



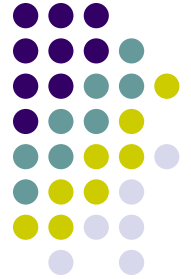
Professor Hausi A. Müller PhD PEng FCAE
Department of Computer Science
Faculty of Engineering
University of Victoria

www.engr.uvic.ca/~seng321/
courses1.csc.uvic.ca/courses/201/spring/seng/321

Deliverable S3a	Fri, Mar 18	S3a Technical Design Spec	15% of project
Deliverable S3b	Tue, Mar 22	S3b Manual	10% of project
Quiz 3: Use cases	Wed, Mar 23	In class	2% of course
Deliverable C3	Thu, Mar 24	C3 feedback on S3a&S3b	10% of project
Easter break	Fri-Mon, Mar 25-28	Fri, no class	
Deliverable S4	Mar 29-Apr 1	S4 project demo (in TWF classes and Tue lab; no lab on Thu)	10% of project
Deliverable C4	Fri, Apr 1	C4 feedback on S4	5% of project
Last Day of Classes	Fri, Apr 1		
Final Exam	Sat, Apr 16	19:00-22:00 ECS 125	35%

SENG 321 Calendar

Announcements

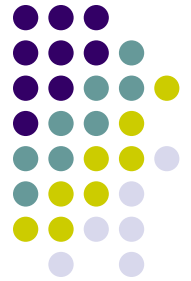


- Tue, March 22
 - S3b due
 - User manual due
- Wed, March 23
 - Quiz 3
- Fri, March 25
 - Good Friday
 - No class
- Tue/Wed/Fri, March 29/30, April 1
 - In class and Tue lab demos
 - No labs on Thu
 - 3 presentations per hour
 - 15 mins per presentation

Final Exam

- Sat, April 16
- 19:00-22:00
- ECS 125

Project Cost and Effort Estimation



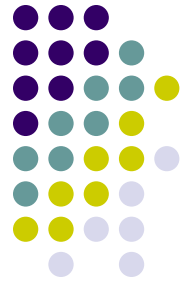
- Project costs
 - Cost components
 - Techniques
 - Advantages and disadvantages
- Algorithmic cost modeling
- COCOMO model



It's all about

MONEY

Project Cost and Effort Estimation

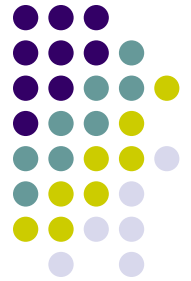


- Software project managers
 - Given specification for software system
 - Responsible for controlling project budgets
 - Must be able to estimate cost of development
- Principal project costs
 - Hardware
 - Travel and training
 - Effort (paying software engineers and others)
- Dominant cost: effort cost
 - Most difficult to estimate and control
 - Has most significant effect on overall costs

LACK OF MONEY
IS THE ROOT OF ALL EVIL

-GEORGE BERNARD SHAW-

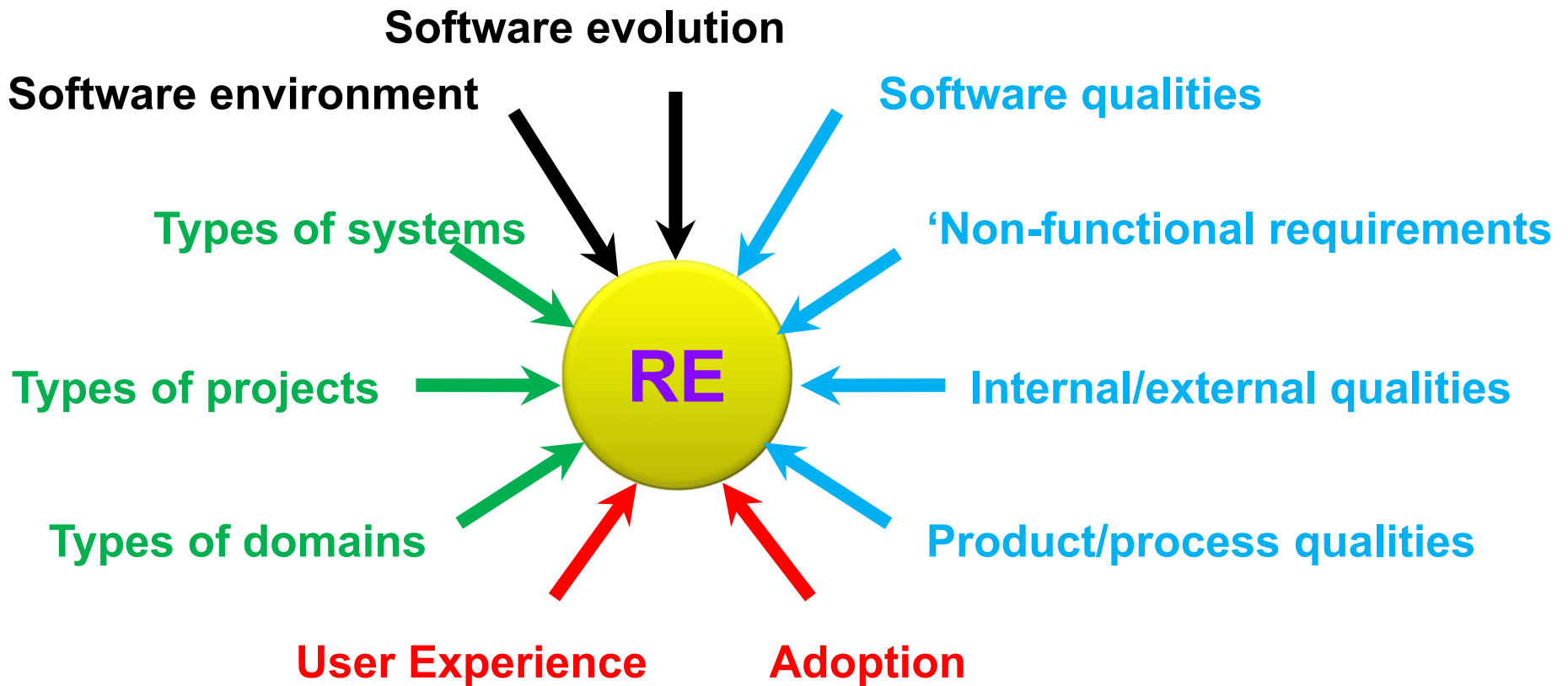
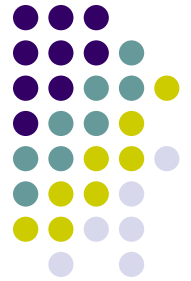
Project Cost and Effort Estimation



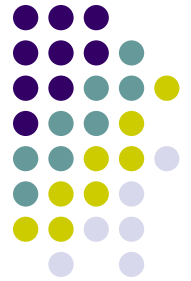
- Objective
 - Rational costing
- Software costing should be carried out objectively
- Accurately predict contractor's development cost
- Software cost estimation is continuous:
 - Starts at proposal stage
 - Continues throughout lifetime of project
 - Projects have budgets ...
 - ... thus cost estimation determines if spending is in line with budget
- Measure-of-effort unit
 - Staff-hour or staff-month

Requirements Engineering

Many Forces at Work

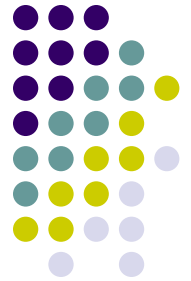


Types of Projects



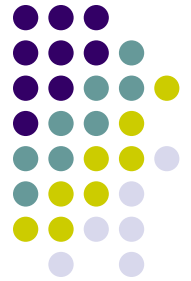
Project types	Customer	Supplier
In-house	User department	IT department
Product development	Marketing/sales department	Software department
COTS-Commercial-of-the-shelf components or products	Company	COTS vendor
Tender	Company	Supplier
Contract development	Company	Software house
Sub-contracting	Supplier	Software house
Other types		

Project Cost and Effort Estimation Techniques



- Seven traditional techniques for software cost estimation
 1. Algorithmic cost modeling
 2. Expert judgment
 3. Estimation by analogy
 4. Parkinson's law
 5. Pricing to win
 6. Top-down estimation
 7. Bottom-up estimation
- Some of these techniques are pathological (i.e., have problems built-in!)
 - → Use more than one method

Project Cost and Effort Estimation Techniques



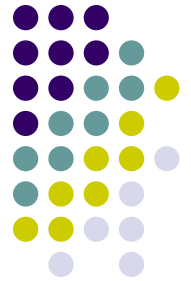
1. Algorithmic cost modelling

- Model developed using historical cost information
- Relates some software metric, usually the size (e.g., KLOC), to project cost
- Estimate made of that metric → model predicts effort required
- For example, COConstructive COst MOdel (COCOMO) model

2. Expert judgement

- One or more experts of proposed development technique are consulted (ideally three experts)
- Should not only be expert in cost modelling, but also expert in application domain
- Estimate project cost
- Final cost is derived by consensus

Project Cost and Effort Estimation Techniques



3. Estimation by analogy

- Applicable when other projects in same application domain have been completed.
- Cost estimated by analogy with completed projects.

4. Parkinson's Law

- Parkinson's Law was first articulated by Cyril Parkinson in a humorous essay in The Economist in 1955:
“Work expands so as to fill the time available for its completion.”
- Cost determined by available resources rather than by objective assessment
 - If software must be delivered in 12 months ...
 - and 5 people are available ...
 - The technique estimates an effort of 60 person-months

MY FAVORITE THINGS IN LIFE
DON'T COST ANY MONEY.
IT'S REALLY CLEAR THAT
THE MOST PRECIOUS RESOURCE
WE ALL HAVE IS TIME.

STEVE JOBS

<http://inspirationalQuotes.Gallery>



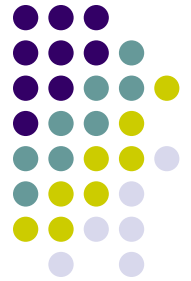


Mythical Man-Month (Brooks)

- Assigning more programmers to a project running behind schedule will make it even later, due to the time required for the new programmers to learn about the project, as well as the increased communication overhead.
 - When N people have to communicate among themselves (without a hierarchy), as N increases, their output M decreases and can even become negative (i.e., the total work remaining at the end of a day is greater than the total work that had been remaining at the beginning of that day, such as when many bugs are created).
 - Group Intercommunication Formula: $n(n - 1) / 2$
 - Example: 50 developers give $50 \cdot (50 - 1) / 2 = 1225$ channels of communication.

http://en.wikipedia.org/wiki/The_Mythical_Man-Month#The_Mythical_Man-Month

Project Cost and Effort Estimation Techniques



5. Pricing to win

- Cost estimated ← what customer has available to spend
- Estimate depends on customer's budget, not on functionality

6. Top-down estimation

- Estimate established by considering overall functionality
- Also determine how functionality provided by interacting sub-functions
- Estimates made on basis of logical function rather than with
- components implementing function.

7. Bottom-up estimation

- Cost of each component estimated
- All costs totalled → final cost estimate

Comparing Techniques



Method	Strengths	Weaknesses
Algorithmic models	<ul style="list-style-type: none"> •Objective, repeatable, analyzable formula •Efficient, good for sensitivity analysis •Objectively calibrated to experience 	<ul style="list-style-type: none"> •Subjective inputs •Assessment of exceptional circumstances •Calibrated to past, not future
Expert judgment	<ul style="list-style-type: none"> •Assessment of interactions, representativeness, exceptional circumstance 	<ul style="list-style-type: none"> •No better than participants •Biases, incomplete recall
Analogy	<ul style="list-style-type: none"> •Based on representative experience 	<ul style="list-style-type: none"> •How representativeness is the experience?
Parkinson's Law	<ul style="list-style-type: none"> •Correlates with some experience 	<ul style="list-style-type: none"> •Reinforces poor practice
Price to win	<ul style="list-style-type: none"> •Often gets the contract 	<ul style="list-style-type: none"> •Generally produces large cost overruns and losses
Top-down	<ul style="list-style-type: none"> •System level focus •Efficient 	<ul style="list-style-type: none"> •Less detailed based •Less stable
Bottom-up	<ul style="list-style-type: none"> •More detailed basis •More stable •Fosters individual commitments 	<ul style="list-style-type: none"> •May overlook system level costs •Requires more effort

