# Lecture 14: Cost Estimation

**Overview**

- Project management activities
  - Project costing
  - Project scheduling and staffing
  - Project monitoring and review
- General cost estimation rules
- Algorithmic Cost Modeling
  - Function point model
  - COCOMO

**Components of Project Cost**

- Tools (both HW and SW)
- Facility, utility, communication, etc.
- Travel, training, etc.
- Acquisition
- Development personnel (cost of person-hours)

**Why Estimating Software Personnel Cost**

- A big portion of overall IS project cost
- Allocate and control budget
  - Justify project value
  - A significant component in economic feasibility study
- Product pricing
  - Fixed vs. variable cost

**Cost Estimation Methods**

- Boehm (1981)
  - Algorithm cost modeling
    - Develop a model using historical cost information
    - Use software metric as input
  - Expert judgment
    - Expert on development technique and application domain
    - Arrive conclusion by consensus
  - Estimation by analogy
    - Other projects in the same project domain
  - Parkinson's Law
    - Work expends to fill the time available
    - The cost is determined by available resources rather than by objective assessments
  - Pricing to win
    - Based on how much the customer can afford

**Estimation Trade-offs**

- Size
  - Function points
  - Lines of code
- Effort
  - Person-months
- Time
  - Months

## Function Points

**Function Point Model**

- Derive cost directly from analysis documents such as DFD and ERD
- Identify five Factors
  - Sizing Data Functions
    1. Files
    2. Program interfaces
  - Sizing Transactional Functions
    3. External Inputs
    4. External Outputs
    5. External Queries

**Function Points Estimation**

Step 1: Total Unadjusted Function Points (TUFP)
Step 2: Project Complexity (PC)
Step 3: Adjusted Project Complexity (PCA)
PCA = 0.65 + 0.01 * PC
Step 4: Total Adjusted Function Points
TAFP = TUFP * PCA

**Lines of Code**

- Conversion:
  LOC = k * TAFP (k is the LOC to FP conversion factor)
- Example:
  - 100 Function Points
  - C: 13,000 Lines
  - VB: 3,000 Lines

**COCOMO Estimation Calculation**

Effort (in person-months) = 1.4 * thousand LOC

**Estimating Schedule Time**

- Rule of thumb for estimation

Schedule Time = 3.0 * Effort ^ (1/3)

## COCOMO

COnstructive COst MOdel

- Use final project size (in terms of lines of code) as model input
- Classify project into three types
  - Organic, semi-detached, embedded
  - Depends on how many other systems to interface with
- Has three forms
  - Basic, intermediate, and detailed

### COCOMO-Basic Form

- Three project types
  - Effort
    - PM= 2.4 $(KLOC)^{1.05}$
    - PM= 3.0 $(KLOC)^{1.12}$
    - PM= 3.6 $(KLOC)^{1.20}$
  - Schedule
    - TDEV= 2.5$(PM)^{0.38}$
    - TDEV= 2.5$(PM)^{0.35}$
    - TDEV= 2.5$(PM)^{0.32}$

- Limitations
  - Does not accommodate *highly sequential forms of incremental development*
  - Calculate *average* staff level in each phases
  - Does not incorporate the effect of *other* software cost drivers

### COCOMO-Detailed Model

- Improvements
  - Phase-sensitive effort multiplier
    - In addition to Size
    - Reliability
    - Application experience
    - Interactive development
  - Three-level product hierarchy
    - Module level
    - Subsystem level
    - System level

## Cost-Effectiveness Analysis

### Performance Models

- Transaction Processing System
  - N = Number of processors
  - S = Processor Speed (Kops/sec)
  - P = Processor Overhead (Kops/sec)
  - M = Multiprocessor overhead factor

- T = Number of operations required per transaction in Kops

**Performance Models**

- Performance

**Production Function**

- Economies of Scale
  - More efficient to produce large quantities than small quantities
- Diseconomies of Scale on large software projects
  - Interpersonal communications
  - Software gold-plating
  - Hardware software
    - Modularity

**Gold Plating**

- Software gold plating
  - Instant response
  - Pinpoint accuracy
  - Unbalanced systems
  - Artificial intelligence (AI) features
  - Interactive multicolor vector graphics
  - "Everything for Everybody" systems

**Gold Plating**

- Usually Not gold plating
  - Humanized Input Preprocessors
  - Humanized Output Postprocessors
  - Modularity and Information hiding
  - Measurement and diagnostic, backup and recovery capabilities
- Sometimes gold plating
  - High generalized control and data structure
  - Sophisticated user command languages
  - General-purpose utilities and support software
  - Automatic trend analysis

**Choosing Among Alternatives**

- Decision Criteria
  - Maximum Available Budget
  - Maximum Performance requirement
  - Minimum Performance requirement
  - Maximum Effectiveness/Cost Ratio
  - Maximum Effectiveness-Cost difference
- Composite options

# Risk Analysis

**Complete Uncertainty**

- No knowledge for chance of success
- Decision Rules
  - Maximin (Most pessimistic)
    - Determine the minimum payoff for each alternative
    - Choose the alternative to maximize the minimum payoff
  - Maximax (Most optimistic)
    - Determine the maximum payoff for each alternative
    - Choose the alternative to maximize the maximum payoff
  - Laplace or Equal-Probability Rule
    - All states of nature are equally likely
    - Determine the expected payoff for each alternative
    - Choose the alternative to maximize the expected payoff
  - Subjective Probabilities
  - Breakeven Analysis

**Value of Information**

- The prototype approach