



A Model-Based Method for Organizing Tasks in Product Development



出處：Research in Engineering Design (1994) 6:1-13

作者：Eppinger S.D., Whitney D.E., Smith R.P. and.
Gebala D. A

報告者：鄭欣卉

指導老師：童超塵教授



Contents

- Introduction
- Sequencing Development Tasks
- The Design Structure System
- Extensions to the Design Structure Matrix Representation
- Exploring Design Structure Data
- Strategies for Designing Better Design Processes
- A Developmental Process Strategy
- Conclusion

Introduction

- 大部分同步工程的文獻描述多功能小團隊(5~10人)於產品開發上，其藉由設計、製造、維修工程師和其他相關的人，緊密地工作在一起，相互理解技術問題，以提早發現和解決問題。
- 但是，同步工程用在複雜專案(數百人)時，其發展被分解成很多小專案，分別由小團隊執行。
- 本研究目標是要協助複雜的同步工程計劃達到小團隊執行同步工程，可以很容易發現技術間的交互作用。

Introduction

- 與傳統專案管理與工程設計研究不同的兩大方向
 - (1) 發展過程之詳細模型，瞭解活動之間複雜交互作用和重複設計。
 - (2) 調整發展程序，試圖減少大專案的複雜性。
- 將設計流程繪製到一個簡單的矩陣，代表設計作業間的複雜交互作用關係。
- 策略包含分割和重新排序作業，新增決策點，切割或結合的作業和其他計劃改進資訊流和決策。

Sequencing Development Tasks

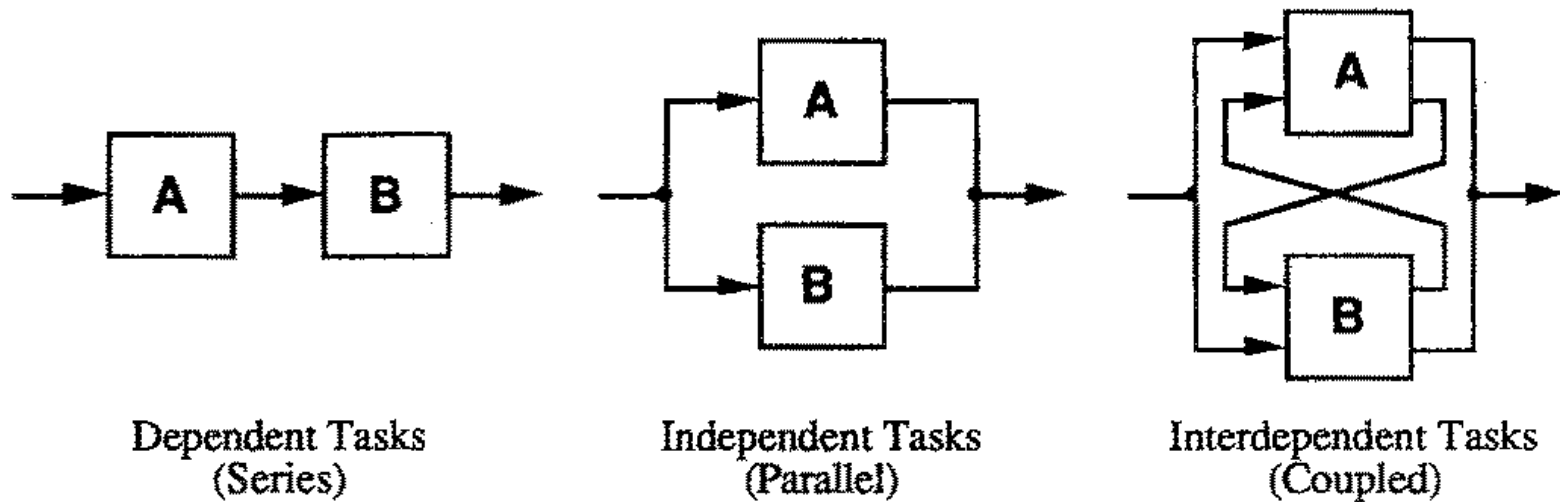


Fig. 1. Three possible sequences for two design tasks.

The Design Structure System - Steward(1981)

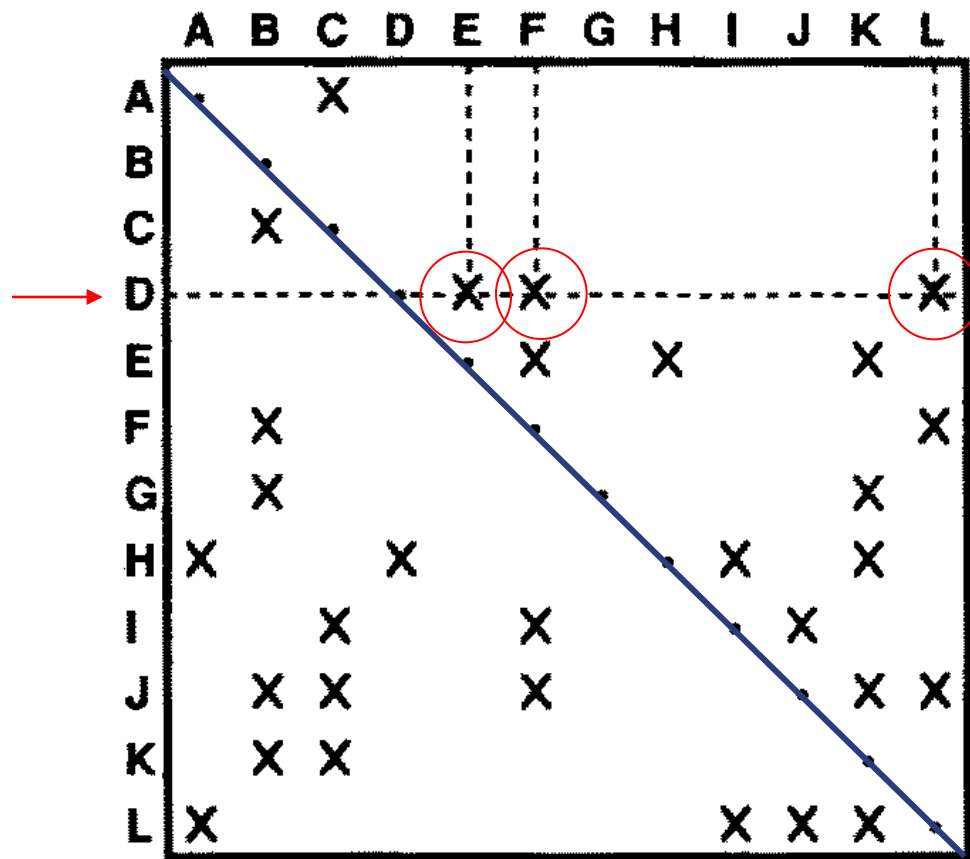


Fig. 2. A binary design structure matrix, unpartitioned.

The Design Structure System

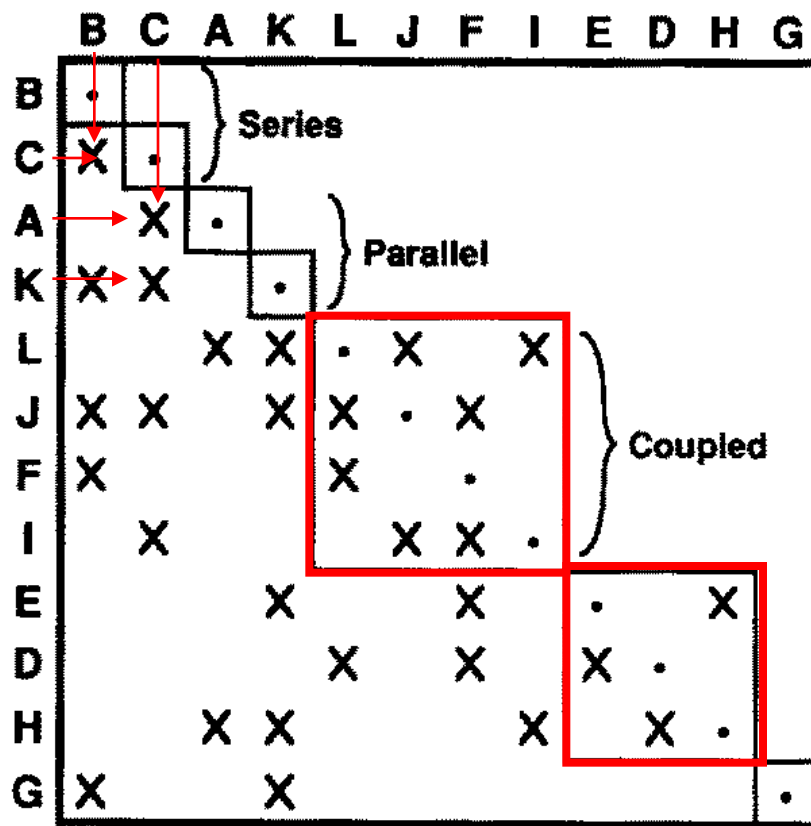


Fig. 3. The binary design structure matrix, partitioned to represent a sequence.

The Design Structure System

- Algorithm：只能處理循序和並行的流程。

DSM：能處理循序、並行與成對的流程。

- 當沒辦法使DSM成為下三角矩陣時，則進行最小化對角線的區塊。將區塊拆解成單一活動，使專案簡單化。

The Design Structure System

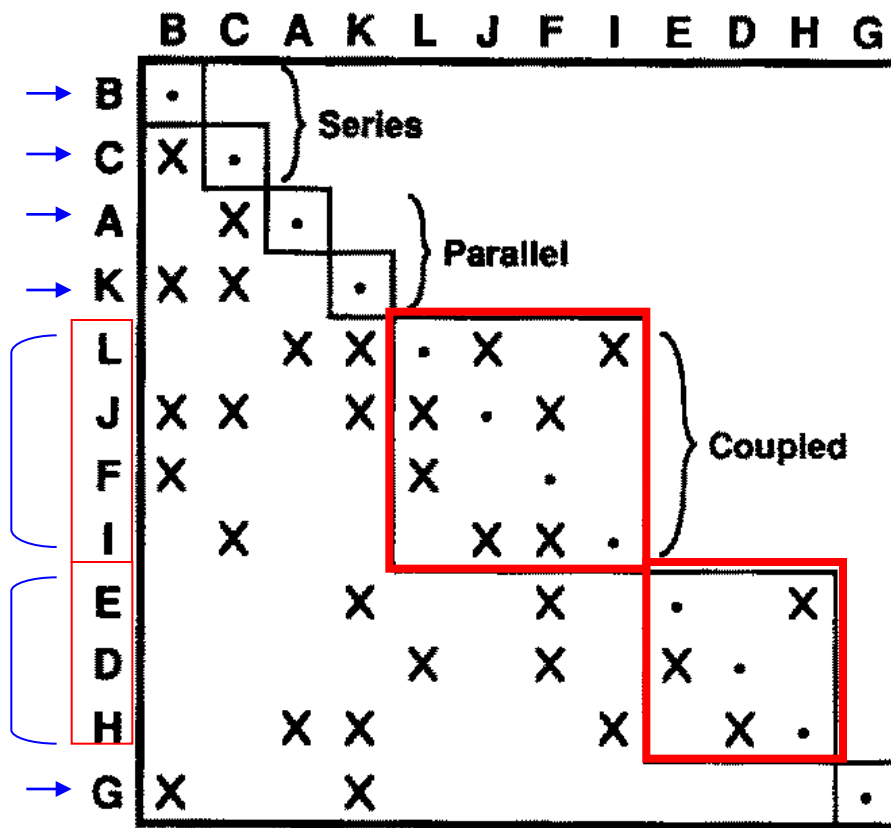


Fig. 3. The binary design structure matrix, partitioned to represent a sequence

這種方法隱藏真正設計的問題，並且阻止任何應用其他技術進一步改進設計程序的機會。



The Design Structure System

- 因為成對作業區塊代表設計重疊，也許是很重要的工作，必需選擇適當的作業排序。
- Steward 提出的tearing不是藉由移除記號來改變矩陣，而是簡化程序以找到適當的區塊間的排序。

Extensions to the Design Structure Matrix Representation

- 二元素設計結構矩陣僅代表活動間優先次序的相關性。
- 本研究發現二元矩陣有很多低相依性，而且沒有活動模型的完成時間，以計算整體專案時間。
- 本研究將明確地衡量相依程度和活動時間，期能更精準地分析程序加以改進。

Extensions to the Design Structure Matrix Representation

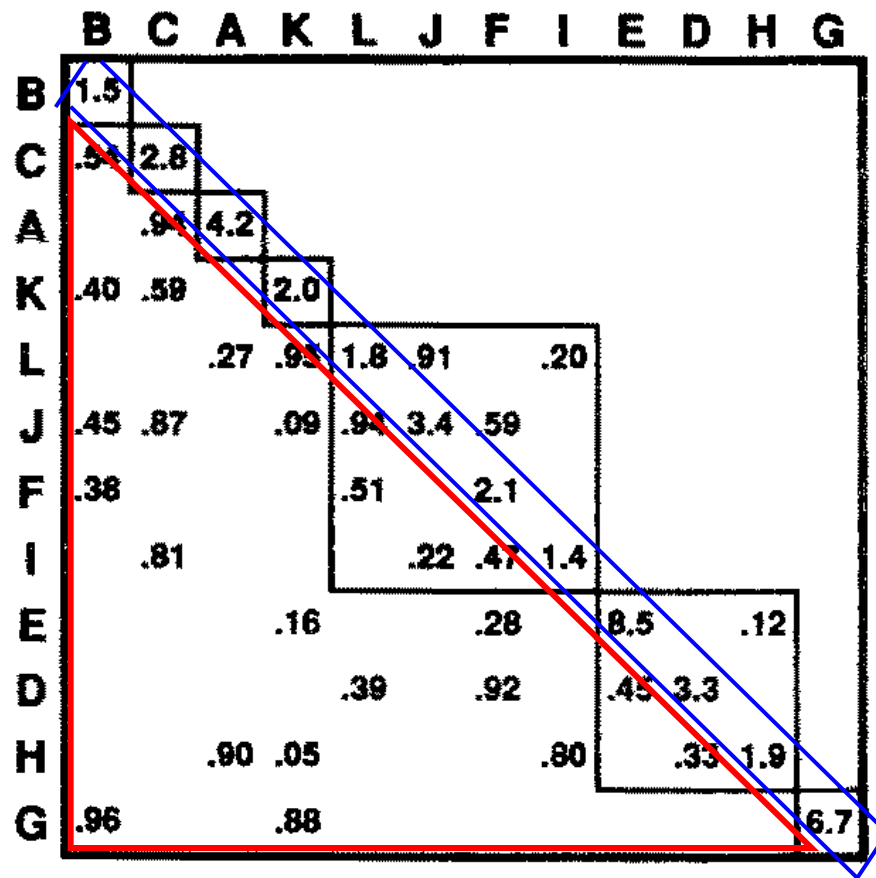


Fig. 4. A numerical design structure matrix.



Extensions to the Design Structure Matrix Representation

- 分割和tearing矩陣原則是重新安排作業，期望能達到最小化成對作業間重複的回饋。
 - 重要的回饋標記：靠近對角線。
 - 相依性小者：遠離對角線。
- 相依程度弱
 - 活動十分依賴另一個活動的資訊且可以預測。
 - 活動依賴大部份不可預測的小資訊。
- 相依程度強
 - 活動需要資訊且其影響和變異很大者。

Extensions to the Design Structure Matrix Representation

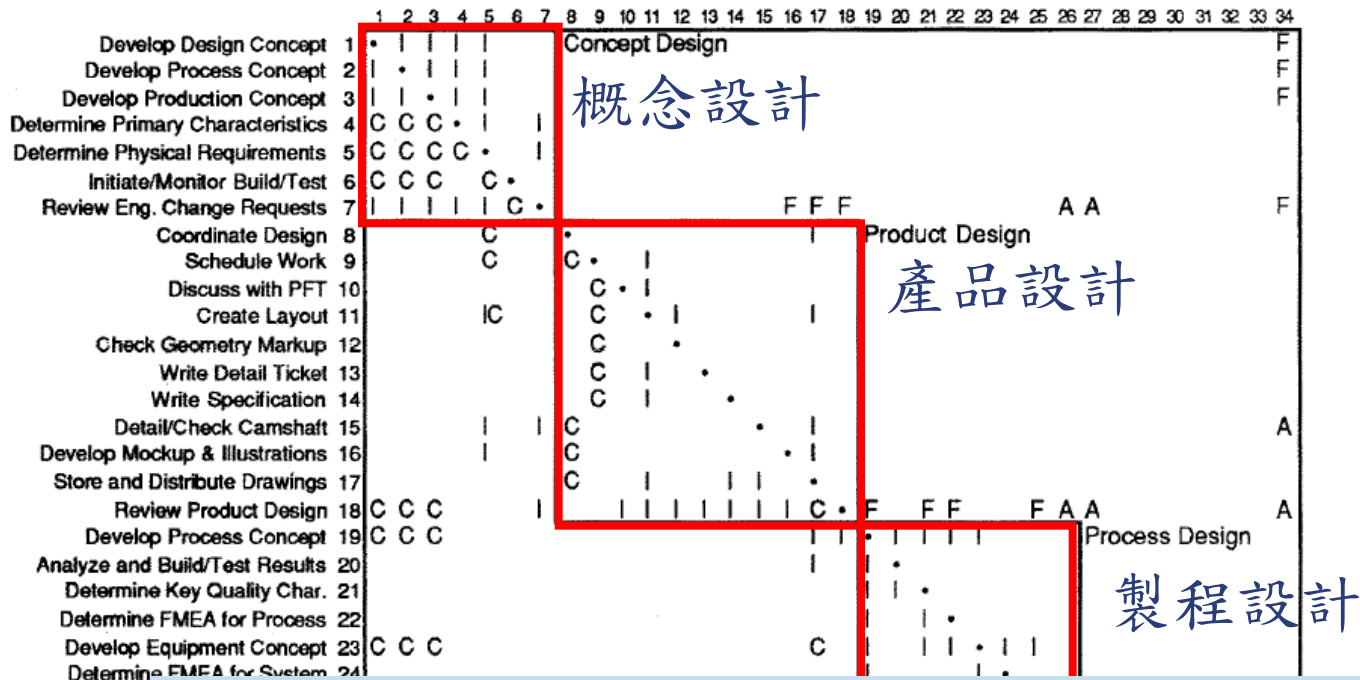
- 開發設計結構的數值模型是相當困難的，但可藉由提出計劃尋找代表作業相依的數值。
 - ▣ 收集初始數據，確定其相依關係。
 - ▣ 工程師們面談找區塊裡的相依程度。
- 本研究開發兩個設計重疊模型，是為了預測重疊時間和估計作業工期。



Exploring Design Structure Data

- **Task-Level Design Description**
- **Parametric-Level Design Description**
- **Hybrid Design Models**

Task-Level Design Description



I和C標誌：兩種不同型態的相依作業
 上三角矩陣 - F標誌：設計程序中代表回餽
 增加A標誌：改進設計程序，提供資訊流

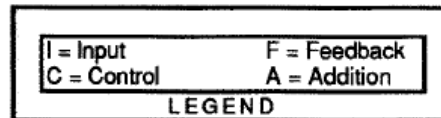


Fig. 5. Task-based design structure matrix for component design.

Task-Level Design Description

- 應用分割演算法在作業排序上只有微小的改變。
- 在這種情況下，本研究進行底層結構流程的排序。
- 4個設計團隊，分別執行一個主要作業。
- 對角線上面的標誌代表回饋訊息傳遞路徑需要長的前置時間，或“設計重工”，這些都要盡量避免。

Task-Level Design Description

- 為了創造一個基於作業數值的DSM，本研究發展一個簡單四層級計畫，量化作業需要資訊的相依程度：

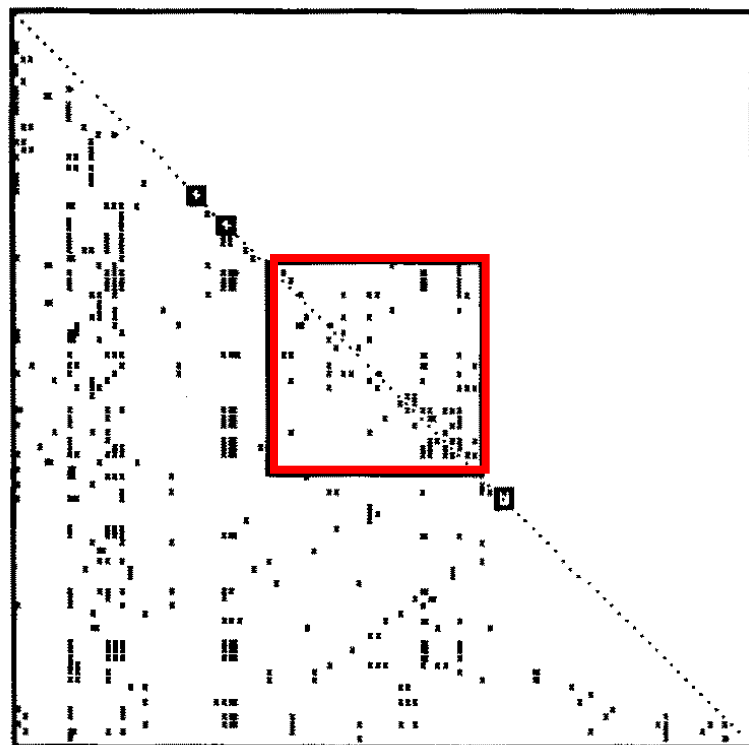
High：開始作業時必須有訊息。

Medium：結束作業時必須有訊息。

Low：只有在檢查結果一致性時需要訊息。

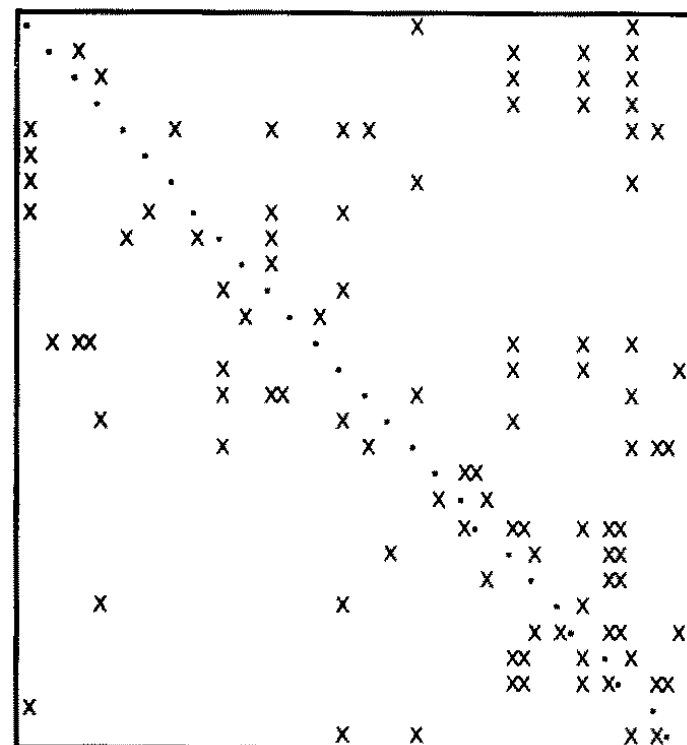
Zero：不需要資訊。

Parametric-Level Design Description



(a)

Knuckle envelope & attach pts
Pressure at rear wheel lock up
Brake torque vs. skidpoint
Line pressure vs. brake torque
Splash shield geometry—front
Drum envelope & attach pts
Bearing envelope & attach pts
Splash shield geometry—rear
Air flow under car/wheel space
Wheel material
Wheel design
Tire type/material
Vehicle deceleration rate
Temperature at components
Rotor cooling coefficient
Lining—rear vol and area
Rotor width
Pedal attach pts
Dash deflection
Pedal force (required)
Lining material—rear
Pedal mechanical advantage
Lining—front vol & swept area
Lining material—front
Booster reaction ratio
Rotor diameter
Rotor envelope & attach pts
Rotor Material



(b)

分割後的整體結構

Fig. 6. Parameter-based design structure matrix for brake system design.

在設計過程中以工程師進行訪問，要求設計的參數必須是已知的，以便確定其他設計參數。



Parametric-Level Design Description

- 剎車系統的矩陣顯示，大約有三分之一的參數是客戶的要求。
- 設計的困難點是30多個作業形成了中心的矩陣的區塊。

Hybrid Design Models

- 同時包含作業層級和參數層級的訊息。
- 高層級工作模式：忽視太多重要的技術細節
低層級的參數模型：缺乏總體架構。
- 混合模式包含兩種類型的特徵。

Hybrid Design Models

- 研究發現，作業往往被描述為一組參數。通過使用混合模式，找出成對作業，有足夠的細節來考慮藉由重組參數成一個新作業，來重新界定作業。
- 首先建立設計過程的模式，再建構參數模式，最後重新分區矩陣到新的工作組。



Strategies for Designing Better Design Processes

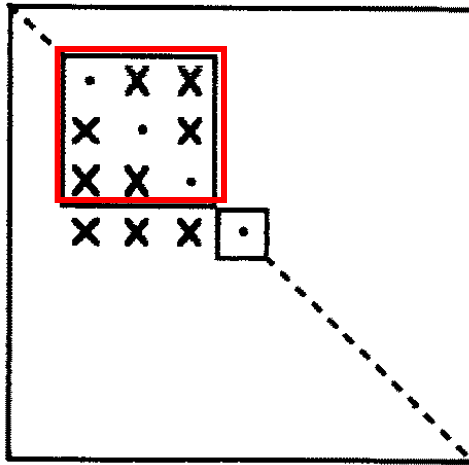
- 藉由重新界定的作業，改變其固有的成對作業關係。
- 消除成對作業與加入成對作業。

Decoupling Tasks To Speed Design

- 不嚴謹的成對組合可以藉由人工分離分成兩個或多個更小的，更緊密成對組群。
- 從矩陣中刪除一個或多個任務相依關係。

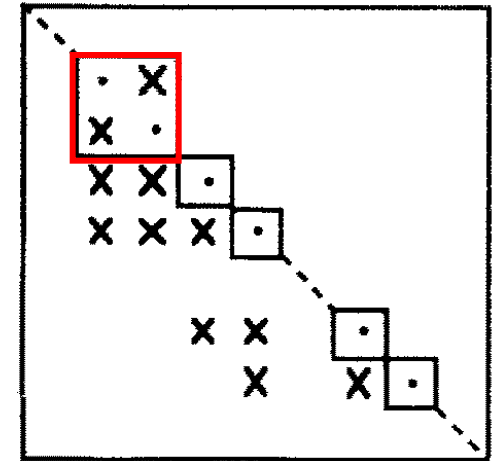
Decoupling Tasks To Speed Design

Casing Design
Wiring Details
Optical Layout
First Prototype



(a)

Casing Design
Optical Layout
Wiring Plan
First Prototype



Wiring Revision
Second Prototype

(b)

Fig. 7. Instrument design task matrices.

設計師認為必須更加速交付雛型，可以接受凌亂的電線的雛型。

接線不在設計過程，只有外殼和光學工程師參與，使設計更迅速地完成。最後配線完成了第二個雛型。

Decoupling Tasks To Speed Design

- 分割是成功的，因為接線通常是單方面的。從接線到其他兩個設計作業的回饋並不那麼重要。
- 批評這種加快設計過程的品質較差。
- 但我們發現最終達到比較好的品質，由於兩個因素：
 - 外殼和光學設計人員提供了一些關於接線限制的了解。
 - 在第二個雛型構建前，容易執行接線修改。

Increasing Coupling to Improve Design Quality

- 增加成對作業策略是同步工程和為製造而設計 (DFM)。
- 早期設計階段時納入生產專業技術，使設計產品製造容易，但增加成對作業，導致產品發展速度慢。
- 整體設計時間可以減少，因為末期重疊會減少。
- 同步工程，加強雙方的關係，重整作業，可以提高品質和加速專案發展。

Increasing Coupling to Improve Design Quality

無法避免重新設計來解決生產問題。

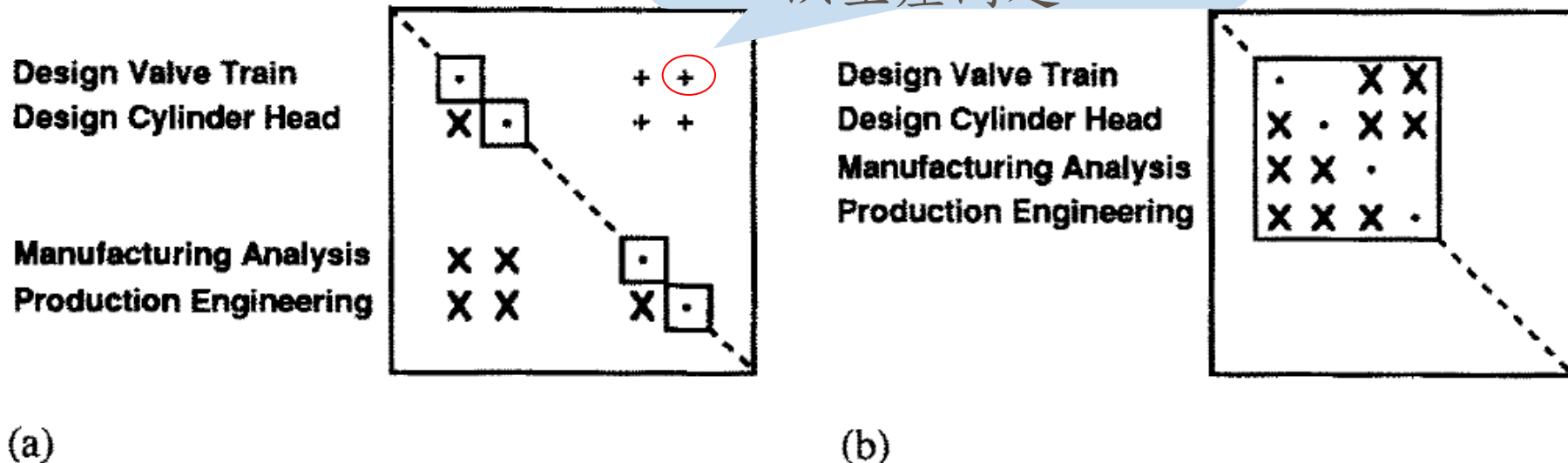


Fig. 8. Sequential and concurrent design procedures.

傳統的設計過程

產品設計人員設計作業
與製造工程師作業無關

同步設計過程

兩項作業必須同時執行

A Developmental Process Strategy

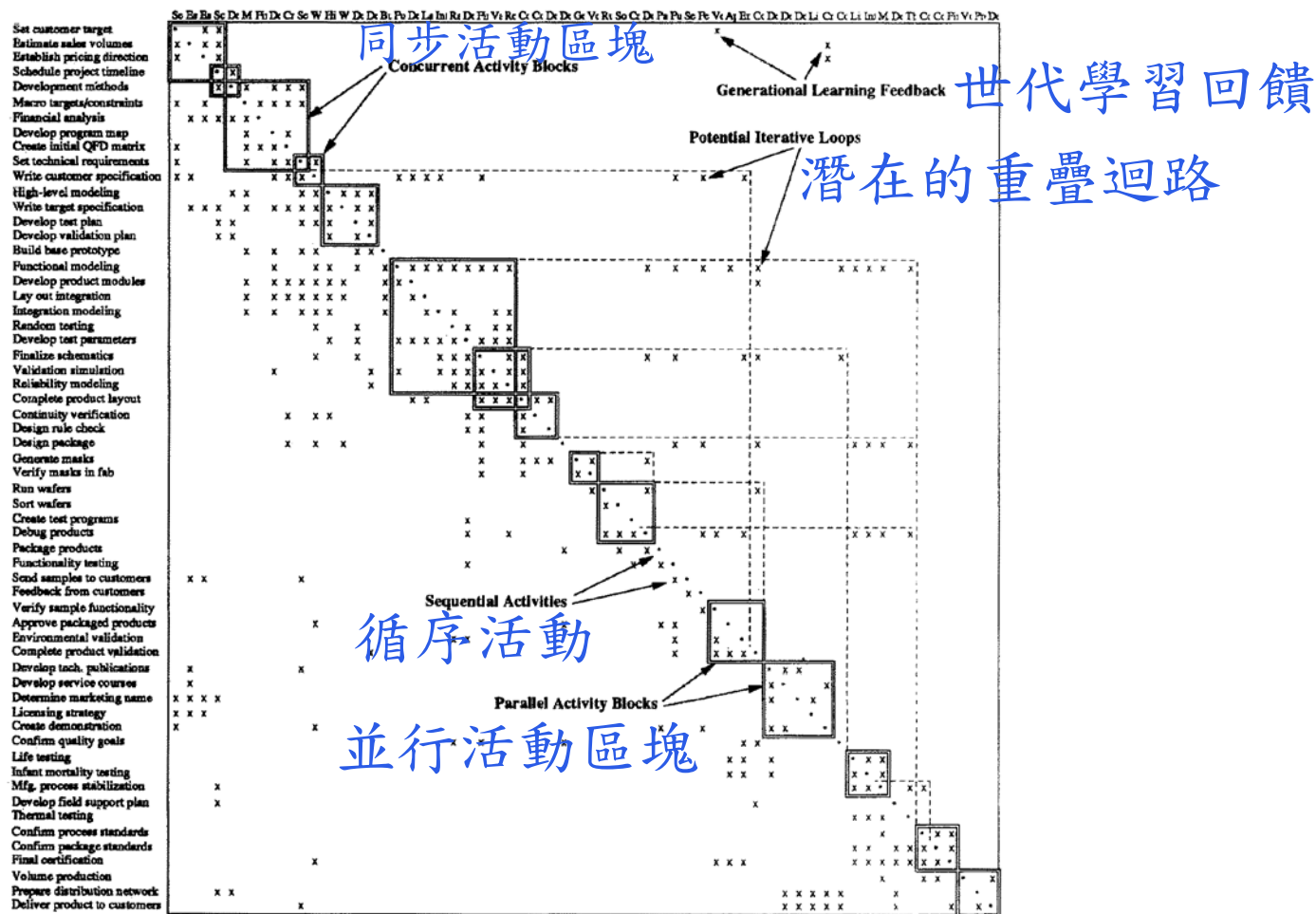


Fig. 9. Structured development process strategy (semiconductor design).

A Developmental Process Strategy

- 有效的設計管理策略，它代表一種混合排序和同步計劃。
- 此發展程序中最重要元素：系統層級的回饋(區塊對角以上)，可能包含昂貴的重工設計，必須非常仔細地進行管理以加速重工，並儘可能避免需要再重工。
- 世代學習回饋：在專案以後的經驗教訓，必須傳遞給下一代的產品。

Conclusion

- 工業產品開發過程包含許多相關的工程設計程序。程序改善第一步驟是建立模型。
- 一個重要擴展，重新排序是藉由分解成參數重新定義活動和重新組合成新的作業。
- 矩陣圖形顯示所有資訊流和更容易看到成對作業的困難點和不必要的延遲，很容易糾正一些問題。

Conclusion

- 由此產生的矩陣也可以作為一種管理工具，使工程設計工作著重於關鍵重疊。
- 研究和產業之間的互動將發展電腦工具，使管理人員找到最佳途徑，調整更加複雜的設計作業。
- 我們已經開始相關工作，建立模型解決重疊設計程序。