



UNIVERSITÀ DEGLI STUDI  
DI MILANO

## SVILUPPO SOFTWARE IN GRUPPI DI LAVORO COMPLESSO

prof. Carlo Bellettini

prof. Mattia Monga

a.a. 2020-21



UNIVERSITÀ DEGLI STUDI  
DI MILANO

## 22 Stime dei costi

È difficile fare previsioni...  
soprattutto sul futuro

NIELS BOHR

# Fonti dei costi

- Struttura
  - uffici, energia elettrica,...
- Risorse informatiche
  - hardware e software di sviluppo
- Materiali di consumo
  - carta, dischetti, ...
- Personale di supporto
  - segreteria, amministrazione,...
- Personale tecnico
  - programmatori, tecnici, manager,...

# Fattori che influenzano produttività

Nelson-Jones Law (L32)

A multitude of factor influence developer productivity

- Numero di istruzioni da codificare
- Capacità, motivazioni, coordinamento del personale
- Complessità dell'applicazione
- Stabilità dei requisiti
- Prestazioni o qualità richieste
- Strumenti di sviluppo usati
- ...

# Tecniche disponibili

- Stime Umane
  - Legge di Parkinson sul lavoro
  - Price to win
  - Esperti
  - Per analogia
- Stime Automatizzate
  - Algoritmiche
  - Machine Learning

# Legge di Parkinson

Il costo dipende dalle risorse disponibili

- *“Work expands to fill the available volume”*

Esempio di applicazione:

*This flight control software must fit on a 65,536 word machine...*

- *therefore its size will be roughly 65,500 words.*

*It must be done in 18 months, and there are 10 people available to work on it...*

- *so the job will take roughly 180 man-months*

# Price to win

Il costo è quanto il cliente è disposto a spendere (sia soldi che tempo)

- VANTAGGIO: spesso si ottiene il lavoro
- SVANTAGGIO: nessuna garanzia sul lavoro

potrebbe anche andare bene se:

- Possibile ricontrattazione dei requisiti a posteriore
- Possibile sviluppo incrementale

# Esperti esterni

Non è una tecnica, ma una prassi comune

- Si delega la responsabilità ad altri
  - perché sono più bravi
  - perché hanno maggiore conoscenza del dominio
  - perché così non si tolgono persone interne da attività più importanti
  - perché si impegnano a darci un risultato entro una certa scadenza
- Come fanno gli esperti a fare le stime?

# Stima per analogia

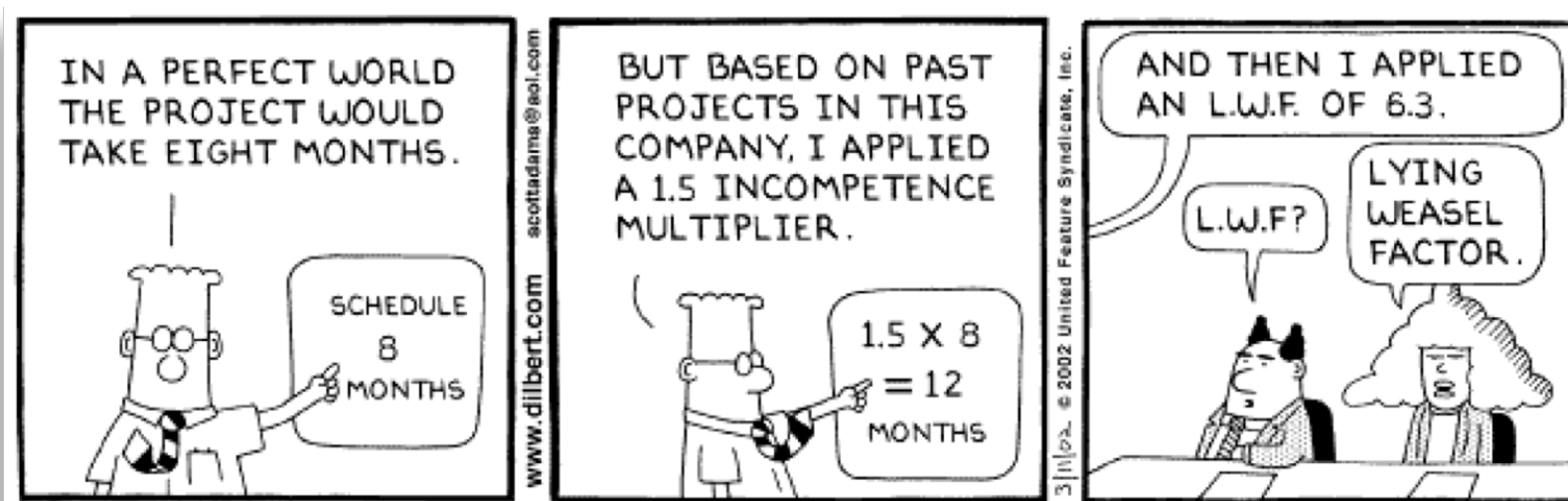
Si cerca di confrontarlo con un progetto precedente di cui la quantità di lavoro necessaria (mesi-uomo) e i tempi di sviluppo (mesi) sono noti

- coefficienti per gli scostamenti
  - più grande del...
  - maggiore esperienza...
  - riuso...

è un metodo top-down

- si prende il progetto nel suo insieme e successivamente si suddivide la spesa

# Le stime dei costi



Copyright © 2002 United Feature Syndicate, Inc.

DeMarco-Glass Law (L34)  
Most cost estimates tend to be low

# Stima basata su dimensione del codice

Si scompone funzionalmente il progetto e si identificano i componenti da sviluppare

- di questi (meno complessi del problema originale) si effettua una stima della dimensione
- bottom-up
  - si ricompone la spesa totale a partire dalla base
- maggiore è la scomposizione
  - maggiore è il dettaglio in cui si scende
  - minore è la difficoltà di stima dei singoli componenti

# Stima basata su quantità di lavoro

- si scompone il progetto secondo una logica operativa (per fasi) e funzionale (componenti)
- si stima il tempo necessario per svolgere le singole fasi per il determinato componente
- ancora bottom-up

# Alcuni dati storici da raccogliere

- Produttività
  - LineeCodice / MeseUomo
- Qualità
  - Errori / LineeCodice
- Costo Unitario
  - CostoTotale / LineeCodice

# Stime automatizzate: Algoritmiche

- Boehm
  - COCOMO (COConstructive COSt MOdel)
  - COCOMO 2
- Putnam
  - Software Life Cycle Management (SLIM)

# Modello COCOMO

- anni 80
- Assume un modello a cascata a 4 fasi
  - Pianificazione e analisi requisiti
  - Progetto
  - Sviluppo
  - Integrazione e testing
- Parte dalla misurazione di casi reali e estrazione di formule da essi

# Tipo applicazione e Modello

Le formule presentano coefficienti dipendenti da **tipo applicazione**

- Semplici (Organic)
- Intermedie (Semi Detached)
- Complesse (Embedded)

Esistono tre **modelli** di formule

- Base
  - dipende solo dalla stima delle linee di codice
- Intermedio
  - 15 fattori correttivi legati a caratteristiche del progetto
  - Possibilità di effettuare stima singoli componenti
- Avanzato

# COCOMO Modello Base

$$\text{Mesi-uomo} = a \text{ LOC}^b$$

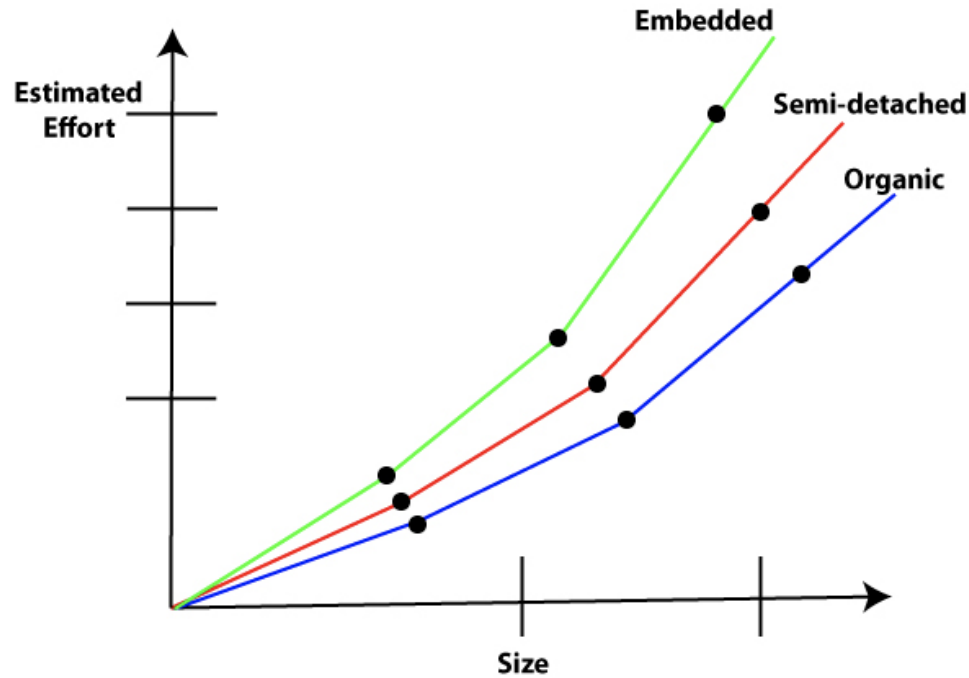
$$\text{Mesi-Sviluppo} = c \text{ Mesi-uomo}^d$$

Bohem's Third Law (L33)

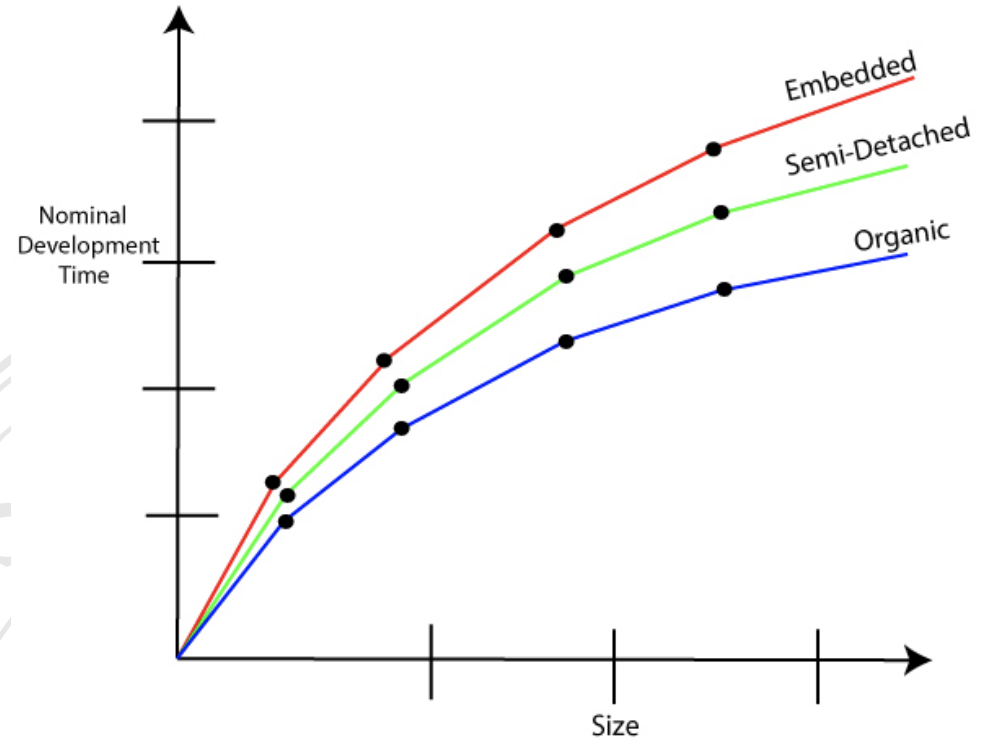
Development effort is a non linear  
function of product size

| parametro | Semplice | Intermedia | Complessa |
|-----------|----------|------------|-----------|
| a         | 2.40     | 3.00       | 3.60      |
| b         | 1.05     | 1.12       | 1.20      |
| c         | 2.50     | 2.50       | 2.50      |
| d         | 0.38     | 0.35       | 0.32      |

# COCOMO andamenti



Effort versus product size



Development time versus size

# Dimensionamento team

Qual è il numero ottimale?

COCOMO fornisce Mesi-uomo e Mesi-sviluppo

Non vede quindi il numero di componenti del team come fattore su cui giocare per accelerare il progetto, ma come misura derivata

$\text{Dimensione} = \text{MesiUomo} / \text{MesiSviluppo}$

# COCOMO Intermedio

15 fattori correttivi

## Legati al prodotto

- affidabilità richiesta
- dimensione database
- complessità

## Legati alle macchine

- efficienza richiesta
- memoria richiesta
- variabilità ambiente sviluppo
- tempi di risposta

## Legati al progetto

- tecniche di programmazione avanzate
- strumenti avanzati
- tempi di sviluppo stretti

## Legati al personale

- esperienza analisi
- esperienza di dominio
- esperienza sviluppo
- conoscenza ambiente
- conoscenza linguaggio

# COCOMO intermedio

$$\text{Mesi-uomo} = a \text{ LOC}^b \prod c_j$$

$$\text{Mesi-Sviluppo} = c \text{ Mesi-uomo}^d$$

(a e b non sono più quelli di prima)

LOC dovrebbe essere calcolato a partire da componenti e non solo stima totale

# COCOMO 2

Raffina COCOMO tenendo conto di nuovi modi di sviluppare il software

- Prototipizzazione
- Riuso
- COTS

Ha diverse formule a seconda del momento della stima

- Prime fasi e prototipizzazione
- Fase di progettazione
- Fase successiva alla definizione della architettura

# Stime automatizzate: Machine learning

- Neural Networks
- Fuzzy Logic
- Case-Based Reasoning
- Analogy
- Rule-Based
- Regression Trees
- Hybrid System

# Stime collettive/condivise

Wideband Delphi ([http://en.wikipedia.org/wiki/Wideband\\_delphi](http://en.wikipedia.org/wiki/Wideband_delphi))

Nei metodi agili la responsabilità della stima passa dai manager (imposizione) a team sviluppatori (autodeterminazione)

- Planning Poker (<http://planningpoker.com>)
- Team Estimation Game (<http://www.agilelearninglabs.com/2012/05/how-to-play-the-team-estimation-game/>)
- #NoEstimate (<http://neilkillick.com/2013/01/31/noestimates-part-1-doing-scrum-without-estimates/>)

# Problemi nelle stime condivise

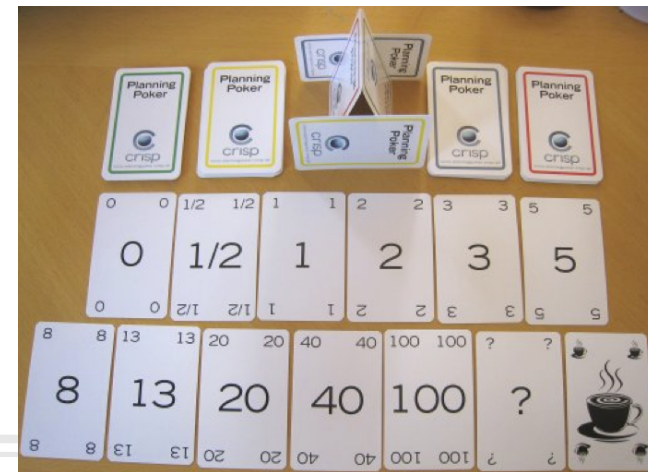
- Perdita di tempo -troppa comunicazione
- Anchoring effect ( <http://en.wikipedia.org/wiki/Anchoring> )
  - dipendenza dai primi pezzi di informazione che si ottengono

# Dilbert on Anchoring effect



# Stime agili: planning poker

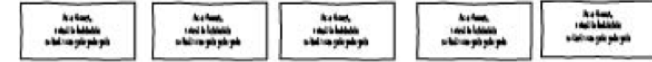
- Vengono presentate brevemente le carte
- Il team può fare domande, richiedere chiarimenti e discutere per chiarire assunzioni e rischi
- Ogni componente sceglie una carta del poker rappresentante la propria stima
  - unità di misura?
- Le carte vengono girate contemporaneamente
- Chi ha espresso le stime più basse e più alte ha “un minuto” per spiegare le sue motivazioni e cercare di convincere gli altri.
- Si ripete la votazione fino a... quando si raggiunge unanimità



# Stime agili: Team Estimation Game

<- Smaller

Larger ->



## PRIMA FASE: VALUTAZIONE COMPARATIVA

- I developer si mettono in fila
- Si fa una “pila” con le storie e si mette la prima carta al centro del tavolo
- Il primo developer della fila prende una carta dalla pila, la rilegge (ad alta voce) e la posiziona a sinistra (più semplice) a destra (più complicata) sotto (equivalente).
- Il prossimo developer può:
  - prendere una nuova carta dalla pila e posizionarla secondo le stesse regole (può anche distanziare due file e metterla in mezzo tra le due)
  - spostare una carta precedentemente posizionata (commentando la motivazione della sua azione)
  - se non ci sono carte nella pila e non vuole spostare esistenti, “passare”
- Si ripete il passo precedente fino a quando a. non ci sono più carte sulla pila E b. nessun developer vuole spostare una carta

# Stime agili: Team Estimation Game

## SECONDA FASE: QUANTIFICARE DISTANZE

- ci si mette di nuovo in coda davanti al tavolo, il mazzo di carte questa volta è tipo quello senza carte "strane"... quindi si è scelta una scala di valori )
- si posiziona la prima carta (di solito si parte dal valore 2 perché potrebbe esserci nella prossima iterazione qualcosa di ancora più semplice) sulla prima colonna
- il primo developer prende il valore successivo e lo posiziona sulla prima colonna che pensa abbia quel valore (rispetto a quello con valore 2) o tra due colonne
- Il prossimo developer può:
  - prendere una nuova carta valore dalla pila e posizionarla secondo le stesse regole (la prima colonna che gli sembra abbia bisogno di tale valore)
  - spostare una carta valore precedentemente posizionata (commentando la motivazione della sua azione)
  - se non ci sono carte nella pila e non vuole spostare esistenti, "passare"
- Si ripete il passo precedente fino a quando a. non ci sono più carte sulla pila E b. nessun developer vuole spostare una carta
- Le colonne senza una carta valore sopra, vengono assimilate alla colonna alla loro sinistra



# Stime agili: Team Estimation Game

## TERZA FASE: SCALA ASSOLUTA

- Si stima il tempo in ore/uomo di una delle carte più semplici (valore 2)... (simil planning poker?)
- Si calcolano con le proporzioni dei valori tutti gli altri

Necessario?

# Velocity

- capacità (osservata) di completare lavori da parte del team
- Sostituisce la necessità di rimappare unità ideali in tempi assoluti
- Permette di compensare tendenze sbagliate di stime all'interno del team
- NON deve essere usato come metro di valutazione tra team, o nel tempo
- NON si devono considerare storie non finite
- NON deve essere imposta

# #NoEstimate

Perché?

- sostiene che fare le stime NON fornisce valore all'utente?

Funziona?

