

Estimating Methods

01418321 System Analysis and Design
Chalothon Chootong (Ph.D.)

Department of Computer Science and Information, Faculty of
Science at Sriracha, Kasetsart University Sriracha Campus

chootong.c@ku.th

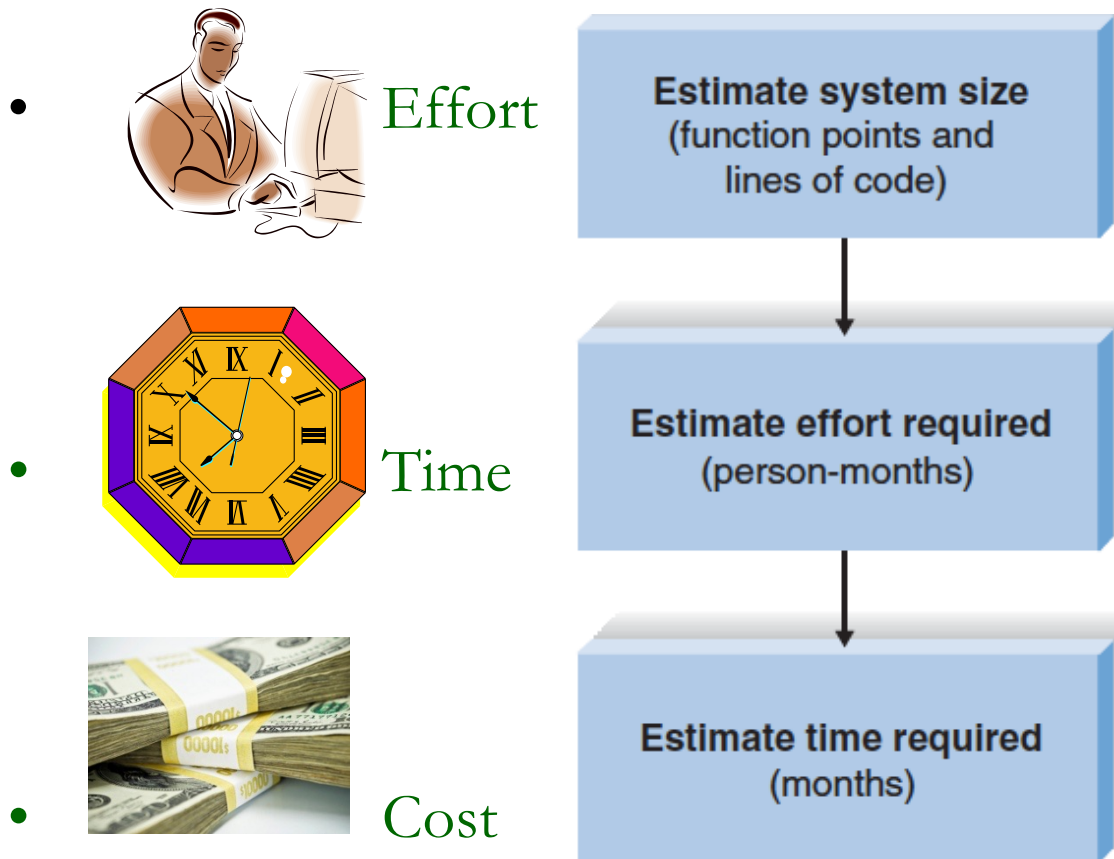


Objective

- ▷ Explain how projects are selected in some organizations
- ▷ Estimating the Project Time Frame
- ▷ Become familiar with project estimation.

Software Estimation

- ▷ Predicting the resources required for a software development process (Size, People, Period, Price)



ขั้นตอนในการประมาณการ

- การเลือกกรรมวิธีการวัด ลักษณะของซอฟต์แวร์(Measurement Method)
- เลือกวิธีที่ใช้ในการประมาณการ(Estimation Techniques)
- ทำการประมาณการในส่วนของคุณค่าใช้จ่ายบุคลากรระยะเวลาและทรัพยากรที่จำเป็น
- (Product Estimation)
- ทำการวิเคราะห์ความเสี่ยง หรือ ผลกระทบจากปัจจัยต่างๆ (Risk Assessment /Impact Analysis)
- ทำการตรวจทานตรวจสอบและปรับแก้ให้เหมาะสม (Review/Revise/Refine/Document)
- ทำการติดตามวัดผลผลิตภัณฑ์หรือซอฟต์แวร์ที่ได้ทำการพัฒนาจริง (Tracking/Product Measurement)
- ทำการปรับปรุง แก้ไขกระบวนการ(Process Improvement)

การประมาณการขนาดของซอฟต์แวร์

- ▷ สิ่งแรกก่อนที่จะทำการประมาณการซอฟต์แวร์ใด ๆ จะต้องทำการวัดขนาดของซอฟต์แวร์เสียก่อน โดยสามารถแบ่งลักษณะของการวัดออกเป็น 2 เชิงด้วยกันคือ
 - การวัดเชิงปริมาณ (Software Quantitative)
วัดในเชิงขนาดบุคลากร ระยะเวลา อัตราความผิดพลาดในการ พัฒนา
ค่าใช้จ่าย
 - การวัดเชิงคุณภาพ (Software Qualitative)
วัดในเชิงลักษณะของการผลิตทีมงาน ความซับซ้อน การ บริหารจัดการ
และ เทคโนโลยีที่นำมาใช้

Popular Estimating Methods

Wideband-Delphi method

A light blue rounded rectangular box containing the text 'Wideband-Delphi method'. A large, light blue downward-pointing arrow is positioned to the right of the box, pointing towards the box below.

Fuzzy-Logic Method

A light blue rounded rectangular box containing the text 'Fuzzy-Logic Method'. A large, light blue downward-pointing arrow is positioned to the right of the box, pointing towards the box below.

Standard-Component Method

Wideband-Delphi method

- ▷ A group of experts is each given the program's specifications and an estimation form
- ▷ They meet to discuss project goals, assumptions and estimation issues
- ▷ Given to the estimate moderator
- ▷ The experts meet to discuss the result
- ▷ Estimation
- ▷ The cycle continues at 3 times

เทคนิคเดลฟายเป็นเทคนิคที่มุ่งแสวงหาข้อมูลจากความคิดเห็นของกลุ่มผู้เชี่ยวชาญในเรื่องใดเรื่องหนึ่ง ด้วยการตอบแบบสอบถาม ดังนั้นผู้เชี่ยวชาญจึงจำเป็นต้องตอบแบบสอบถามที่ผู้วิจัยได้กำหนดขึ้นในแต่ละขั้นตอนการตอบหรือการตัดสินใจของผู้เชี่ยวชาญจะมีความถูกต้องและความตรงสูง เมื่อผู้เชี่ยวชาญนั้นเป็นผู้ที่มีความรู้และมีความเชี่ยวชาญในเรื่องที่ศึกษา

Example

▷ Round 1

○ E1 : estimate 24

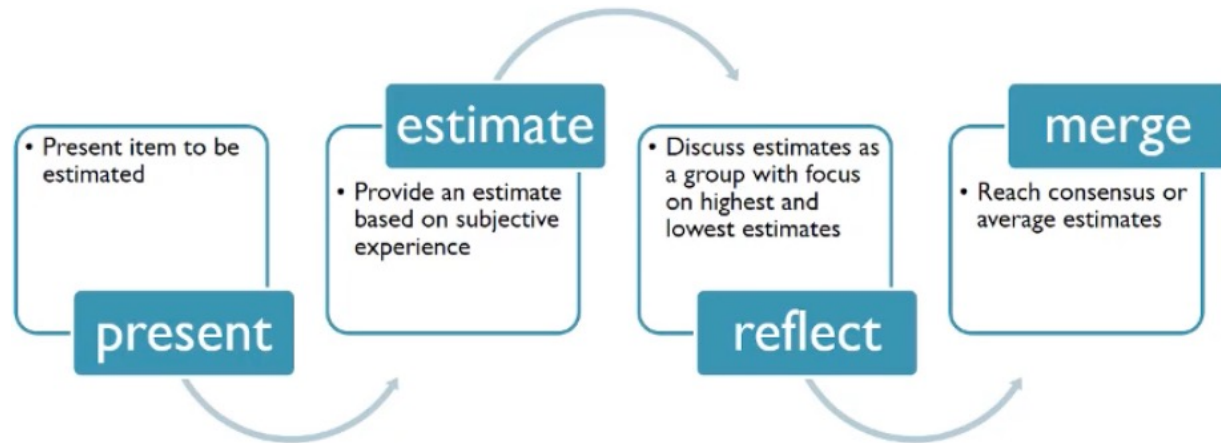
○ E2 : estimate 30

○ E3 : estimate 20

○ E4 : estimate 12

○ 12 20 24 30

$$(20+24)/2 = 22$$



Single-Band Delphi



How many hours will the information system cut-over take?



20



13



8

Wide-Band Delphi

- ▶ For each task, each estimator provide the
 - #Optimistic Effort: the least cost of least time
 - #Pessimistic Effort: the most cost or time
 - #Expected Effort: the realistic forecast
- ▶ Calculate the weighted average of these estimates from each estimator.

$$E = \frac{O + P + 4B}{6}$$

Confidence Interval

- ▷ The estimate should be provided as a range
- ▷ The 95% Confidence Interval is:

$$E \pm 1.96 * SE$$

where

$$SE = \frac{|\bar{O} - \bar{P}|}{6}$$

SE = Standard Error

1.96 is Standard Error for 95% Confidence

Example: Wide-Band Delphi

The following estimates were provided in the second round by four estimators for next year's data use:

Estimator	P	O	B	E
Deepen	87	62	70	72
Alan	91	72	80	81
Melanie	88	50	70	70
Tonya	97	88	95	94

1. What is the weighted estimate for each estimator?
2. What is the average joint estimate?
3. What is the 95% confidence interval for the estimate?

	A	B	C	D	E
Estimator	P	O	B	E	
Deepen	87	62	70	72	$= (B2+C2+ (4*D2)) / 6$
Alan	91	72	80	81	
Melanie	88	50	70	70	
Tonya	97	88	95	94	
Average Estimate			79		$=AVERAGE (E2:E5)$
Standard Error			3.79		
Confidence Interval		95%	72		$= (ABS (AVERAGE (B2:B5) - AVERAGE (C2:C5))) / 6$
	z-score =	1.96	86		$=D7- (C10*D8)$
$=NORM.INV (C9+ ((1-C9) / 2), 0, 1)$					

Advantages

- ▷ Many expert
- ▷ Use in every size product

Disadvantages

- ▷ Use many time
- ▷ Value depend on each person
- ▷ Have many expert

Fuzzy-Logic Method

- ▷ Estimate size with **historical data**
- ▷ Separate developed products into size categories

Fuzzy-Logic Method

Range	Size-LOC
Very Small	2000
Small	8000
Medium	32000
Large	128000
Very Large	512000

LOC = Line Of Code

Advantages

- ▷ Refer to historical data
- ▷ Use to easy
- ▷ Don't have tool
- ▷ No train
- ▷ Good estimate because use historical data

Disadvantages

- ▷ Must have many historical data
- ▷ If new project unlike old project so incorrect
- ▷ Cannot use new project type
- ▷ Cannot in case New project bigger or smaller historical data

Standard-Component Method

Benefit
size of project
effort (person-months)
time (months)

- ▷ Estimate system size (**function points and lines of code**)
- ▷ Estimate effort required (**person-months**)
- ▷ Estimate time required (**months**)

A function point is a measure of program size that is based on the system's number and complexity of inputs, outputs, queries, files, and program interfaces.

Function Point (FP)

- ▷ กระบวนการนับฟังก์ชันพอยต์ มีลักษณะดังนี้
- ▷ ขั้นที่ 1 นำ Requirement ที่เก็บรวบรวมไว้มาทำการแบ่งฟังก์ชันพอยต์
- ▷ ขั้นที่ 2 ประเมินความซับซ้อนของฟังก์ชัน
- ▷ ขั้นที่ 3 เปรียบเทียบความซับซ้อน เพื่อให้ได้ระดับความซับซ้อน เพื่อคำนวณฟังก์ชันพอยต์ที่ยังไม่ได้ปรับค่า (Unadjusted Function Point : UFP)
- ▷ ขั้นที่ 4 คำนวณค่าตัวแปรปรับค่า (Value Adjustment Factor) ตามลักษณะของโครงการ
- ▷ ขั้นที่ 5 คำนวณจำนวนฟังก์ชันพอยต์ที่ผ่านการปรับค่า (Adjusted Function Point : AFP)
- ▷ ขั้นที่ 6 ฟังก์ชันพอยต์ ที่ผ่านการปรับค่าสามารถนำไปคำนวณเป็น LOC ได้

Function Points

By Allen Albrecht of IBM (1979)

Description	Complexity			Total
	Low	Medium	High	
Inputs	...*3	...*4	...*6	...
Outputs	...*4	...*5	...*7	...
Inquiries	...*3	...*4	...*6	...
Files	...*7	...*10	...*15	...
Interface	...*5	...*7	...*10	...
Total unadjusted function points (TUFP)			

Function points (Cont.)

Inputs (EI: External Input)

- Screen or form through which human users of an application
- Other program add new data or update existing data
- If an input screen is too large for a single normal display (usually 80 columns by 25 lines) and flows over onto a second screen

Function points (Cont.)

Outputs (EO: External Output)

- Screen or report for human users
- Separate processing are the unit to count

Function points (Cont.)

Inquiries (EQ: External Query)

- Screens that allow users to interrogate an application
- Ask for assistance or information
- Such as help screens

Function points (Cont.)

Files (ILF: Internal Logical Files)

- Logical collections of records that the application modifies or update
- One table within a relational database
- One path through a network database

Function points (Cont.)

Interface(EIF: External Interface File)

- Files shared with other applications
- Incoming or outgoing tape files
- Shared database

Project Complexity

▷ ค่าที่ประเมินตั้งแต่ 0 (ไม่เกี่ยวข้อง) ถึง 5 (เกี่ยวข้องมาก)

	คุณลักษณะ	ค่า		คุณลักษณะ	ค่า
1	การติดต่อสื่อสารข้อมูล (Data Communication)		8	การปรับปรุงข้อมูลแบบออนไลน์ (Online Update)	
2	การประมวลผลข้อมูลแบบกระจาย (Distribution Data Processing)		9	ความซับซ้อนของการประมวลผล (Complex Processing)	
3	ประสิทธิภาพของระบบ (Performance)		10	การนำไปใช้ซ้ำได้ (Reusability)	
4	การแก้ไขค่าของระบบ (Configuration)		11	ความง่ายในการติดตั้ง (Installation Ease)	
5	ปริมาณรายการข้อมูล (Transaction)		12	ความง่ายในการดำเนินงาน (Operational Ease)	
6	การป้อนข้อมูลเข้าสู่ระบบแบบออนไลน์ (Online Data Entry)		13	การใช้งานได้หลายไซต์ (Multiple Sites)	
7	ประสิทธิภาพการใช้งานของผู้ใช้ (End-user Efficiency)		14	รองรับการเปลี่ยนแปลงความต้องการของผู้ใช้ (Change Requirement)	

Project complexity (PC):

Data Communication

- ▷ The data and control information used in the application are sent or received over communication facilities.

Heavy use configuration

- ▷ A heavily used operational configuration, requiring special design considerations, is a characteristic of the application.

Transaction rate

- ▷ The transaction rate is high and influences the design, development, installation and support.

End-user efficiency

- ▷ The on-line functions provided emphasize a design for end-user efficiency.

Complex processing

- ▷ Complex processing is a characteristic of the application.

Installation ease

- ▷ Conversion and installation ease are characteristics of the application. A conversion and installation plan and/or conversion tools were provided and tested during the system test phase.

Multiple sites

- ▷ The application has been specifically designed, developed and supported to be installed at multiple sites for multiple organizations.

Performance

- ▷ Application performance objectives, stated or approved by the user, in either response or throughput, influence (or will influence) the design, development, installation and support of the application.

Distributed functions

- ▷ Distributed data or processing functions are a characteristic of the application within the application boundary.

On-line data entry

- ▷ On-line data entry and control information functions are provided in the application.

On-line update

- ▷ The application provides on-line update for the internal logical files.

Reusability

- ▷ The application and the code in the application have been specifically designed, developed and supported to be usable in other applications.

Operational ease

- ▷ Operational ease is a characteristic of the application. Effective start-up, backup and recovery procedures were provided and tested during the system test phase.

Extensibility

- ▷ The application has been specifically designed, developed and supported to facilitate change.

The APC factor has a baseline value of 0.65

$$\text{Adjusted Project complexity (APC)} = 0.65 + (0.01 * PC)$$

$$\text{Total adjusted function points (TAFP)} = APC * \text{TUFP}$$

Lines of code (LOC)

<https://www.qsm.com/resources/function-point-languages-table>

▷ Calculated by

TAFP * Number LOC per FP of each Language

Language	Approximate Number of Lines of Code per Function Point
C	130
COBOL	110
Java	55
C++	50
Turbo Pascal	50
Visual Basic	30
PowerBuilder	15
HTML	15
Packages (e.g., Access, Excel)	10-40

Source: Capers Jones, Software Productivity Research, <http://www.spr.com>

เปรียบเทียบค่า FP เพื่อแปลงไปเป็น LoC ตามมาตรฐานของ QSM
(Quantitative Software Management: www.qsm.com)

Estimate effort

- ▷ **By Constructive Cost Model : COCOMO**

- ▷ **COCOMO Model depend on**

 - Complexity of software

 - Size of the system

 - Experience of developer

 - The type of software

- ▷ **Software have 3 level**

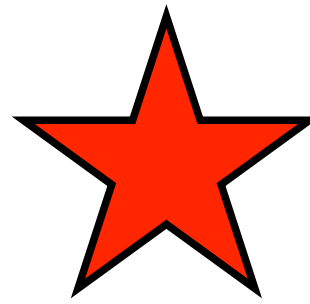
 - Organic

 - Semi-detached

 - Embedded

การประมาณการต้นทุนและ Effort ด้วยแบบจำลอง COCOMO นั้น เป็นการคำนวณจากขนาดของซอฟต์แวร์ ร่วมกับปัจจัยแวดล้อมอื่น ๆ ที่เกี่ยวข้อง เช่น ความแน่นอนของกระบวนการ และความสามารถในการผลิตซอฟต์แวร์ ของทีมงาน ความยืดหยุ่น ความเสี่ยง และวิธีจัดการกับความเสี่ยง

COCOMO Model



สามารถแบ่งออกเป็น 3 ประเภทโครงการดังนี้

ประเภท	ลักษณะโครงการ			
	ขนาด <small>ดูจาก LOC Fluzzie</small>	นวัตกรรมใหม่	ระยะเวลาและ ข้อจำกัด	ลักษณะการ พัฒนา
Organic	เล็ก	เล็กน้อย	มีความยืดหยุ่นมาก	คงที่
Semi Detached	กลาง	บางส่วน	ระยะเวลาเหมาะสม ไม่มีข้อจำกัดมากนัก	ปานกลาง
Embedded	ใหญ่	ค่อนข้างมาก	ระยะเวลาดำเนินการ ต้อง เร่งด่วน และ ข้อจำกัดสูง	ซับซ้อน

$$\text{Effort (E)} = a \text{ KLOC } b$$

	a	b	c	d
Organic	2.4	1.05	2.5	0.38
Semi-detached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

Estimate time

$$\text{schedule time (month)} = c * E^d$$

- This estimate is for
 - ▣ Analysis
 - ▣ Design
 - ▣ Implementation
 - ▣ Does not include the planning

Advantages

- ▷ COCOMO is transparent
- ▷ Estimator to understand the impact of difference factors that affect project costs

Disadvantages

- ▷ It is hard to accurately estimate early on in the project
- ▷ Is not a size measure it is a length measure
- ▷ Vulnerable to mis-classification of the development mode
- ▷ Success depends largely on tuning the model to the needs of the organization, using historical data which is not always available

Example

Description	Complexity			Total
	Low	Medium	High	
Inputs	1*3	3*4	1*6	21
Outputs	3*4	4*5	1*7	39
Inquiries	2*3	5*4	2*6	38
Files	0*7	1*10	0*15	10
interface	2*5	2*7	3*10	54
Total unadjusted function points (TUFP)				162

Project Complexity

	0-5
1. Data Communication	2
2. Heavy use configuration	3
3. Transaction rate	1
4. End-user efficiency	3
5. Complex processing	3
6. Installation ease	1
7. Multiple sites	4

Project Complexity (Cont.)

	0-5
8. Performance	5
9. Distributed functions	3
10. On-line data entry	3
11. On-line update	3
12. Reusability	4
13. Operational ease	4
14. Extensibility	4
Project complexity (PC):	43

$$\begin{aligned}\text{Adjusted Project complexity (APC)} &= 0.65 + (0.01 * 43) \\ &= 1.08\end{aligned}$$

$$\begin{aligned}\text{Total adjusted function points (TAFP)} &= 1.08 * 162 \\ &= 174.96\end{aligned}$$

With JAVA

- ▷ Line of Code = $174.96 * 55$
 - ▷ $= 9623$
 - ▷ Note LOC ทำการหารด้วย 1000 เพื่อทำการ nomalization
- ## Semi-detached

$$\begin{aligned}\text{Effort (E)} &= a \text{ KLOC } b \\ &= 3 * 9.623 * 1.12 \\ &= 29.99 \\ &\approx 30 \text{ persons month}\end{aligned}$$

$$\begin{aligned}\text{schedule time (month)} &= c * E^d \\ &= 2.5 * 30^{0.35} \\ &= 8.22 \text{ months}\end{aligned}$$

Thanks!

Any questions?