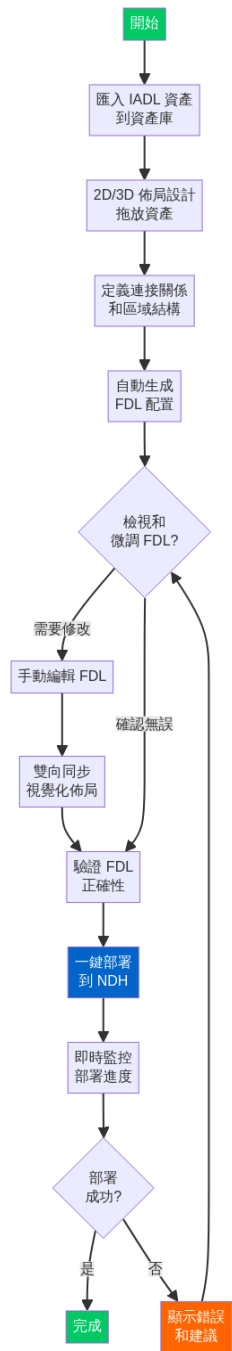
核心價值

Factory Designer 的核心價值在於:

- 視覺化設計:** 將複雜的 FDL 編寫工作轉化為直觀的拖放操作
- 自動化部署:** 一鍵將設計推送到執行環境,極大縮短從設計到上線的時間
- 即時回饋:** 與 NDH 的直接連結,讓工程師可以即時驗證設計的正確性

4. 生態閉環: 作為連結離線設計與線上執行的橋樑,完成整個 IDTF 生態系統的閉環

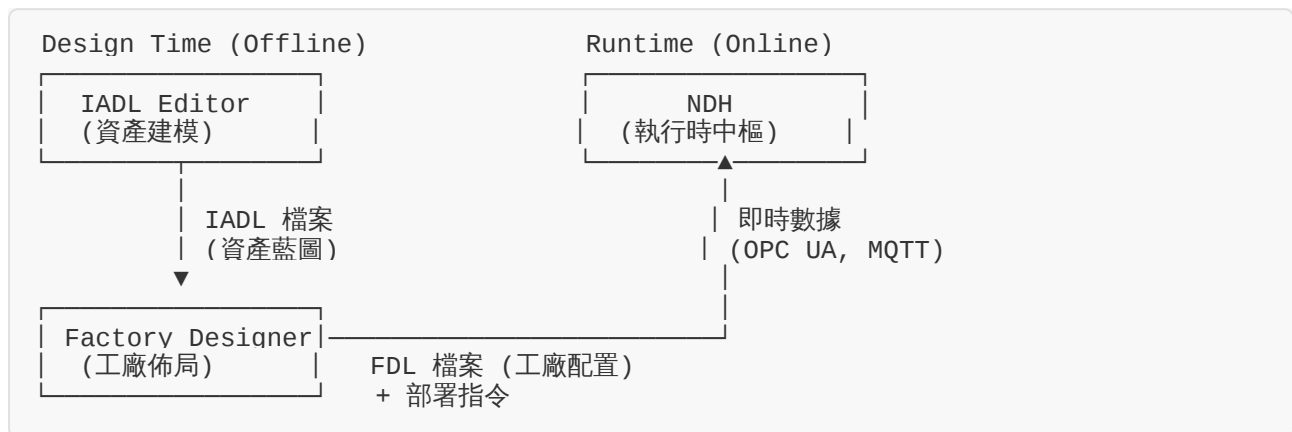
工作流程概覽



專案定位

在 IDTF 生態系統中的角色

Factory Designer 在 IDTF 生態系統中扮演關鍵的橋樑角色:



生態系統中的角色定位

- **IADL Editor:** 負責定義「單一資產」的藍圖 (例如一台泵、一個感測器)
- **Factory Designer:** 負責定義「整座工廠」的佈局,包括數百個資產實例的位置、數量和連接關係
- **NDH:** 負責在執行時管理這些資產實例,接收即時數據,並提供統一的查詢介面

Factory Designer 的獨特之處在於,它**同時連結了 IADL Editor 和 NDH**,成為設計與執行之間的關鍵橋樑。

為什麼需要 Factory Designer?

在沒有 Factory Designer 的情況下,工程師需要:

1. 在 IADL Editor 中設計資產藍圖
2. **手動編寫 FDL 檔案**,定義工廠中每個資產實例的位置、參數和連接
3. **手動將 FDL 檔案上傳到 NDH**,觸發實例化

這個過程不僅繁瑣,而且容易出錯。Factory Designer 的出現,將這個過程**視覺化**和**自動化**:

- **視覺化設計:** 工程師可以在一個類似 CAD 的介面中,拖放資產到工廠平面圖上
- **自動生成 FDL:** Factory Designer 自動根據視覺化佈局生成 FDL 檔案

- **一鍵部署:** 點擊「部署」按鈕,FDL 檔案直接推送到 NDH,觸發實例化

核心功能

Factory Designer 的核心功能可以分為以下四個模組:

1. 資產庫管理 (Asset Library Management)

功能描述

Factory Designer 需要一個「資產庫」,用於儲存和管理從 IADL Editor 匯入的資產藍圖。

具體實現

匯入 IADL 檔案: - 支援從本地檔案系統或雲端儲存 (如 S3) 匯入 IADL 檔案 - 解析 IADL 檔案,提取資產的關鍵資訊 (名稱、類型、參數、3D 模型等)

資產預覽: - 在資產庫中顯示每個資產的縮圖 (從 3D 模型生成) 和基本資訊 - 支援搜尋和篩選功能 (例如,按類型、標籤或製造商篩選)

版本管理: - 支援同一資產的多個版本 (例如,Pump v1.0, Pump v1.1) - 在佈局設計時,可以選擇使用特定版本的資產

範例介面

資產庫			
「搜尋框」		「篩選: 類型 製造商」	
Pump v1.0	Sensor v2.1	Valve v1.5	

2. 工廠佈局設計器 (Factory Layout Designer)

功能描述

這是 Factory Designer 的核心介面,提供視覺化的工廠佈局設計環境。

2D 佈局模式

- 提供一個類似 CAD 軟體的 2D 平面圖編輯器
- 使用者可以從資產庫中拖放資產到平面圖上
- 支援定義區域 (Area)、樓層 (Floor) 和建築物 (Building) 的階層結構
- 自動生成資產的唯一 ID (例如, `Area1_Pump_001`)

3D 佈局模式 (進階功能)

- 整合 Three.js 或 Babylon.js 等 WebGL 引擎,提供 3D 視覺化
- 支援從 BIM 軟體 (如 Revit) 匯入工廠建築的 3D 模型作為底圖
- 在 3D 空間中精確定位資產的位置 (x, y, z 座標)

連接與關係定義

- 支援在資產之間繪製連接線,定義管線、電纜或邏輯連接
- 自動將這些連接關係寫入 FDL 的 `relationships` 部分

操作示例

1. 從資產庫拖放 `"StandardPump"` 到平面圖
2. 系統自動分配 ID: `"Area1_Pump_001"`
3. 設定位置: `x=10, y=20, z=0`
4. 繪製管線連接到 `"Tank_001"`
5. 系統自動記錄連接關係

3. FDL 生成與編輯器 (FDL Generator & Editor)

功能描述

Factory Designer 會根據視覺化的佈局,自動生成 FDL 檔案。

自動生成邏輯

```
# 自動生成的 FDL 範例
factory_layout:
- area: ProductionFloor1
  building: MainBuilding
  floor: 1
  instances:
    - type: StandardPump          # 從 IADL 資產庫引用
      count: 5                    # 使用者在佈局中放置了 5 個
      naming_prefix: "Pump_"
      location: "x:10, y:20, z:0" # 從 2D/3D 佈局中提取
      initial_params:
        power: 15
        status: "Stopped"
    - type: TemperatureSensor
      count: 20
      naming_prefix: "TempSensor_"
      location: "x:15, y:25, z:2"
  relationships:
    - from: "Pump_001"
      to: "Tank_001"
      type: "pipe"
    properties:
      diameter: 50mm
      material: "stainless_steel"
```

內建 FDL 編輯器

- 提供語法高亮的文字編輯器,讓進階使用者可以手動微調 FDL
- 支援 FDL 語法驗證,即時提示錯誤
- **雙向同步**: 手動修改 FDL 後,視覺化佈局也會相應更新

4. NDH 整合與部署 (NDH Integration & Deployment)

功能描述

Factory Designer 的最大創新在於其與 NDH 的直接連結。

NDH 連線配置

- 在設定中輸入 NDH 的 API 端點 (例如 `http://ndh.factory.com:8000`)
- 支援 API 金鑰或 OAuth 2.0 認證
- 支援多個 NDH 環境 (開發、測試、生產)

一鍵部署

點擊「部署到 NDH」按鈕,Factory Designer 會:

1. 驗證 FDL 檔案的正確性
2. 檢查語法錯誤
3. 驗證資產引用是否存在
4. 檢查連接關係的完整性
5. 推送到 NDH API ``http POST /api/deploy Content-Type: application/json
Authorization: Bearer

```
{ "fdl": "...", "environment": "production", "dry_run": false } ``
```

1. NDH 實例化資產物件
2. NDH 接收 FDL 後,觸發實例化引擎
3. 開始創建資產物件實例
4. 建立數據連接
5. 啟動數據採集

即時狀態監控

- 透過 WebSocket 連線,即時接收 NDH 的部署進度和狀態更新
- 顯示每個資產實例的創建狀態 (Pending, Active, Error)
- 如果部署失敗,顯示詳細的錯誤訊息和建議的修復方法

回饋與迭代

- 部署完成後,可以從 NDH 查詢實際運行中的資產實例資料
 - 支援「反向同步」: 將 NDH 中的實際配置拉回 Factory Designer,用於故障排查或版本比對
-

技術架構

技術棧選擇

前端 (推薦 Web 應用)

- **框架:** React 或 Vue.js (提供響應式的使用者介面)
- **2D 繪圖:** Konva.js 或 Fabric.js (Canvas 操作)
- **3D 視覺化:** Three.js 或 Babylon.js (WebGL 渲染)
- **狀態管理:** Redux 或 Vuex
- **UI 組件庫:** Ant Design 或 Material-UI

後端 (可選,用於複雜邏輯)

- **框架:** Node.js (Express.js) 或 Python (FastAPI)
- **用途:** FDL 驗證、版本控制、使用者管理

資料儲存

- **本地儲存:** IndexedDB (瀏覽器端) 或 SQLite (桌面應用)
- **雲端儲存:** PostgreSQL + S3 (用於團隊協作)

與 NDH 通訊

- **協議:** RESTful API (部署) + WebSocket (即時狀態)
- **資料格式:** JSON (FDL 檔案以 JSON 格式傳輸)

核心模組設計

1. AssetLibraryModule

負責 IADL 檔案的解析和資產庫的管理。

主要功能: - IADL 檔案解析 - 資產元數據管理 - 3D 模型載入和預覽 - 版本控制

2. LayoutDesignerModule

提供 2D/3D 佈局編輯器的核心邏輯。

主要功能: - Canvas 繪圖和互動 - 資產拖放和定位 - 連接線繪製 - 區域和樓層管理

3. FDLGeneratorModule

根據佈局資料自動生成 FDL 檔案。

主要功能: - 佈局數據序列化 - FDL 語法生成 - 語法驗證 - 雙向同步

4. NDHClientModule

封裝與 NDH API 的所有互動邏輯。

主要功能: - HTTP 客戶端 - WebSocket 連線 - 認證和授權 - 錯誤處理

5. VersionControlModule

整合 Git 或自訂的版本控制系統,管理工廠佈局的歷史版本。

主要功能: - 版本歷史記錄 - 差異比較 - 回滾和恢復 - 分支管理

使用場景與工作流程

典型工作流程

步驟 1: 匯入資產藍圖

- 工程師從 IADL Editor 匯出 IADL 檔案
- 在 Factory Designer 中匯入這些 IADL 檔案到資產庫
- 預覽資產的 3D 模型和參數

步驟 2: 設計工廠佈局

- 在 2D/3D 佈局設計器中,拖放資產到工廠平面圖上
- 定義區域、樓層和建築物的階層結構

- 繪製資產之間的連接關係
- 調整資產的位置和方向

步驟 3: 生成 FDL 配置

- Factory Designer 自動生成 FDL 檔案
- 工程師可以在內建編輯器中檢視和微調 FDL
- 驗證 FDL 的正確性

步驟 4: 部署到 NDH

- 點擊「部署到 NDH」按鈕
- Factory Designer 將 FDL 推送到 NDH 的 API
- NDH 接收 FDL 並開始實例化資產物件

步驟 5: 監控與驗證

- 透過 Factory Designer 的監控面板,即時查看部署進度
- 驗證所有資產實例都已成功創建並處於活躍狀態
- 檢查數據連接是否正常

步驟 6: 迭代與優化

- 根據實際運行情況,在 Factory Designer 中調整佈局
- 重新部署更新後的 FDL 到 NDH
- 持續優化工廠配置

進階場景

多廠部署

- 使用同一套資產藍圖,在 Factory Designer 中設計多個工廠的佈局
- 分別生成不同的 FDL 檔案,部署到不同地點的 NDH 實例
- 實現標準化的多廠管理

虛擬調試

- 在部署到實體工廠之前,先部署到一個模擬的 NDH 環境
- 進行虛擬調試,驗證控制邏輯和資料流程
- 降低實體部署的風險

災難恢復

- 當實體工廠發生故障時,可以從 Factory Designer 的版本庫中快速恢復之前的工廠配置
- 支援快速重建和恢復

設計變更管理

- 使用版本控制追蹤所有設計變更
 - 支援設計審查和批准流程
 - 記錄變更歷史和原因
-

與生態系統整合

與 IADL Editor 的整合

資料流向: IADL Editor → IADL 檔案 → Factory Designer

整合方式:

1. 檔案匯入:
2. Factory Designer 提供「從 IADL Editor 匯入」功能
3. 支援批次選擇 IADL 檔案
4. 自動解析和驗證
5. API 整合 (進階):
6. IADL Editor 可以提供一個「推送到 Factory Designer」的功能
7. 直接透過 API 將 IADL 檔案傳送到 Factory Designer 的資產庫
8. 支援即時同步

與 NDH 的整合

資料流向: Factory Designer → FDL 檔案 → NDH

整合方式:

1. **RESTful API:**

2. NDH 提供一個 `/api/deploy` 端點,接收 FDL 檔案 (JSON 格式)

3. 支援部署驗證 (dry-run 模式)

4. 返回部署任務 ID

5. **WebSocket:**

6. NDH 提供一個 WebSocket 端點,用於推送部署進度和狀態更新

7. 即時通知部署成功或失敗

8. 提供詳細的錯誤訊息

9. **認證與授權:**

10. 使用 API 金鑰或 OAuth 2.0 確保只有授權的 Factory Designer 實例可以部署到 NDH

11. 支援角色基礎的存取控制 (RBAC)

與 BIM/CAD 系統的整合 (可選)

資料流向: Revit/AVEVA E3D → 3D 模型 → Factory Designer

整合方式:

- Factory Designer 可以匯入 IFC 或 USD 格式的 3D 建築模型,作為佈局設計的底圖
 - 這使得工程師可以在真實的工廠建築模型上精確定位資產
 - 支援 BIM 模型的樓層和區域資訊提取
-

實施建議

開發優先級

第一階段: MVP (最小可行產品)

- 資產庫基本功能 (匯入、預覽、搜尋)
- 2D 佈局設計器 (拖放、定位)
- 基本 FDL 生成
- 簡單的 NDH 部署 (REST API)

第二階段: 核心功能完善

- 連接關係定義
- FDL 編輯器和雙向同步
- WebSocket 即時監控
- 版本控制基礎

第三階段: 進階功能

- 3D 佈局設計器
- BIM 模型匯入
- 多工廠支援
- 完整的版本控制和協作功能

技術挑戰與解決方案

挑戰 1: 大規模資產的性能

解決方案: - 使用虛擬滾動和分頁載入 - 實施 LOD (Level of Detail) 策略 - Canvas 渲染優化

挑戰 2: 實時協作

解決方案: - 使用 CRDT (Conflict-free Replicated Data Types) - WebSocket 廣播更新 - 樂觀更新和衝突解決

挑戰 3: FDL 與視覺化的同步

解決方案: - 建立統一的數據模型 - 實施觀察者模式 - 使用 JSON Schema 驗證

最佳實踐

1. **模組化設計:** 保持各模組的獨立性和可測試性
 2. **用戶體驗:** 提供直觀的拖放介面和即時回饋
 3. **錯誤處理:** 提供清晰的錯誤訊息和修復建議
 4. **性能優化:** 針對大規模工廠進行性能優化
 5. **文檔完善:** 提供完整的使用文檔和 API 文檔
-

總結

Factory Designer 是 IDTF V3.3 生態系統的關鍵組件,它填補了從資產建模到工廠部署之間的缺口,實現了真正的端到端工作流程。

核心貢獻

1. **視覺化設計:** 將複雜的 FDL 編寫轉化為直觀操作
2. **自動化部署:** 一鍵將設計推送到執行環境
3. **即時回饋:** 與 NDH 直接連結,即時驗證設計
4. **生態閉環:** 完成 IDTF 生態系統的最後一塊拼圖

下一步

- 閱讀 [FDL 規範](#) 了解工廠設計語言
 - 閱讀 [NDH 架構](#) 了解數據中樞設計
 - 查看 [實施指南](#) 了解實施細節
-

Factory Designer 讓「一次建模,到處使用」的理想成為現實! 🚀