* 內部變異性：我們可以控制的軟體開發過程與專案管理  
  工作流程在運行時系統內部的可控制變異。
* 整個過程透過一組**策略定義**來管理系統的運行方式。
* 局部優化
* 如何控制內部變異性：
  + 使用者故事的工作粒度  
    故事應該要有優先順序的排列  
    故事應該夠小，以便在一次的開發週期內完成  
    估算故事時採用費氏數列  
    切割故事時要遵循**INVEST**規則  
    獨立性 (Independent)：避免與其他故事的依賴性。  
    可談判性 (Negotiable)：Scrum中的故事不是開始某事的合約，Stories不必太過詳細，開發人員可以給出適當的建議。  
    有價值性 (Valuable)：故事需要體現出對於使用者的價值。  
    可估計性 (Estimable)：故事應可以估計出Task的開發時間。  
    合理的尺寸 (Sized Right)：故事應該盡量小，並且使團隊盡量在一個sprint(兩週)中完成。  
    可測試性 (Testable)：故事應該是可以測試的，最好有介面可以測試和自動化測試
  + 工作項目的類型分類  
    區分類型可以有效提高可預測性。Ex: 史詩(epic) or 沙粒(grain of sand)
* 外部變異性：客戶的需求或是市場的變化
  + 依靠使用者故事映射(User Story Mapping)，來結構化使用者故事的構建。
  + P.221 圖7-4&7-5
  + 身為一個 <角色>，我想要 <活動>，從此以後 <商業價值>
* 利用「五個為什麼」來阻止發生**捨本逐末**(Shifting the Burden)的現象，找尋真正的根本原因。
* P. 225圖 7-7
* 沒有銀子彈，我們能做的只是**持續改善**。
* 持續改善並不是一件容易的事。  
  外部變異  
  對於使用者故事而言：改變代表的是更接近客戶的需求。  
  內部變異  
  對於工作流程而言：改變代表的是流程更趨近於完善。