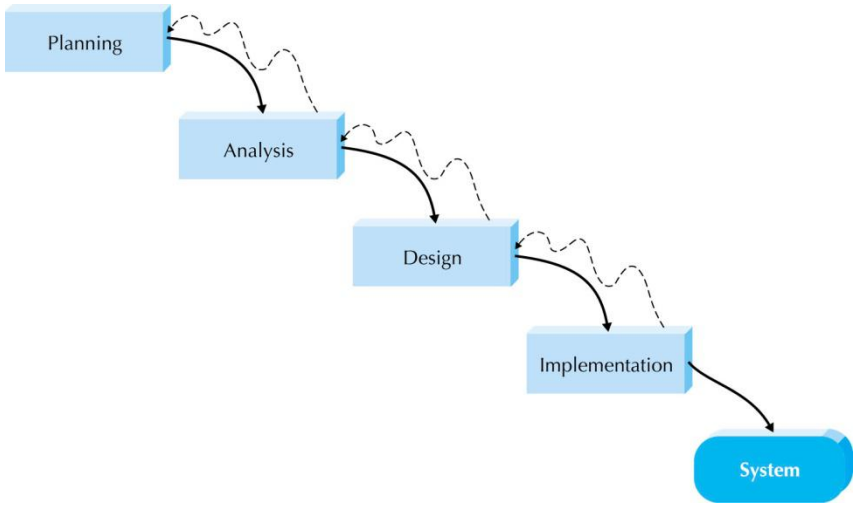
justify Identifying

Structured Development

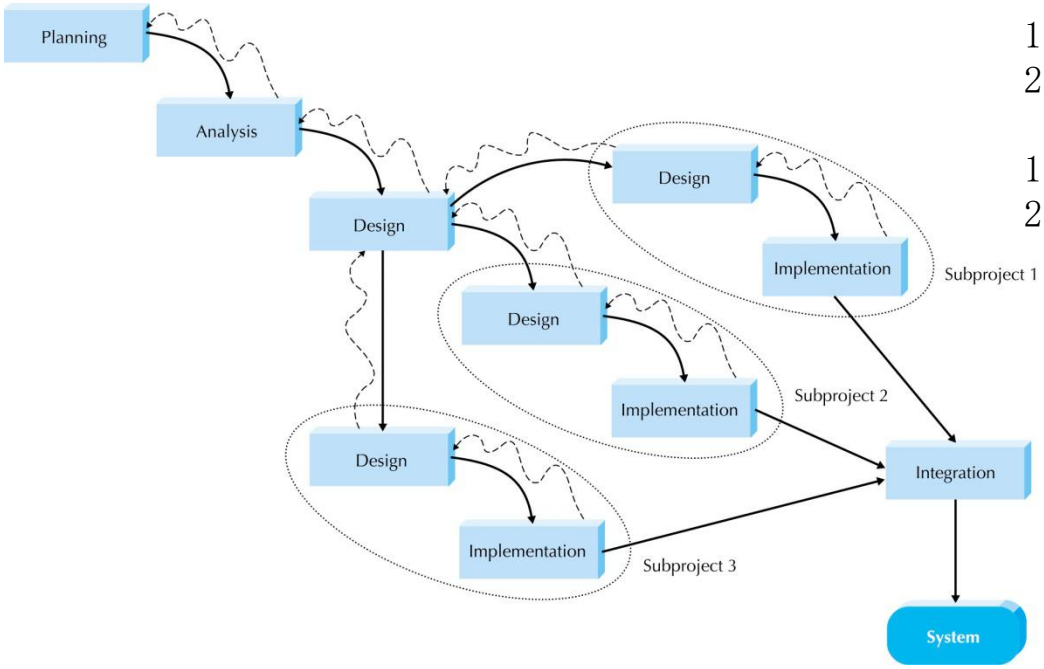
Waterfall Development



1. 會花較長時間找出需求
2. 當專案進行時會限制需求的改變（因為限制回溯）

1. 分析階段到系統程式完成，花太長時間
2. 測試對程式設計階段，好像是添加的（測試不周全）
3. 若分析缺乏了重要需求，則必須高昂花費

Parallel Development

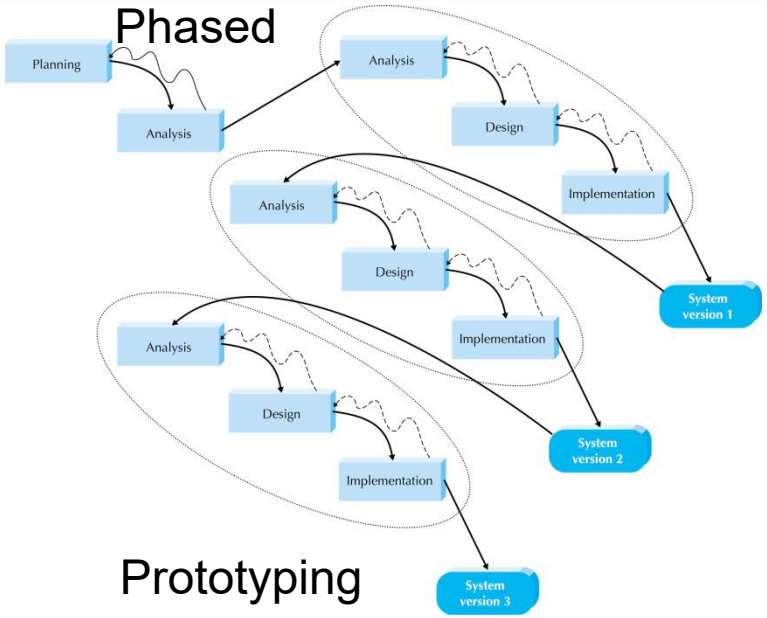


1. 能較快完成系統
2. 改變時比較不需要整個系統重做因為允許回溯

1. 有大量文件產生（做文件要花時間）
2. 當子系統沒有完全獨立時，整合系統時會發生問題

Rapid Application Development

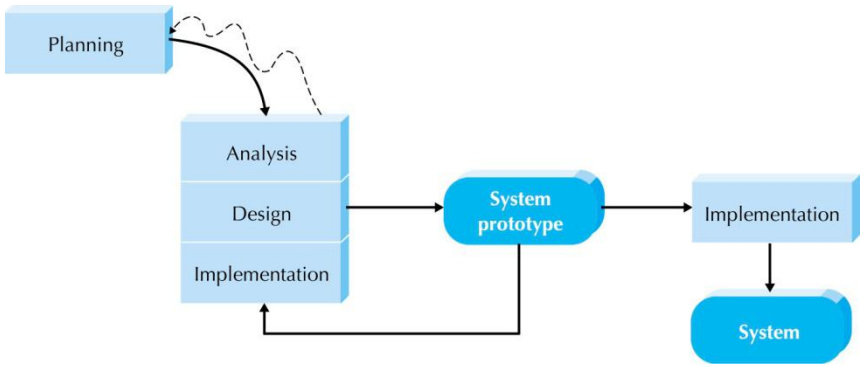
Phased



1. 快速就能夠開發出來

1. 版本1是最重要的，以這做為基礎疊升

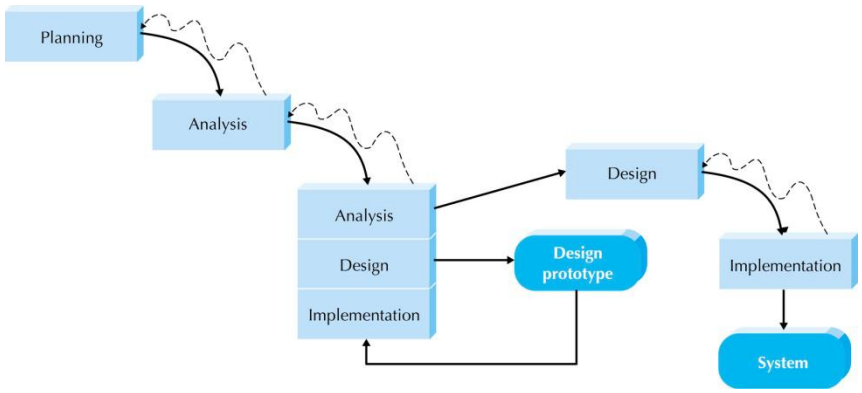
Prototyping



1. 使用者可以即時進行互動，不需要紙上作業
2. 使用者可以放心，系統是根據其意見來做改進的
3. 適合使用在小型專案

1. 缺少系統化的方法來進行分析工作
2. 缺乏系統基礎性的考量（系統架構，安全，控制上的考量）
3. 快速但是並不可靠

Throwaway Prototyping



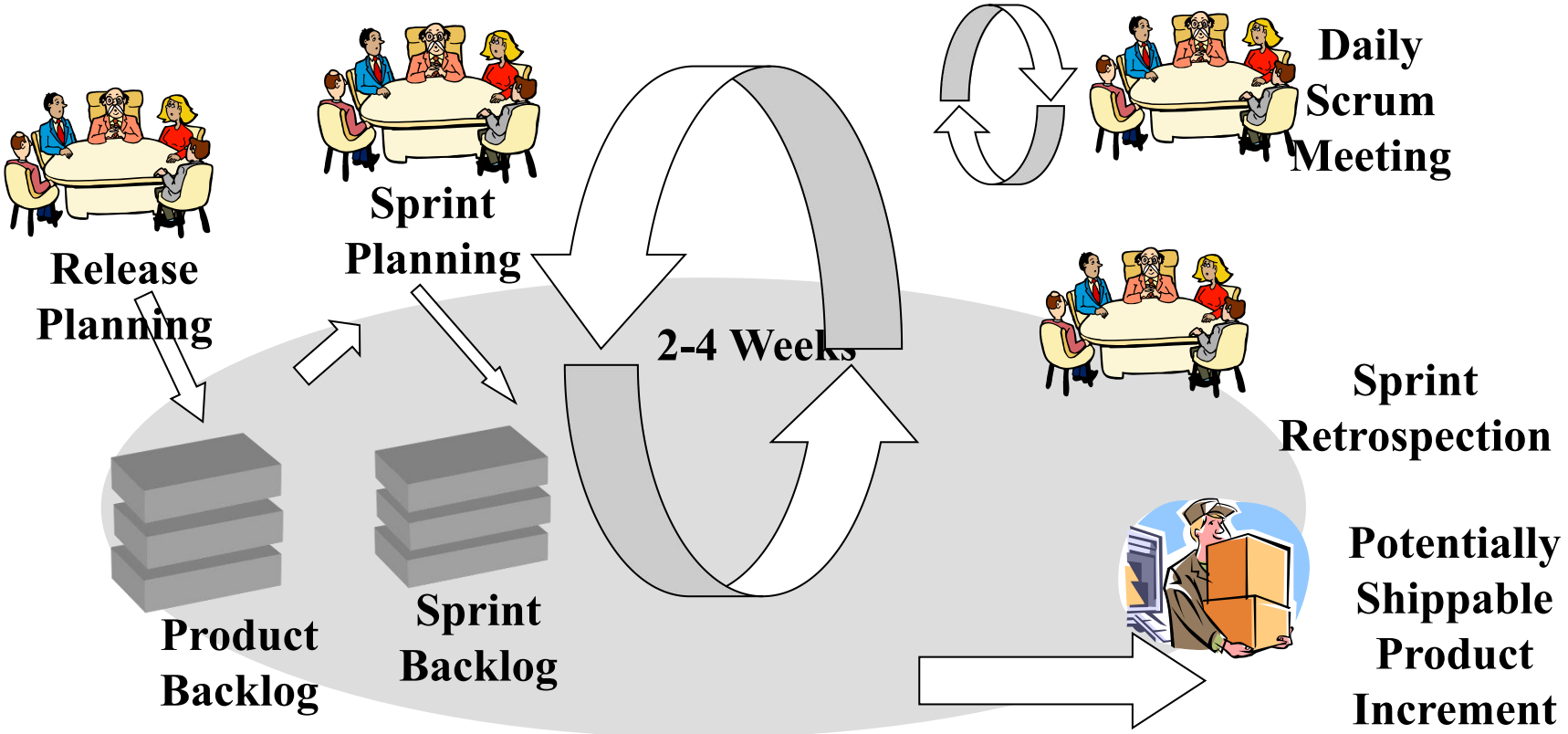
1. 系統建立之前可確定問題，比較可靠謹慎的系統
2. 不忽略基本的基礎架構，安全，控制的需求

1. 比雛形法花更多的時間
   （因為要做整體性的分析與設計，而且要重新撰寫程式）

eXtreme Programming

1. 比 RAD 的方法更快能產生系統
2. 適合用在小專案

1. 在分析與設計階段，使用 XP 方法，產生的文件很少
   因此要不可能維護 XP 所建構的資訊系統
3. 此方法需要使用者在發展系統時一起工作
   (1) 定義需求(2) 客戶測試這一點常常很難遵守



Which Methodology to Use?

Ability to Develop Systems	Structured Methodologies		RAD Methodologies			Agile Methodologies	
	Waterfall	Parallel	Phased	Prototyping	Throwaway Prototyping	XP	SCRUM
With Unclear User Requirements	Poor	Poor	Good	Excellent	Excellent	Excellent	Excellent
With Unfamiliar Technology	Poor	Poor	Good	Poor	Excellent	Good	Good
That Are Complex	Good	Good	Good	Poor	Excellent	Good	Good
That Are Reliable	Good	Good	Good	Poor	Excellent	Excellent	Excellent
With a Short Time Schedule	Poor	Good	Excellent	Excellent	Good	Excellent	Excellent
With Schedule Visibility	Poor	Poor	Excellent	Excellent	Good	Excellent	Excellent

The Unified Process

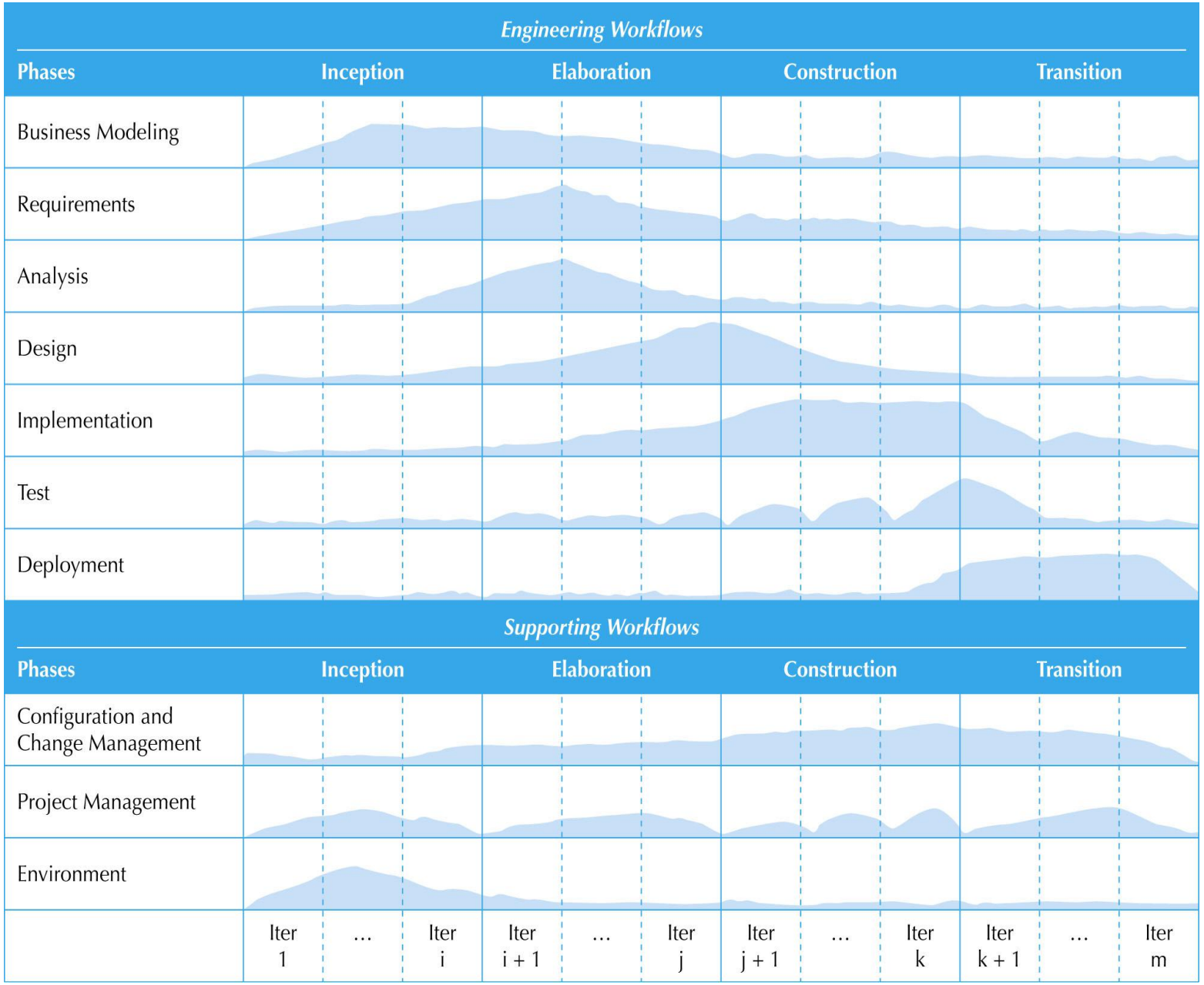
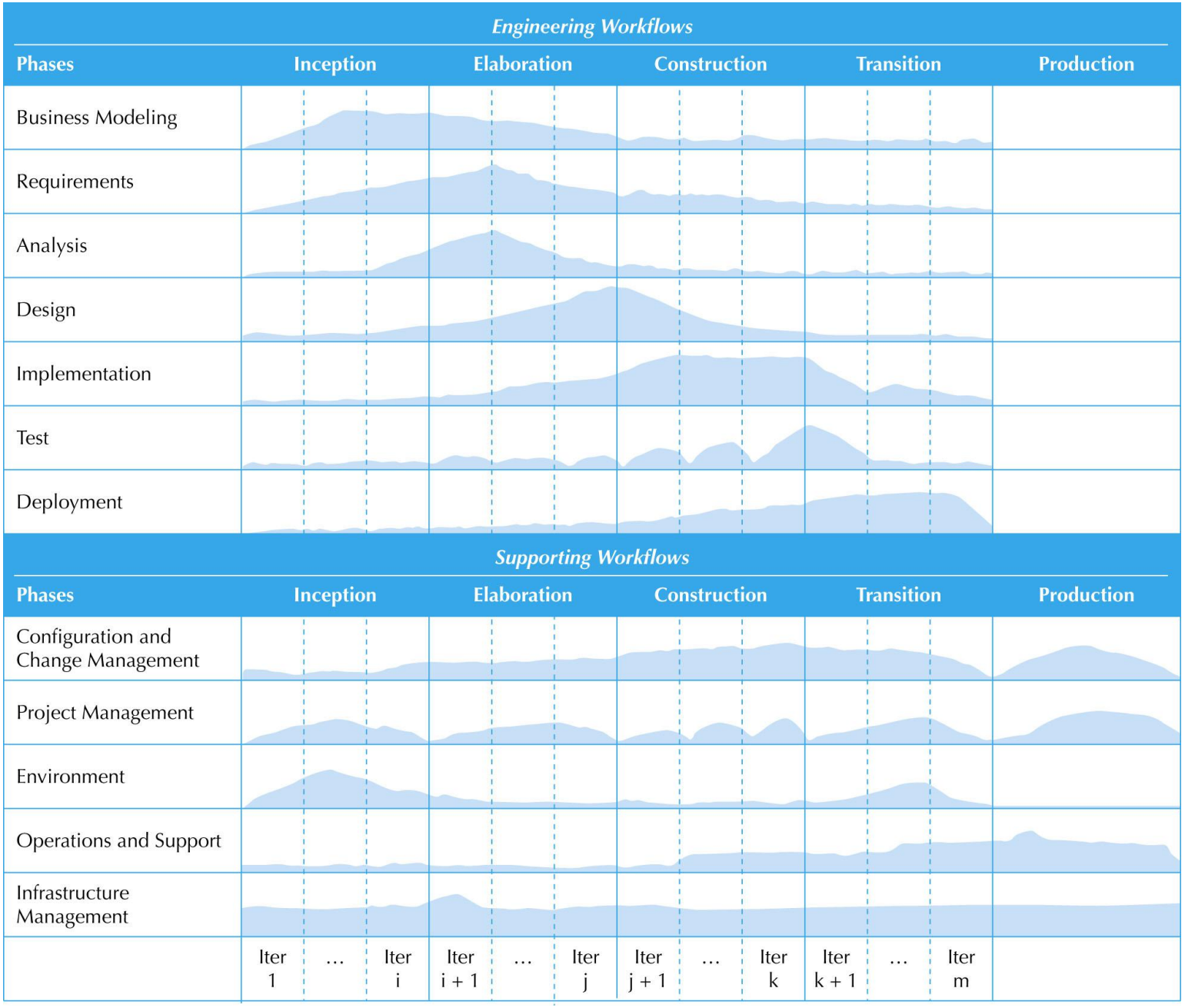


fig_01_10

Extensions to the Unified Process



Characteristics of Object-Oriented Systems

- **類別與物件 Classes & Objects**
  - 物件 Object (instance): instantiation of a class
  - 屬性 Attributes: information that describes the class
  - 狀態 State: describes its values and relationships at a point in time
- **方法與消息 Methods & Messages**
  - 方法 Methods: the behavior of a class
  - 消息 Messages: information sent to an object to trigger a method
- Unified Process的三大精神 Characteristics of OOAD :
  - 使用案例導向 Use-case Driven
  - 架構為中心 Architecture Centric
  - 反覆與漸進 Iterative and Incremental

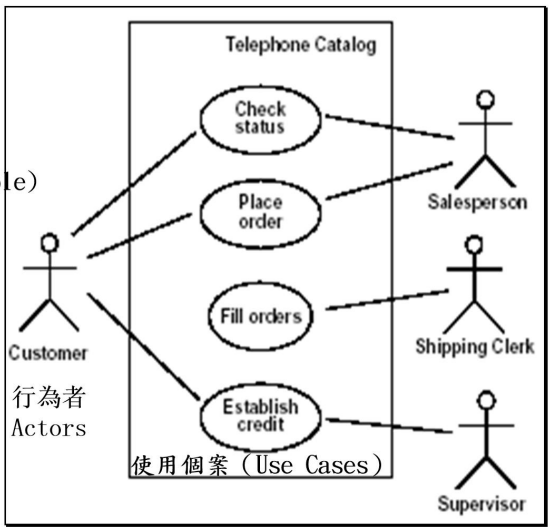
Unified Modeling Language (UML&OOA D)

- UML『只是』一種『畫圖的表示法』而已 且有很多種
  - **1.個案圖Use case Diagram**
  - 2.**類別圖Class Diagram**
  - **3.物件圖Object Diagram**
  - 4.循序圖Sequence Diagram
  - 5.合作圖Collaboration Diagram
  - 6.狀態圖State Diagram
  - 7.活動圖Activity Diagram
  - 8.元件圖Component Diagram
  - 9.佈署圖Deployment Diagram

(1)使用個案圖(Use case diagram)

使用個案圖元件	符 號
行為者	
使用個案	
行為者與使用個案間之互動	
使用個案間之Include與Extend關係	
使用個案間之Generalization關係	

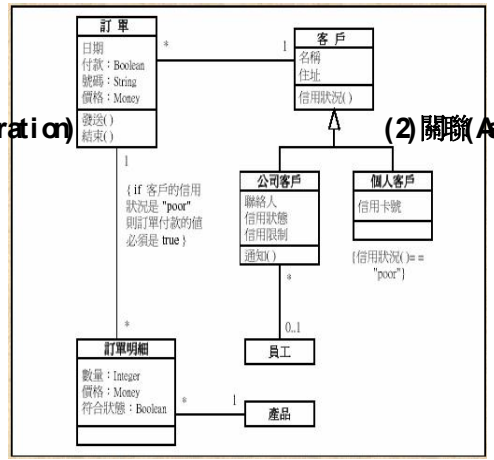
角色(Role)



系統中一系列的交易，以完成某一特定工作，並對系統之行為者產生可衡量的價值。
也就是說，工作結果對行為者產生一些可看得見、可量化或質化的效益。

系統使用個案和行為者之間互動的關係。

(2)類別圖(Class diagram)



(1)操作(Operation)
物件之行為
(Method)

(2)關聯(Association) 表示類別間的連結

(3)基數(Multiplicity)

類別間之關係	符 號
相依關係 (Dependency)	
一般化關係 (Generalization)	
關聯 (Association)	
實現化關係 (Realization)	

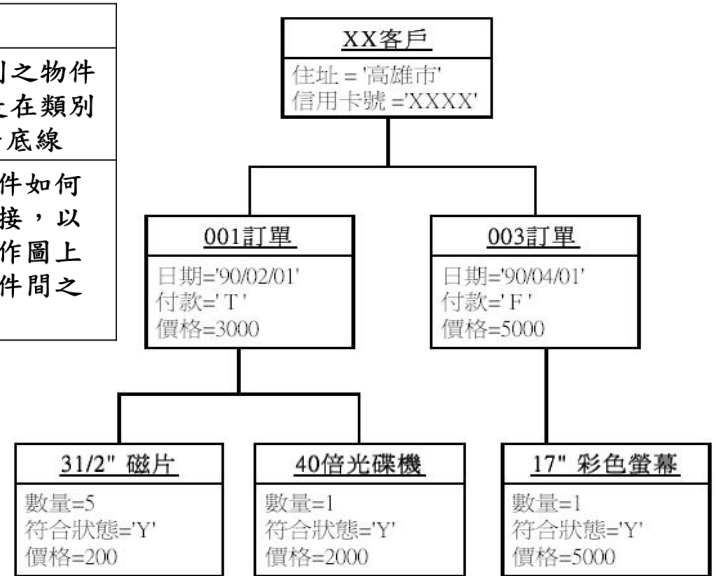
系統存在之類別以及類別間的邏輯關係

(1)在一個關聯關係中，常須表達有多少物件參與此關係，此種資訊與實體關係模式中之基數表達相同。
(2)基數表示有多少個案例參與該關聯，“多少”稱為關聯角色的基數(Multiplicity)
(3)有三種關聯，分別為一對一(1:1)、一對多(1:N)或多對多(M:N)，其中以阿拉伯數字1表示一，以英文字母N或M表示多，或用“*”表示多

(3)物件圖(Object Diagram)

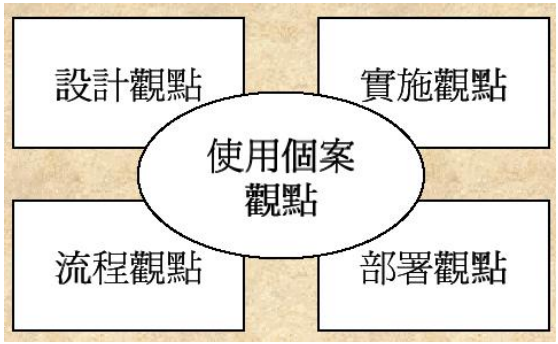
描述一系統於某一時間點的靜態結構

基本元件	所表達資訊
物件	[1]類別圖上類別之物件 [2]表達之方式是在類別之名稱下劃一底線
連結線	用來表示一個物件如何與另一個物件連接，以直線來表示。合作圖上之連結也就是物件間之路徑（Path）



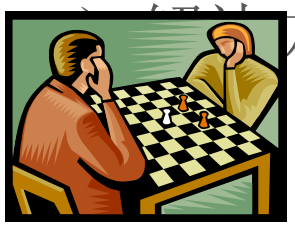
4+1 View model

使用個案觀點：	由描述系統行為的使用個案組成，這些系統行為是由使用者、分析師、測試者的觀點來描述，並不實際描述軟體系統的組織。
設計觀點：	由類別、介面與合作組成，這些是來自於描述問題及其解決方法中之辭彙描述。這個觀點主要支援系統的功能需求，表達系統應提供給使用者之服務。
流程觀點：	由執行緒與流程所組成，這些是來自於系統的平行與同步機制，這個觀點主要表達系統之績效、產出與可擴充性。
實施觀點：	由可以不同方式組裝實際可運作系統之獨立的元件與檔案所組成，這個觀點主要表達系統版本的結構配置管理。
部署觀點：	由構成系統之硬體類型的節點(Nodes)所組成，這個觀點主要表達組成實際系統之零件的分配、傳遞訊息與安裝。



Tame Problem

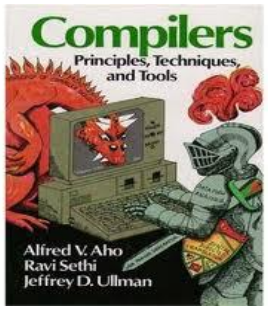
1）tame problem可以完全specified。
2）對於tame problem，specification和solution可以分開。
3）對於tame problem ，有停止規則。
4）可以根據正確或錯誤來評估tame problem的解決方案。
）解決方案可以立即測試：一旦測試，它永遠保持正確。



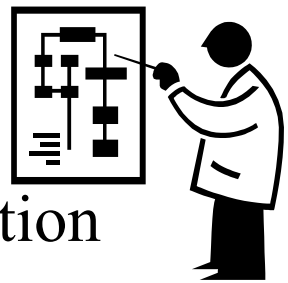
Chess playing

Operating systems

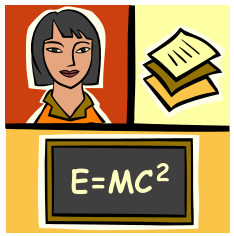
AI problems



Compiler construction



Query optimization



Math problems

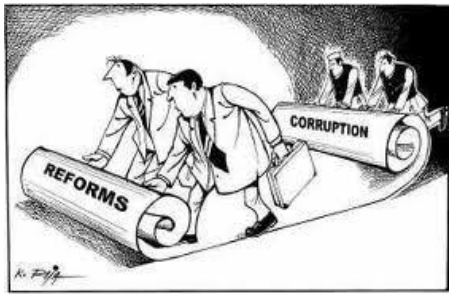
Operations research

Wicked Problem

1）wicked problem沒有明確的解法。
2）specification和solution不能分開。
3）沒有停止規則 - 你總是可以做得更好。
4）解決方案只能根據好壞評估
5）問題解決過程的每一步都有無數的選擇
6）每一個wicked problem都是獨一無二的。

Urban planning

National policy making



Economic reforms

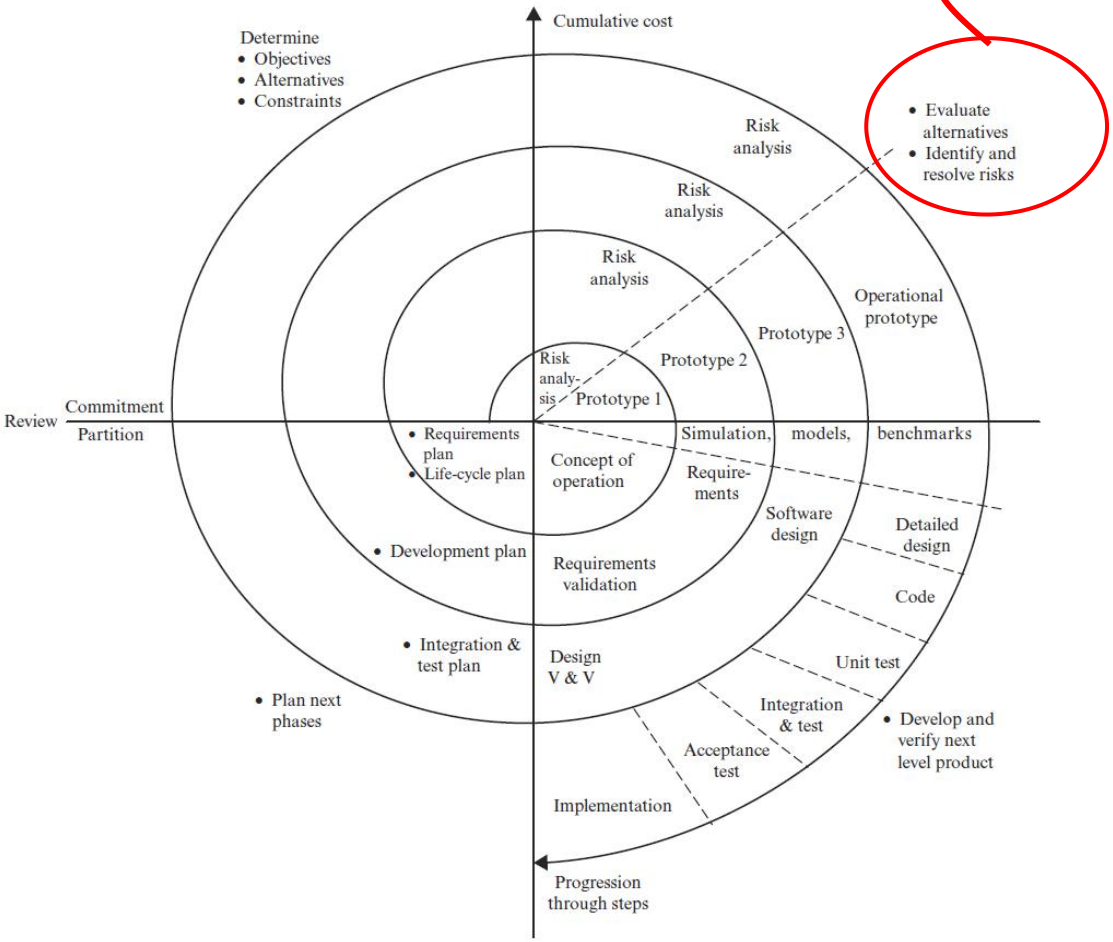
Application software development

Software Process Models

- 原型模型 Prototyping Process Model
- 演化模型 Evolutionary Process Model
- **螺旋模型 Spiral Process Model**
- **統一開發模型Unified Process Model**
- 個人軟體模型 Personal Software Process Model
- **團體軟體模型 Team Software Process Model**
- **敏捷模型 Agile Process Models**

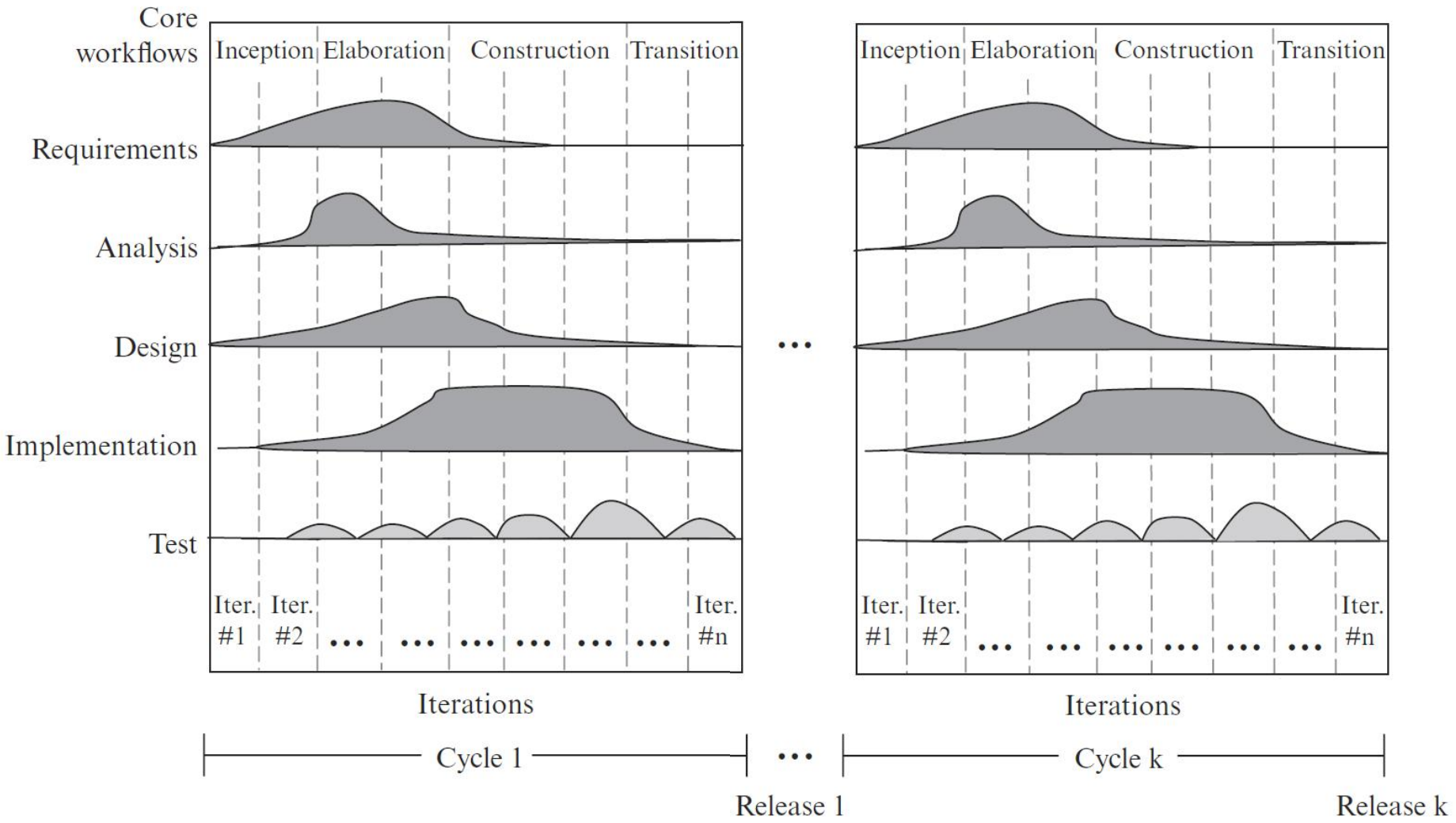
Spiral Process Model

It begins here.

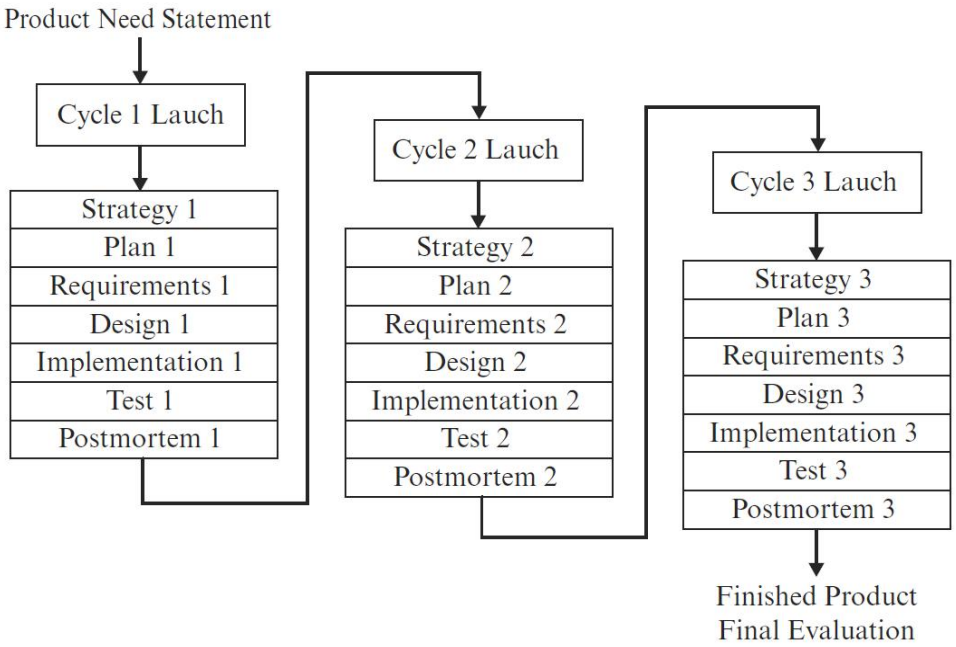


如果風險存在→計畫
If risks remains {
  plan next phase(SW)
  conduct prototyping
}
如果風險解決→瀑布式
else if risks resolved {
  proceed as waterfall(SE)
}
如果雛形繼續→演化模型
else if prototype works&robust {
  proceed as evolutionary model(NE)
}

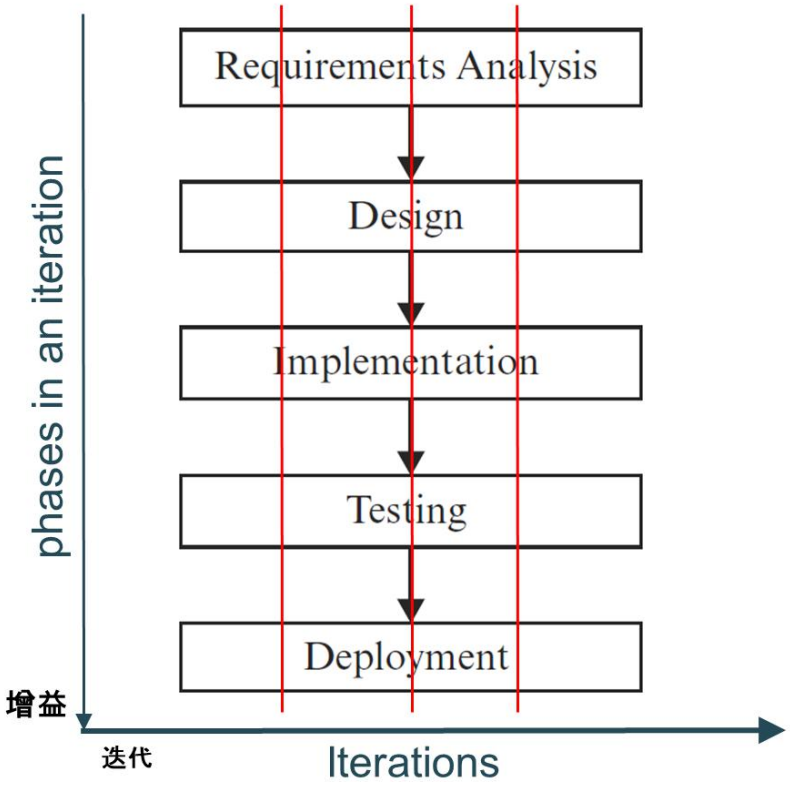
Rational Unified Process (RUP)



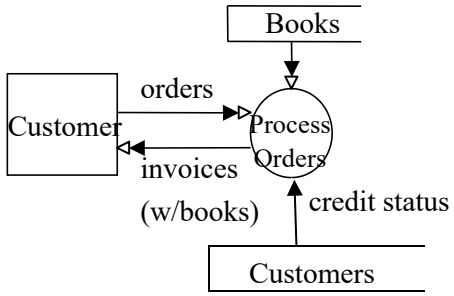
Team Software Process



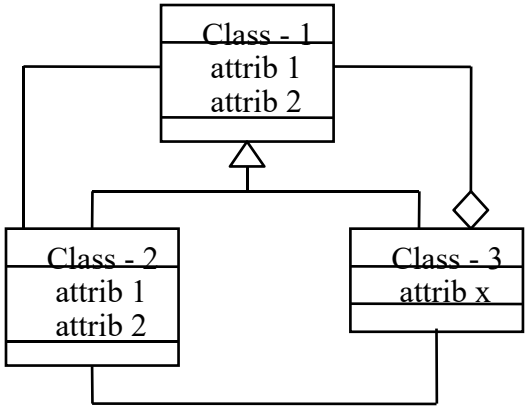
Agile Process Models



Paradigm and Methodology

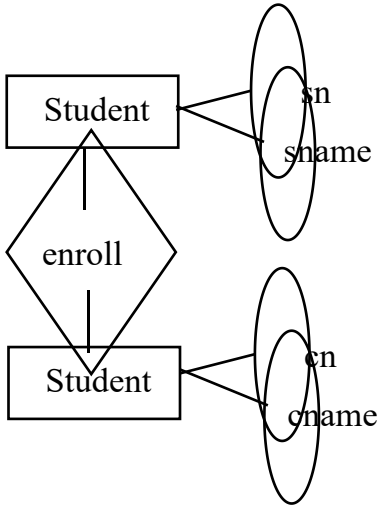


Structured Analysis

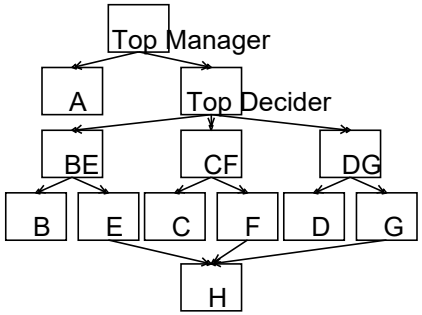


Domain model

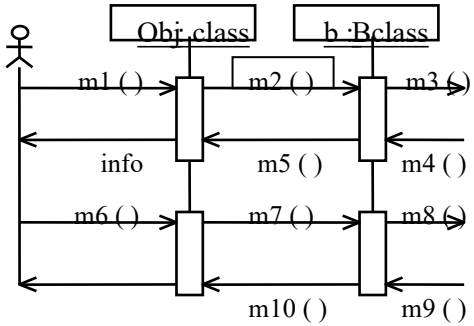
Object-Oriented Analysis



Data-Oriented Analysis

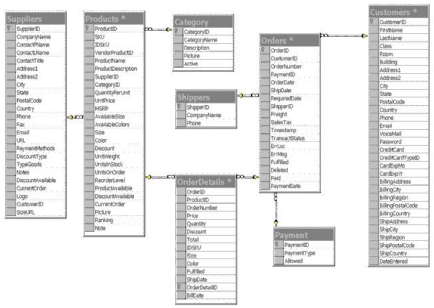


Structured Design



Sequence diagram

Object-Oriented Design



Data-Oriented Design

Procedural Paradigm

OO Paradigm

Data-Oriented Paradigm

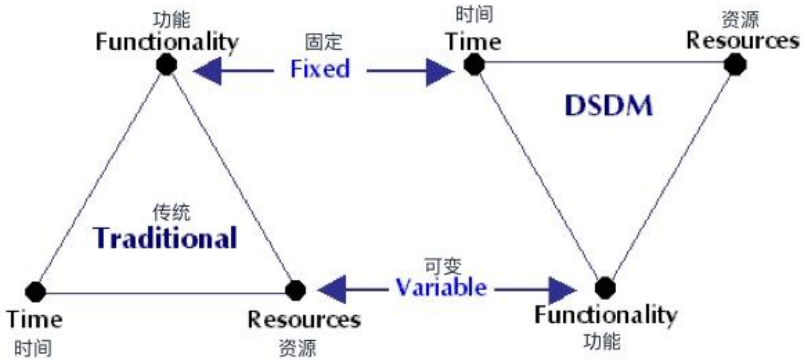
Agile Methods

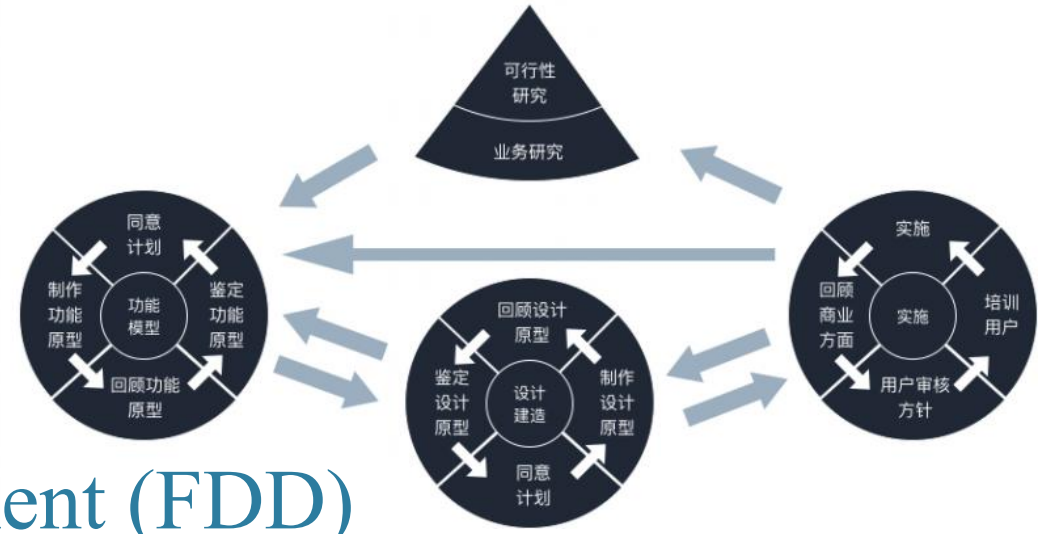
- Dynamic Systems Development Method (DSDM)
- Feature Driven Development (FDD)
- Scrum
- Extreme Programming (XP)
- Crystal Clear
- Lean Development

DSDM Unique Key Features

- based on the 80-20 principle.(20分付出創造80分價值)

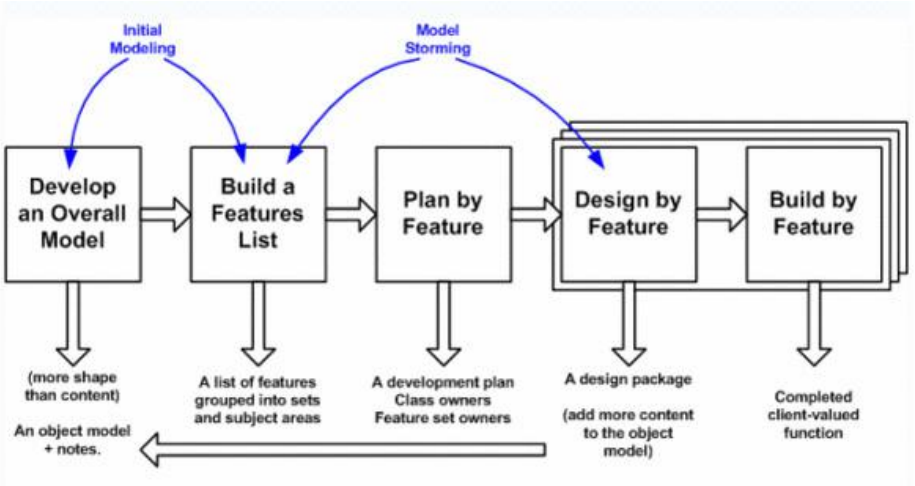
- 在傳統開發中，功能固定，時間人力是可變的，
  而在DSDM中，時間固定，功能資源是可變的(no delay)

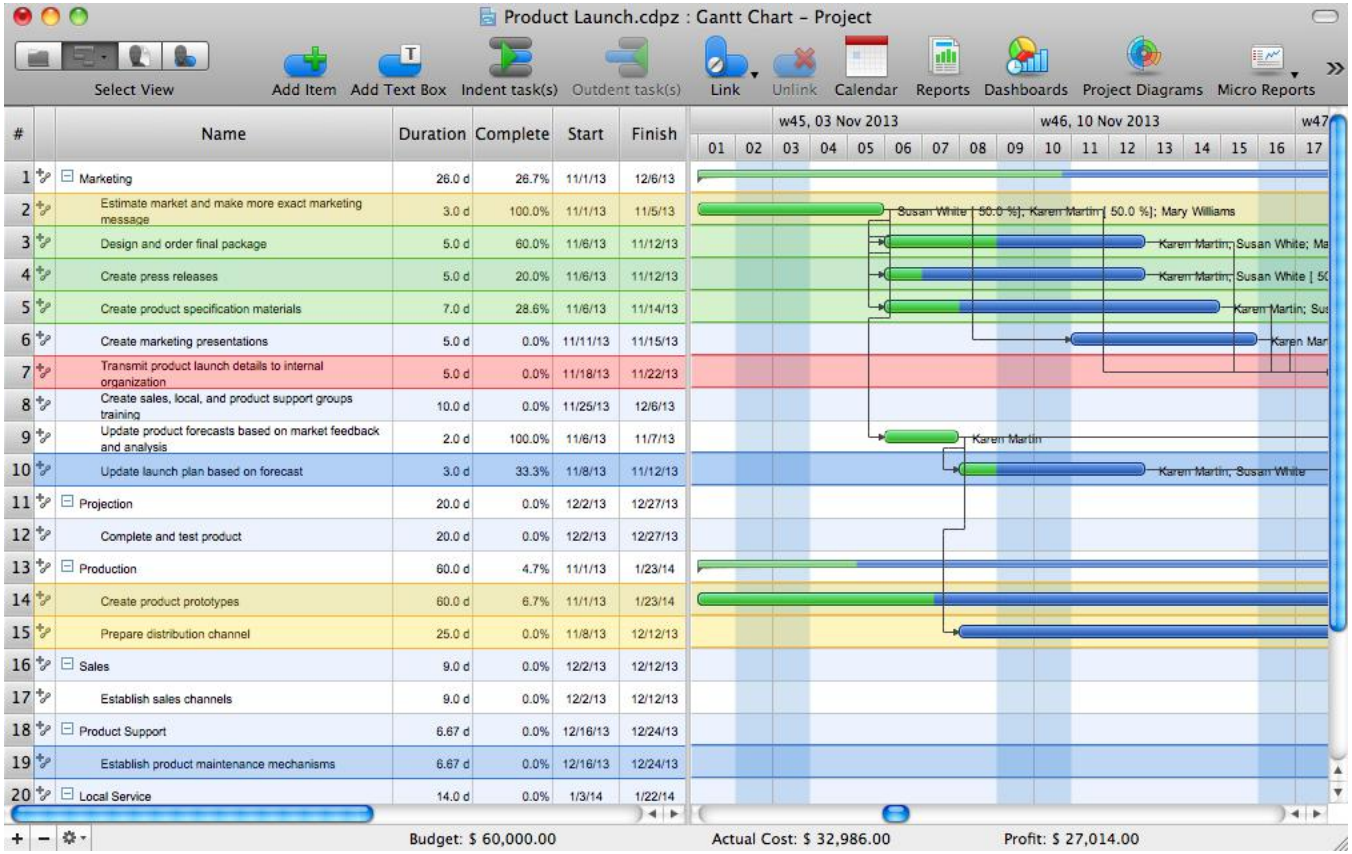




Feature Driven Development (FDD)

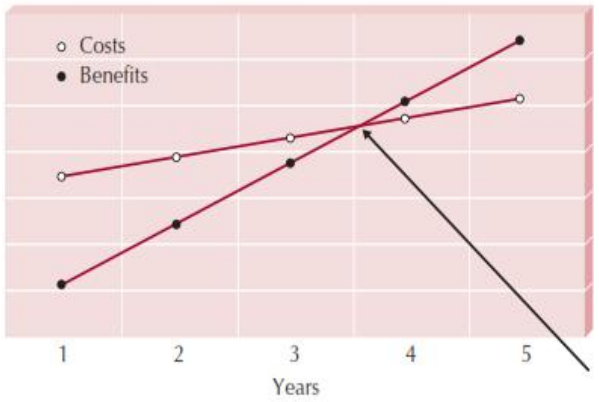
- 它跟 Scrum 和 XP 最大差別的地方,
  就是 FDD 會先對要做的系統,
  「先分析」出一個 model





	Activity	Predecessor Activities	Duration (days)
A	Planning	None	4
B	Database Design	A	4
C	Module Layout	A	2
D	Database Capture	B	5
E	Database Interface	B	2
F	Input Module	C	3
G	Output Module	C	2
H	GUI Structure	C	3
I	I/O Interface Implementation	E, F, G	2
J	Final Testing	D, H, I	2

	2008	2009	2010	2011	2012	Total
Increased sales	500,000	530,000	561,800	595,508	631,238	
Reduction in customer complaint calls	70,000	70,000	70,000	70,000	70,000	
Reduced inventory costs	68,000	68,000	68,000	68,000	68,000	
TOTAL BENEFITS:	638,000	668,000	699,800	733,508	769,238	
PV OF BENEFITS:	619,417	629,654	640,416	651,712	663,552	3,204,752
PV OF ALL BENEFITS:	619,417	1,249,072	1,889,488	2,541,200	3,204,752	
2 Servers @ $125,000	250,000	0	0	0	0	
Printer	100,000	0	0	0	0	
Software licenses	34,825	0	0	0	0	
Server software	10,945	0	0	0	0	
Development labor	1,236,525	0	0	0	0	
TOTAL DEVELOPMENT COSTS:	1,632,295	0	0	0	0	
Hardware	54,000	81,261	81,261	81,261	81,261	
Software	20,000	20,000	20,000	20,000	20,000	
Operational labor	111,788	116,260	120,910	125,746	130,776	
TOTAL OPERATIONAL COSTS:	185,788	217,521	222,171	227,007	232,037	
TOTAL COSTS:	1,818,083	217,521	222,171	227,007	232,037	
PV OF COSTS:	1,765,129	205,034	203,318	201,693	200,157	2,575,331
PV OF ALL COSTS:	1,765,129	1,970,163	2,173,481	2,375,174	2,575,331	
TOTAL PROJECT BENEFITS – COSTS:	(1,180,083)	450,479	477,629	506,501	537,201	
YEARLY NPV:	(1,145,712)	424,620	437,098	450,019	463,395	629,421
CUMULATIVE NPV:	(1,145,712)	(721,091)	(283,993)	166,026	629,421	
RETURN ON INVESTMENT:	24.44%	(629,421/2,575,331)				
BREAK-EVEN POINT:	3.63 years	[break-even occurs in year 4; (450,019 – 166,026)/450,019 = 0.63]				
INTANGIBLE BENEFITS:	This service is currently provided by competitors Improved customer satisfaction					



Present Value (PV)	The amount of an investment today compared to that same amount in the future, taking into account inflation and time.	$\frac{\text{Amount}}{(1 + \text{interest rate})^n}$ n = number of years in future
Net Present Value (NPV)	The present value of benefit less the present value of costs.	PV Benefits – PV Costs
Return on Investment (ROI)	The amount of revenues or cost savings results from a given investment.	$\frac{\text{Total benefits} - \text{Total costs}}{\text{Total costs}}$
Break-Even Point	The point in time at which the costs of the project equal the value it has delivered.	$\frac{\text{Yearly NPV*} - \text{Cumulative NPV}}{\text{Yearly NPV*}}$

	A	B	C	D	E	F	G
1		**2015**	**2016**	**2017**	**2018**	**2019**	**Total**
2							
3	Increased sales	500,000	530,000	561,800	595,508	631,238	
4	Reduction in customer complaint calls	70,000	70,000	70,000	70,000	70,000	
5	Reduced inventory costs	68,000	68,000	68,000	68,000	68,000	
6	**TOTAL BENEFITS:**	638,000	668,000	699,800	733,508	769,238	
7	**PV of BENEFITS:**	**619,417**	**629,654**	**640,416**	**651,712**	**663,552**	**3,204,752**
8	**PV of ALL BENEFITS:**	**619,417**	**1,249,072**	**1,889,488**	**2,541,200**	**3,204,752**	
9							
10	2 Servers @ $125,000	250,000	0	0	0	0	
11	Printer	100,000	0	0	0	0	
12	Software licenses	34,825	0	0	0	0	
13	Server software	10,945	0	0	0	0	
14	Development Labor	1,236,525	0	0	0	0	
15	**TOTAL DEVELOPMENT COSTS:**	1,632,295	0	0	0	0	
16							
17	Hardware	54,000	81,261	81,261	81,261	81,261	
18	Software	20,000	20,000	20,000	20,000	20,000	
19	Operational Labor	111,788	116,260	120,910	125,746	130,776	
20	**TOTAL OPERATIONAL COSTS:**	185,788	217,521	222,171	227,007	232,037	
21							
22	**TOTAL COSTS:**	1,818,083	217,521	222,171	227,007	232,037	
23	**PV of COSTS:**	**1,765,129**	**205,034**	**203,318**	**201,693**	**200,157**	**2,575,331**
24	**PV of ALL COSTS:**	**1,765,129**	**1,970,163**	**2,173,481**	**2,375,174**	**2,575,331**	
25							
26	**Total Project Benefits - Costs:**	**-1,180,083**	**450,479**	**477,629**	**506,501**	**537,201**	
27	**Yearly NPV:**	**-1,145,712**	**424,620**	**437,098**	**450,019**	**463,395**	**629,421**
28	**Cumulative NPV:**	**-1,145,712**	**-721,092**	**-283,993**	**166,026**	**629,421**	
29							
30	**Return on Investment:**	**24.44%**	(629,421 / 2,575,331)				
31							
32	**Breakeven Point:**	**3.631069**	(breakeven occurs in year 4; 450,019 - 166,026 / 450,019 = .63)				
33							
34							
35	**Intangible Benefits:**	This service is currently provided by competitors					
36		Improved customer satisfaction					

Unadjusted Actor Weighting Table:

Actor Type	Description	Weighting Factor	Number	Result
Simple	External system with well-defined API	1	0	0
Average	External system using a protocol-based interface, e.g., HTTP, TCT/IP, or a database	2	0	0
Complex	Human	3	4	12
			Unadjusted Actor Weight Total (UAW)	**12**

Unadjusted Use-Case Weighting Table:

Use Case Type	Description	Weighting Factor	Number	Result
Simple	1–3 transactions	5	3	15
Average	4–7 transactions	10	4	40
Complex	>7 transactions	15	1	15
			Unadjusted Use Case Weight Total (UUCW)	**70**

Unadjusted Use-Case Points (UUCP) = UAW + UUCW **82 = 12 + 70**

Technical Complexity Factors:

Factor Number	Description	Weight	Assigned Value (0 – 5)	Weighted Value	Notes
T1	Distributed system	2.0	0	0	
T2	Response time or throughput performance objectives	1.0	5	5	
T3	End-user online efficiency	1.0	3	3	
T4	Complex internal processing	1.0	1	1	
T5	Reusability of code	1.0	1	1	
T6	Ease of installation	0.5	2	1	
T7	Ease of use	0.5	4	2	
T8	Portability	2.0	0	0	
T9	Ease of change	1.0	2	2	
T10	Concurrency	1.0	0	0	
T11	Special security objectives included	1.0	0	0	
T12	Direct access for third parties	1.0	0	0	
T13	Special user training required	1.0	0	0	
			Technical Factor Value (TFactor)	**15**	

Technical Complexity Factor (TCF) = 0.6 + (0.01 * TFactor) **0.75 = 0.6 + (0.01 * 15)**

Environmental Factors:

Factor Number	Description	Weight	Assigned Value (0 – 5)	Weighted Value	Notes
E1	Familiarity with system development process being used	1.5	4	6	
E2	Application experience	0.5	4	2	
E3	Object-oriented experience	1.0	4	4	
E4	Lead analyst capability	0.5	5	2.5	
E5	Motivation	1.0	5	5	
E6	Requirements stability	2.0	5	10	
E7	Part-time staff	–1.0	0	0	
E8	Difficulty of programming language	–1.0	4	–4.0	
			Environmental Factor Value (EFactor)	**25.5**	

Environmental Factor (EF) = 1.4 + (−0.03 * EFactor) **0.635 = 1.4 + (−0.03 * 25.5)**
Adjusted Use Case Points (UCP) = UUCP * TCF * ECF **33.3375 = 70 * 0.75 * 0.635**
Effort in person-hours = UCP * PHM **666.75 = 20 * 33.3375**

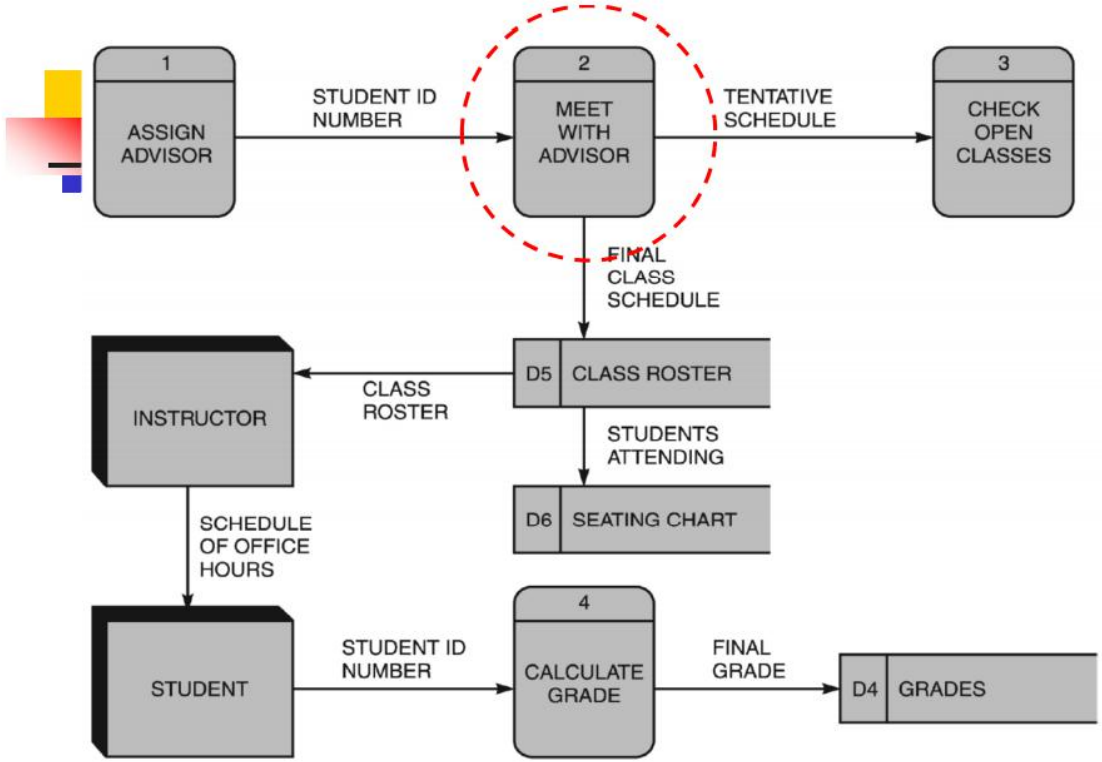
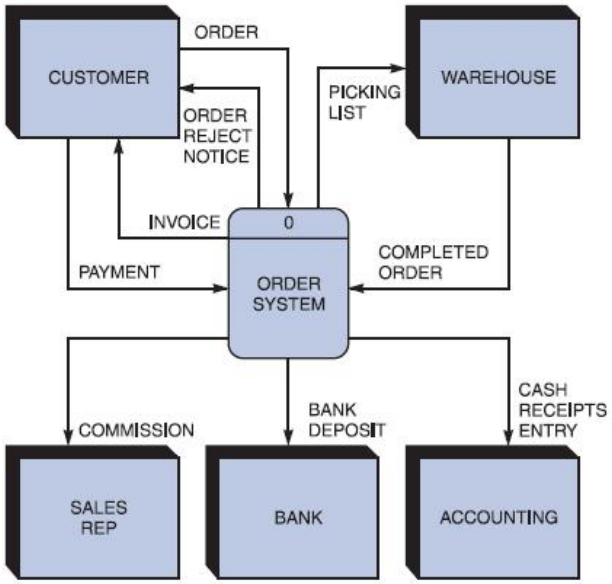
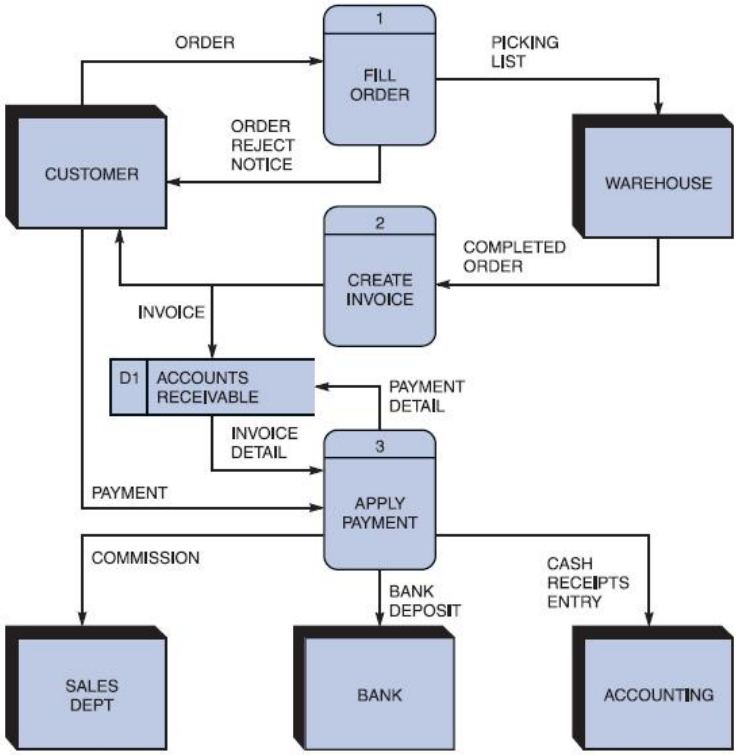


圖4-22 在這個DFD圖0中，有哪些地方是錯誤的?

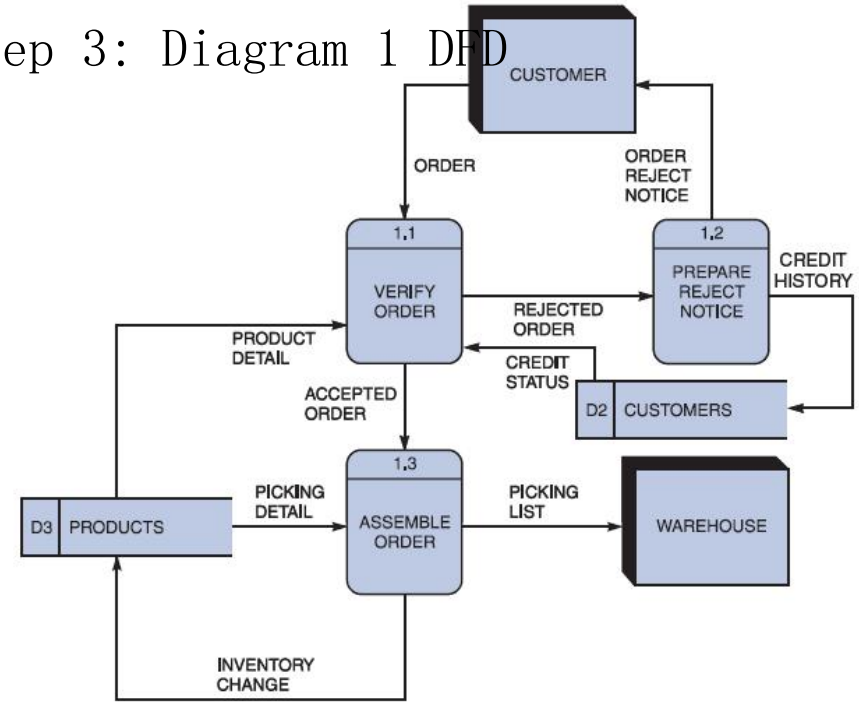
Step 1: Context Diagram

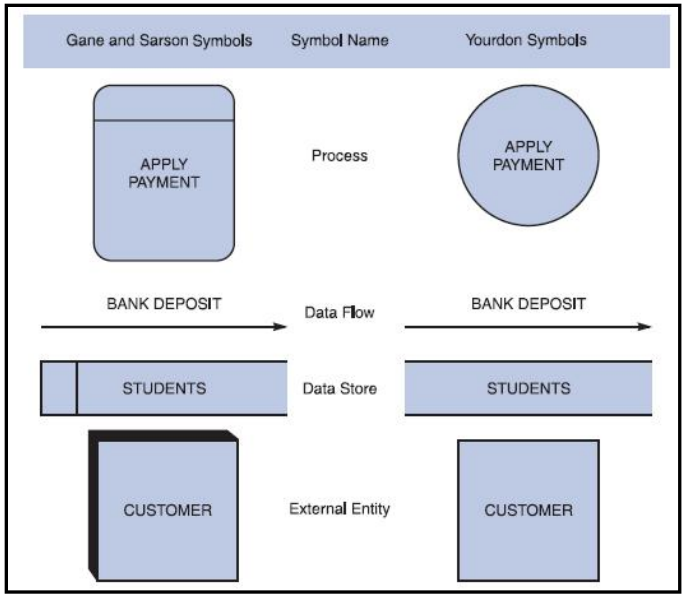


Step 2: Diagram 0 DFD



Step 3: Diagram 1 DFD





- Step 1: Context Diagram

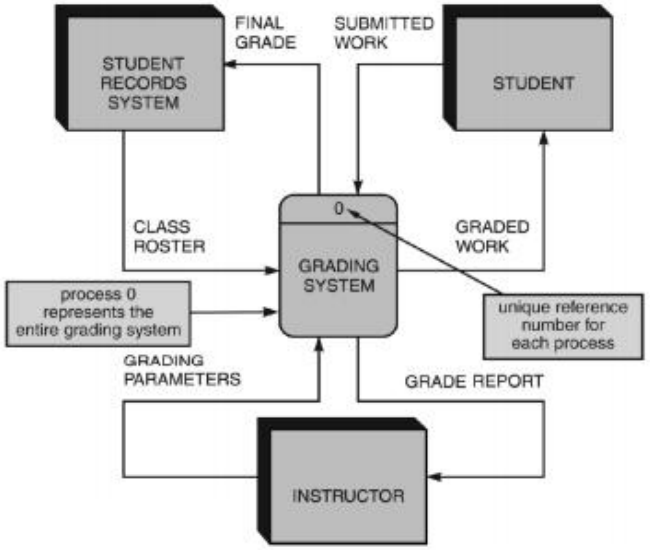


圖4-12 評分系統的環境圖(3種符號)。

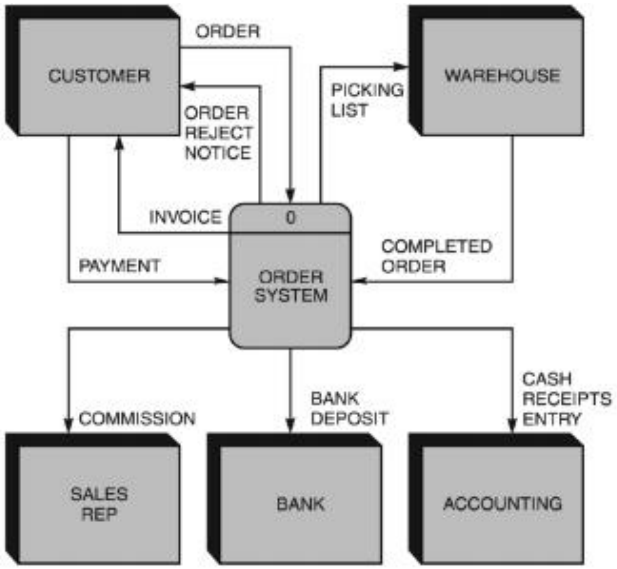
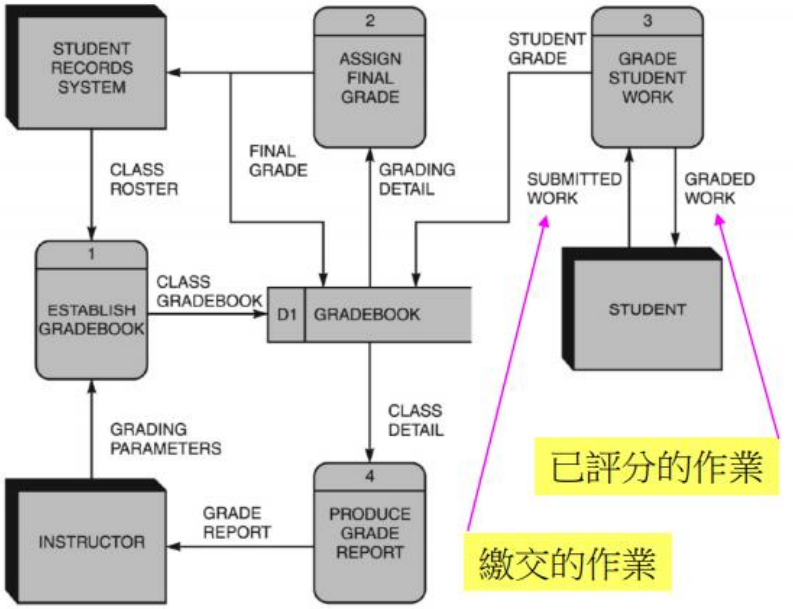


圖4-13 訂單系統的環境圖。

- Step 2: Diagram 0 DFD



評分系統的DFD圖0

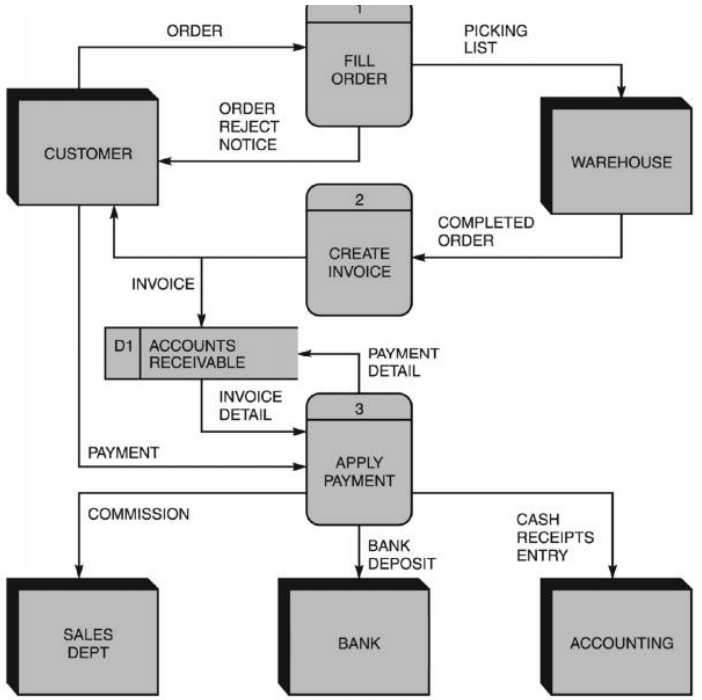


圖4-16 訂單系統的DFD圖0。

Process Description Tools (Cont.)

VERIFY ORDER Business Process with Two Conditions
• An order will be accepted only if the product is in stock and the customer's credit status is OK.
• All other orders will be rejected.

The Verify Order business process has two conditions. For an order to be accepted,
the product must be in stock and the customer must have an acceptable credit status
為了接受訂單，產品必須存貨且客戶必須具有可接受的信用狀態

VERIFY ORDER Process

	1	2	3	4	
Credit status is OK	Y	Y	N	N	3
Product is in stock	Y	N	Y	N	
Accept order	X				4
Reject order		X	X	X	

VERIFY ORDER Business Process with Three Conditions
• An order will be accepted only if the product is in stock and the customer's credit status is OK.
• The credit manager can waive the credit status requirement.
• All other orders will be rejected.

However, the credit manager now has the authority to waive the credit status requirement
然而，信貸經理現在有權放棄信貸狀況要求

VERIFY ORDER Process with Credit Waiver (initial version)

	1	2	3	4	5	6	7	8
Credit status is OK	Y	Y	Y	Y	N	N	N	N
Product is in stock	Y	Y	N	N	Y	Y	N	N
Waiver from credit manager	Y	N	Y	N	Y	N	Y	N
Accept order	X	X			X			
Reject order			X	X		X	X	X

Process Description Tools (Cont.)

在最終版本中，只剩下四條規則。
這些規則記錄了邏輯，並在系統開發時轉成程序

VERIFY ORDER Process with Credit Waiver (with rules marked for combination)

	1	2	3	4	5	6	7	8
Credit status is OK	Y	Y	-	-	N	N	-	-
Product is in stock	Y	Y	N	N	Y	Y	N	N
Waiver from credit manager	-	-	-	-	Y	N	-	-
Accept order	X	X			X			
Reject order			X	X		X	X	X

(1) (2)

VERIFY ORDER Process with Credit Waiver (after rule combination and simplification)

	1 (COMBINES PREVIOUS 1,2)	2 (PREVIOUS 5)	3 (PREVIOUS 6)	4 (COMBINES PREVIOUS 3,4,7,8)
Credit status is OK	Y	N	N	-
Product is in stock	Y	Y	Y	N
Waiver from credit manager	-	Y	N	-
Accept order	X	X		
Reject order			X	X

第一個聲明包含兩個單獨的條件 –
一個用於5%的折扣，另一個用於額外的折扣

SALES PROMOTION POLICY – Holiday Season, 2014

- Preferred customers who order $1,000 or more are entitled to a 5% discount, and an additional 5% discount if they use our charge card.
- Preferred customers who do not order $1,000 or more will receive a $25 bonus coupon.
- All other customers will receive a $5 bonus coupon.

Sales Promotion Policy (initial version)

	1	2	3	4	5	6	7	8
Preferred customer	Y	Y	Y	Y	N	N	N	N
Ordered $1,000 or more	Y	Y	N	N	Y	Y	N	N
Used our charge card	Y	N	Y	N	Y	N	Y	N
5% discount	X	X						
Additional 5% discount	X							
$25 bonus coupon			X	X				
$5 bonus coupon					X	X	X	X

Sales Promotion Policy (final version)

	1	2	3	4	5	6	7	8
Preferred customer	Y	Y	Y	Y	N	N	N	N
Ordered $1,000 or more	Y	Y	N	N	-	-	-	-
Used our charge card	Y	N	-	-	-	-	-	-
5% discount	X	X						
Additional 5% discount	X							
$25 bonus coupon			X	X				
$5 bonus coupon					X	X	X	X

Decision Trees