

# 程式人



## 用十分鐘搞懂

# 《系統分析、軟體工程、專案管理與設計模式》

陳鍾誠

2016 年 1 月 20 日

# 話說、我大學的時候

- 念的是《交通大學資訊科學系》
- 直到畢業為止、都沒有老師開設《軟體工程》和《專案管理》的課程。

# 我覺得很奇怪

- 為甚麼這麼重要的課程
- 卻沒有安排老師來教呢？

# 直到我當了資工系的老師

- 我終於瞭解為甚麼了！

原因很簡單

# 那就是

- 沒有任何老師能教這門課！

# 你一定會說

- 這些老師也太誇張了，沒辦法教就不教了嗎？

# 但是、你可曾想過

- 沒辦法教還硬教的情況 ...



# 難道

- 那樣會更好嗎？

# 問題是、為何大學資工系

- 沒有老師可以教軟體工程呢？

# 這個問題的答案

- 非常的簡單！

# 請稍微想一分鐘

- 從那些老師到底是怎麼培養出來的 ...

您就會瞭解了！

# 還是不懂的話

- 再給個提示、就是那些老師  
通常都是博士！

是博士又怎樣？

# 是博士

- 所以通常業界經驗不多！



業界經驗不多又怎樣？

# 業界經驗不多

- 所以很少有團隊合作的經驗

# 沒有團隊合作的經驗

- 又會怎樣呢？

# 沒有團隊合作的經驗

# 那你還敢讓他們

- 教你軟體工程和專案管理嗎？

# 他們有多少人

- 寫過大型專案
- 目前正在參與軟體專案開發的呢？

# 所以

- 當初交大資科不開軟體工程和專案管理
- 是對的、是對的、是對的！

# 但是今天

- 我還是要以一個大學老師的身分，談談《軟體工程、專案管理和系統分析》



# 因為

- 至少我的經驗比學生多一點點！

# 而且

- 業界的高手們，都忙著做專案，哪有時間寫文章呢？

# 話說

- 如果你所寫的程式，從來沒有超過一千行，那是完全不需要《軟體工程》的。

# 如果你總是一個人寫程式

- 那也根本不需要《專案管理》

# 如果你的問題

- 一看就知道怎麼寫，那連《系統分析》都可以省了。

# 這樣的話

- 到底甚麼時候才需要《系統分析》、  
《專案管理》和《軟體工程》呢？

# 這個問題的答案很簡單

- 就是把上面的情況都反過來
- 那就是
  - 你不是一個人寫程式
  - 你寫的程式規模不小
  - 你的問題並不是一看就知道怎麼寫了。

# 在這種情況下

- 你就需要《系統分析》、《專案管理》和《軟體工程》了



# 因為

- 不分析就不知道該怎麼寫
- 不管理就沒辦法一起寫程式
- 不用軟體工程專案就會失控

# 舉例而言

- 我大學的時候，寫作業從來不做系統分析

# 但是有一次

- 資料結構的老師出了一個《運算式微分》的作業。我馬上去寫，結果發現寫不出來。
- 後來仔細靜下心來，把《微分公式》和《鏈鎖規則》寫下來，仔細想想如何用遞迴模擬連鎖規則完成這個作業，後來就順利做完了

# 這就是

- 一個簡單的《系統分析》  
案例！

# 接著你可能會問

- 那《專案管理》又是甚麼呢？

# 關於這個問題

- 只要做過《專題》的同學都應該有所體悟

# 我大學的時候

- 專題企圖三個人用 X-Window 合寫一個資料庫管理系統
- 於是就一個人寫視窗、一個人寫資料庫引擎、一個人寫資料庫前端

# 換句話說

- 就是採用《一個蘿蔔一個坑》的戰略。



# 結果如何呢？

- 當然可想而知！
- 最後整合的時候，整合不起來
- 好不容易整合起來了，卻很難用，一點都不像資料庫管理系統

# 這就是因為

- 沒有《管理好專案》的緣故！

# 接下來

- 就剩下《軟體工程》還沒講了！

到底甚麼是軟體工程呢？

# 如果你去看教科書

- 很可能會覺得講得好像有道理
- 但就是沒辦法拿來用！

# 像是那個 CMMI

- 規定了一堆東西，就好像  
ISO-9000 一樣。

# 問題是若你照著做

- 就只會寫出一大堆應付上級的文件！
- 難道這就叫做軟體工程嗎？

靠、當然不是囉！



# 就是因為 CMMI 沒辦法用

- 所以後來甚麼 SCRUM 和 XP 就跑出來了。
- 至少這兩個還有告訴你該怎麼用！

# 這樣的話

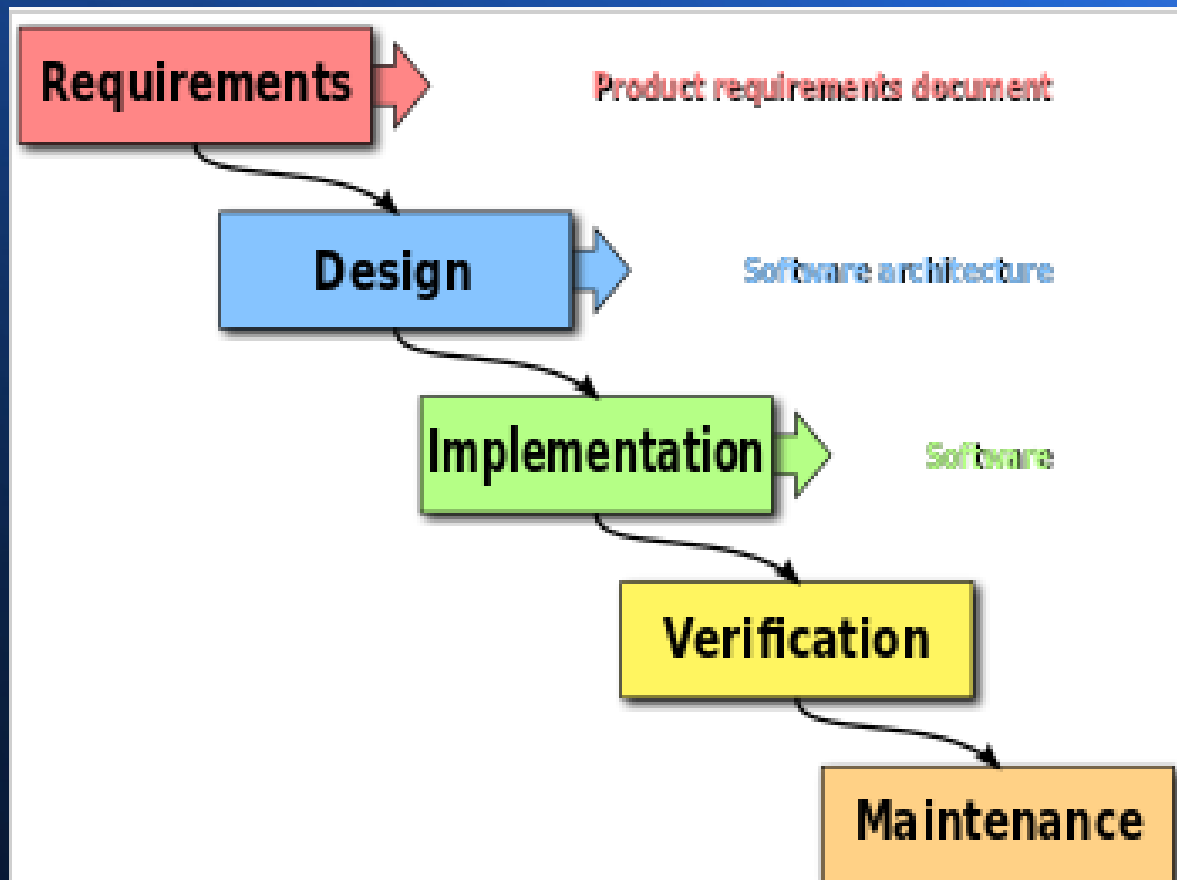
- CMMI 的設計者是腦袋壞掉嗎？

我想也不是如此！

# 如果你唸過軟體工程

- 應該對《瀑布模式》和《螺旋模式》之類的一堆模式有印象。

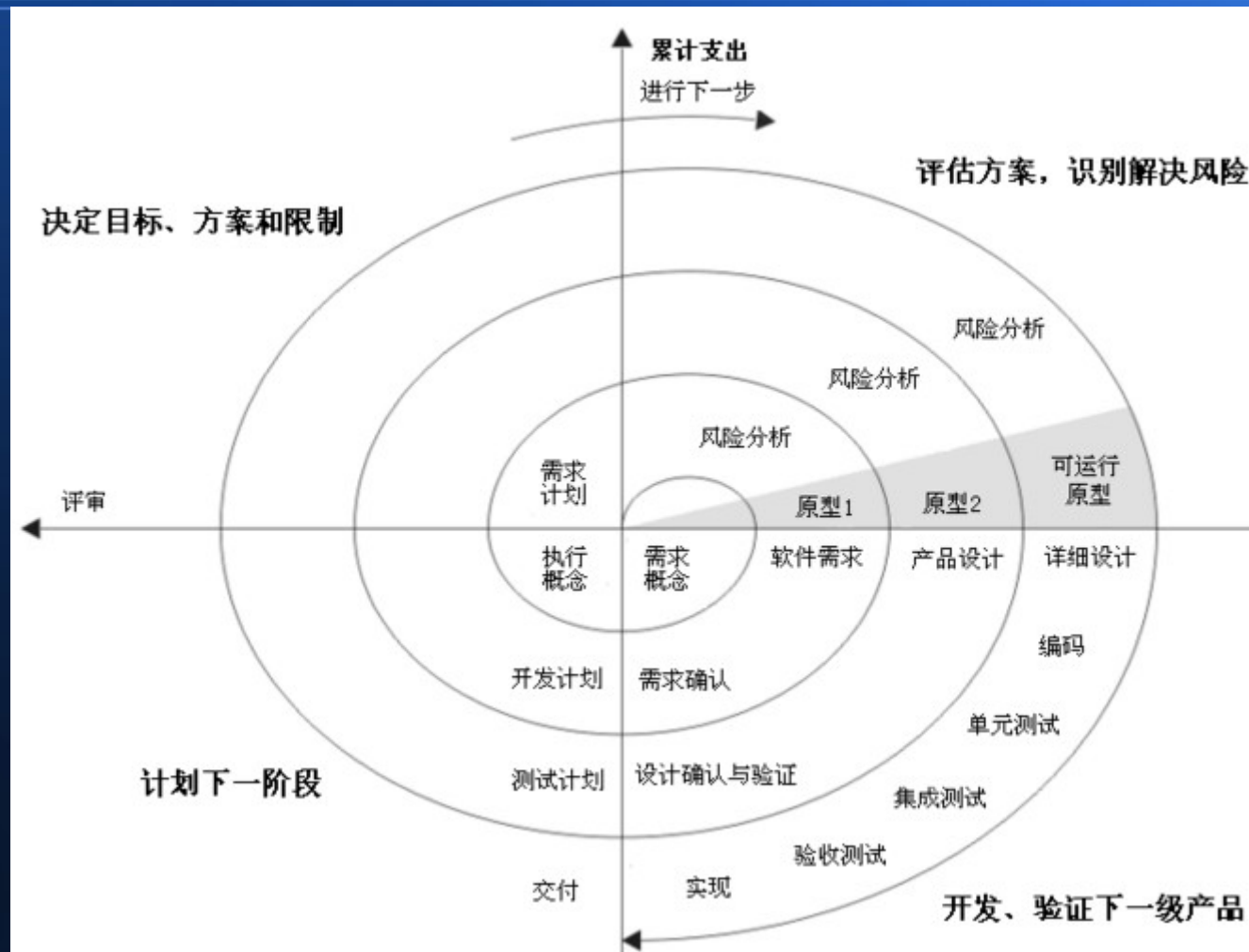
# 瀑布模式



The unmodified "waterfall model". Progress flows from the top to the bottom, like a cascading waterfall.



# 螺旋模式



# 請您想想

- 甚麼樣的專案適合用瀑布模式，什麼樣的專案適合用螺旋模式呢？

# 其實答案很簡單

- 如果您是專案負責人，必須決定採用哪一種模式。
- 那麼您該怎麼決定呢？



# 如果、那種專案

- 您已經做過  $n$  遍，現在要做第  $n+1$  遍的話
- 此時、您的目標很確定，經驗很夠，所以
  - 需求  $\Rightarrow$  分析  $\Rightarrow$  設計  $\Rightarrow$  驗證  $\Rightarrow$  上線  $\Rightarrow$  維護
  - 完全可以一氣呵成，不需要回頭！

# 這時你應該採用

- 《瀑布模式 + CMMI 》的方法
- 也就是正規軍的作戰方式！

# 如果你的專案

- 是第一次，而且成員都沒有經驗。
- 或者全世界都沒有案例可循，  
是創新型專案
  - 哪麼採用《瀑布模式》你就死定了！

為甚麼呢？

答案很簡單

# 這就好像

- 你第一次爬山的時候，沒有地圖、沒有嚮導、沒有去過、也沒有去過那座山勘察過，就決定要攀登聖母峰，而且絕不回頭，只能勇往直前，一路向上一樣。

必死無疑！

# 為了

- 不要帶著整個團隊一起去死！



# 您必須小心謹慎、步步為營

- 先從簡單的雛型開始，探查客戶的需求，然後在需求愈來愈明確，技術愈來愈成熟之後，逐漸發展出您想要的軟體。

# 而這種做法

- 就是所謂的《螺旋模式》了！

# 最後還有一個問題

- 甚麼是《設計模式》呢？

# 所謂的設計模式

- 就是其他人把《軟體開發與專案經驗》寫下來，發現某些《程式模式》對軟體開發很有幫助的時候
- 這時你直接去拿他們現成的《武功心法》來用，不用重新摸索，就是套用《設計模式》了。

# 好了

- 這就是我所理解的《軟體工程》、  
《系統分析》、《專案管理》與  
《設計模式》了

# 希望

- 這次的十分鐘系列投影片

能夠對您有所幫助！

我們下回見囉！