Development Methodologies

Prof. Dr. Josef M. Joller
jjoller@hsr.ch

CHAPTER 7

PLANNING AND ESTIMATING

Overview

Planning and the software process

Estimating duration and cost

Planning and Estimating

Before starting to build software, it is essential to plan the entire development effort in detail

Planning continues during development and then maintenance

- Initial planning is not enough
- The earliest possible detailed planning is after the specification phase

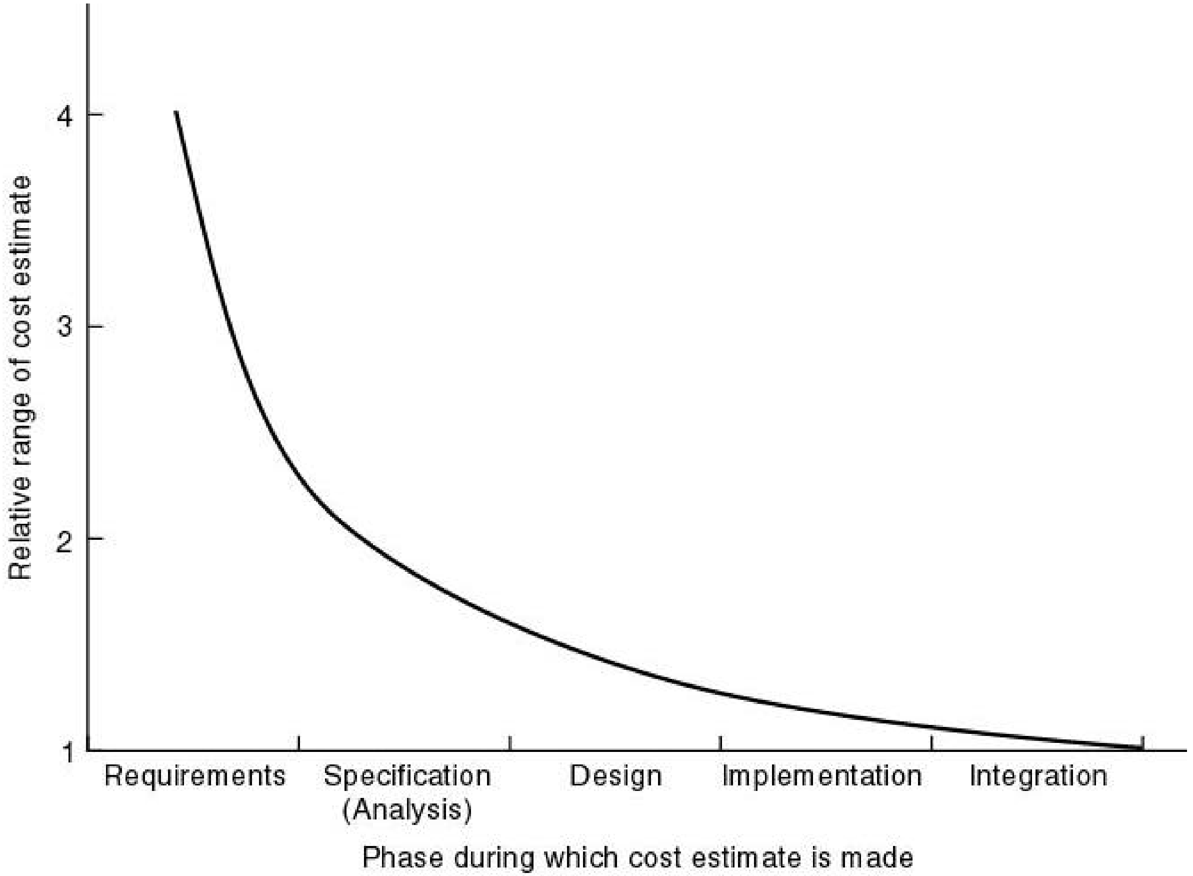
Accurate duration estimation is critical – but how?

Accurate cost estimation is critical – but how?

- Internal, external costs

There are too many variables for accurate estimate of cost or duration

Planning and the Software Process



Accuracy of estimation increases as the process proceeds

Planning and the Software Process (contd)

Example

- Cost estimate of $1 million during the requirements phase
  - Likely actual cost is in the range ($0.25M, $4M)
- Cost estimate of $1 million in the middle of the specification phase
  - Likely actual cost is in the range ($0.5M, $2M)
- Cost estimate of $1 million end of the specification phase (earliest appropriate time)
  - Likely actual cost is in the range ($0.67M, $1.5M)

This model is old (1976)

- Estimating techniques have improved
- But the shape of the curve is likely to be similar

Human Factors

Sackman (1968) showed differences of up to 28 to 1 between pairs of programmers

He compared matched pairs of programmers

- Product size
- Product execution time
- Development time
- Coding time
- Debugging time

Critical staff members may resign during project

Metrics for the Size of a Product

Different estimation techniques exist

- Lines of Code (LOC)
- Software Science
- Function Points
- COCOMO and its variants
- And many more….

Lines of Code

Lines of code (LOC), or

Thousand delivered source instructions (KDSI)

- Source code is only a small part of total software effort
- Different languages ⇒ different lengths of code
- LOC not defined for nonprocedural languages (like LISP)
- It is not clear how to count lines of code
  - Executable lines of code?
  - Data definitions ?
  - Comments?
  - JCL statements?
  - Changed/deleted lines?
- Not everything written is delivered to the client

Lines of Code (contd)

LOC is known when the product finished

Estimation based on LOC is doubly dangerous

- To start estimation process, LOC in finished product must be estimated
- LOC estimate is then used to estimate the cost of the product — uncertain input to an uncertain cost estimator

FFP Metric

For cost estimation of medium-scale DP systems

The three basic structural elements of DP systems

- files, flows, and processes

Given number of files (Fi), flows (Fl), processes (Pr)

- Size (S), cost (C) given by
  - $S = Fi + Fl + Pr$
  - $C = b \times S$

Constant b varies from organization to organization

Validity and reliability of FFP metric were demonstrated using a purposive sample

- BUT, the metric was never extended to include databases

Function Points

Based on number of inputs (Inp), outputs (Out), inquiries (Inq), master files (Maf), interfaces (Inf)

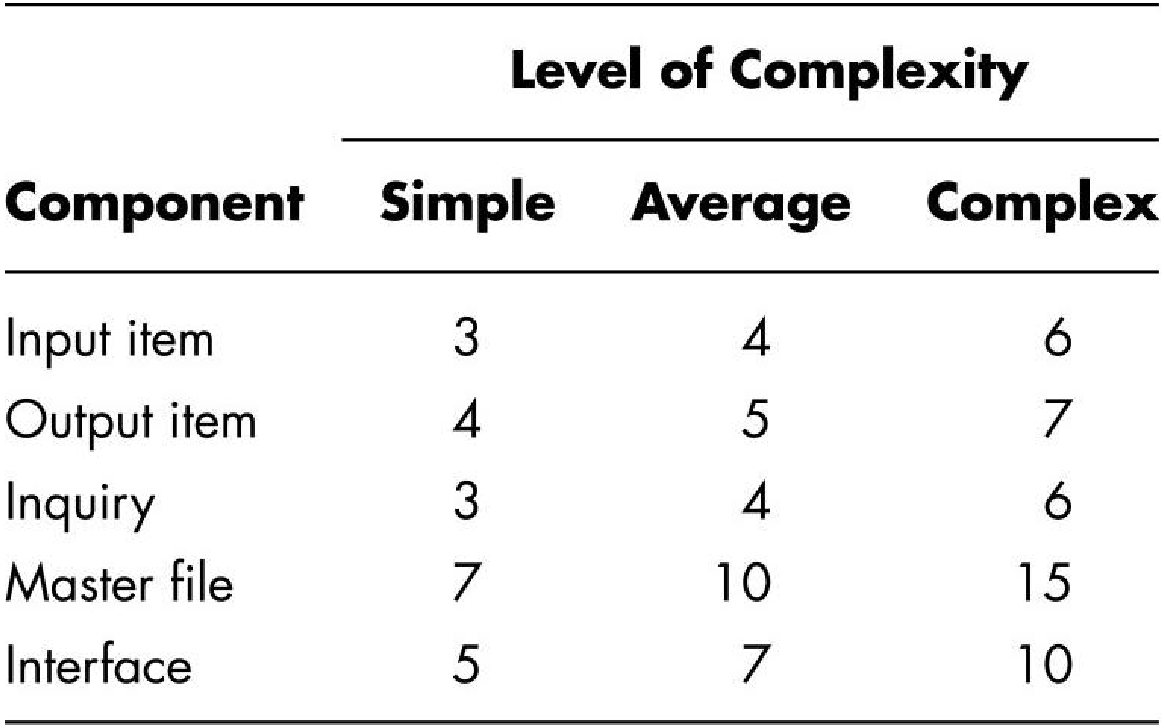
For any product, size in "function points" is given by

$$FP = 4 \times Inp + 5 \times Out + 4 \times Inq + 10 \times Maf + 7 \times Inf$$

Oversimplification of a 3-step process.

Function Points (contd)

1. **Classify each component of product (Inp, Out, Inq, Maf, Inf) as simple, average, or complex.**
   - Assign appropriate number of function points
   - Sum gives UFP (unadjusted function points)



	Level of Complexity		
Component	**Simple**	**Average**	**Complex**
Input item	3	4	6
Output item	4	5	7
Inquiry	3	4	6
Master file	7	10	15
Interface	5	7	10

Function Points (contd)

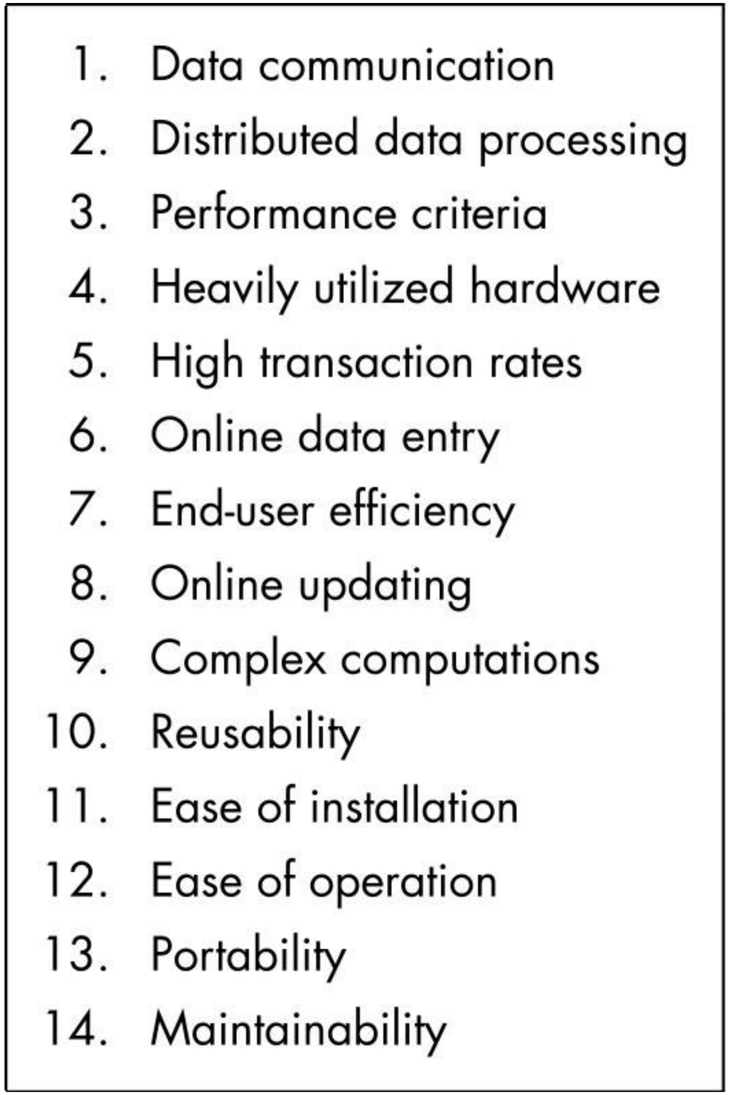
2. **Compute technical complexity factor (TCF)**
   - Assign value from 0 (“not present”) to 5 (“strong influence throughout”) to each of 14 factors such as transaction rates, portability
   - Add 14 numbers ⇒ total degree of influence (DI)

     $$\text{TCF} = 0.65 + 0.01 \times \text{DI}$$

   - Technical complexity factor (TCF) lies between 0.65 and 1.35

3. **Number of function points (FP) given by**

   $$\text{FP} = \text{UFP} \times \text{TCF}$$



1. Data communication
2. Distributed data processing
3. Performance criteria
4. Heavily utilized hardware
5. High transaction rates
6. Online data entry
7. End-user efficiency
8. Online updating
9. Complex computations
10. Reusability
11. Ease of installation
12. Ease of operation
13. Portability
14. Maintainability

Analysis of Function Points

Function points are usually better than KDSI—but there are some problems

“Errors in excess of 800% counting KDSI, but *only* 200% in counting function points” (Jones, 1987)

Like FFP, maintenance can be inaccurately measured

Techniques of Cost Estimation

Expert judgment by analogy

Experts compare target product to completed products

- Guesses can lead to hopelessly incorrect cost estimates
- Experts may recollect completed products inaccurately
- Human experts have biases
- However, results of estimation by broad group of experts may be accurate

Bottom-up approach

- Break product into smaller components
- Smaller components may be no easier to estimate
- Process-level costs