

# Sommario

- Gestione personale
- Gestione rischi
- Gestione qualita'
- Gestione modifiche e configurazione

# Nuova gestione

- Continuo impatto tecnologia
- Reti di organizzazioni
- Gestione innovativa e tollerante del rischio
- Collisione con contabilita' e proceduralita'
- Collaborazione opportunistica tra individui del team

# Personale

- Piano di formazione e gestione dello staff
- Team autogestiti (in voga)
- Managers e leaders
  - Leader importante in grandi progetti
  - Membri del team ispirati e motivati (?!)

# Creazione team

- Riassegnazione da altri gruppi
  - Conosciuti, ma non sempre i migliori o piu' adatti
- Assunzione
  - I migliori? conosciuti?
  - Test e colloqui
- Motivazione dei tecnici

# Teorie motivazione personale

- Teoria della realizzazione
- Intrinseca od estrinseca ?
- Gerarchia di Maslow
- Teoria dell'igiene
- Teoria delle aspettative

# McClelland

- La combinazione dei 3 fattori determina il risultato
  - Desiderio di successo (achievement)
  - Potere
  - Affiliazione

# McClelland

- Achievement

- 
- Has a strong need to set and accomplish challenging goals.
- Takes calculated risks to accomplish their goals.
- Likes to receive regular feedback on their progress and achievements.
- Often likes to work alone.
- 

- Affiliation

- 
- Wants to belong to the group.
- Wants to be liked, and will often go along with whatever the rest of the group wants to do.
- Favors collaboration over competition.
- Doesn't like high risk or uncertainty.
- 

- Power

- 
- Wants to control and influence others.
- Likes to win arguments.
- Enjoys competition and winning.
- Enjoys status and recognition.
-

# Gerarchia di Maslow a 5 livelli

- Ogni volta che soddisfo un livello, sono motivato dal successivo, studio solo grandi personalita'
  - Fisiologico
    - Respirare cibo sesso...
  - Sicurezza
    - fisica, salute, lavoro, proprieta'
  - Socialita'
    - Amicizia, famiglia
  - Stima
    - Autostima, realizzazione, rispetto per/verso altri
  - Autorealizzazione
    - Creativita', moralita, soluzione problemi, mancanza pregiudizi



# Igiene-motivazione di Herzberg

- Fattori igienici
  - Company policy ,
  - Supervision
  - Relationship w/Boss
  - Work conditions
  - Salary
  - Relationship w/Peers
- Prevengono l'insoddisfazione ma non motivano veramente

# Igiene-motivazione di Herzberg

## Motivatori (non opposti agli igienici)

- Achievement
- Recognition
- Work itself
- Responsibility
- Advancement
- Growth

# Igiene-motivazione di Herzberg

- - Impegnare appieno risorse del lavoratore
  - Maggiori capacita' -> maggiori responsabilita'
  - Altrimenti lavoratori meno qualificati o automazione

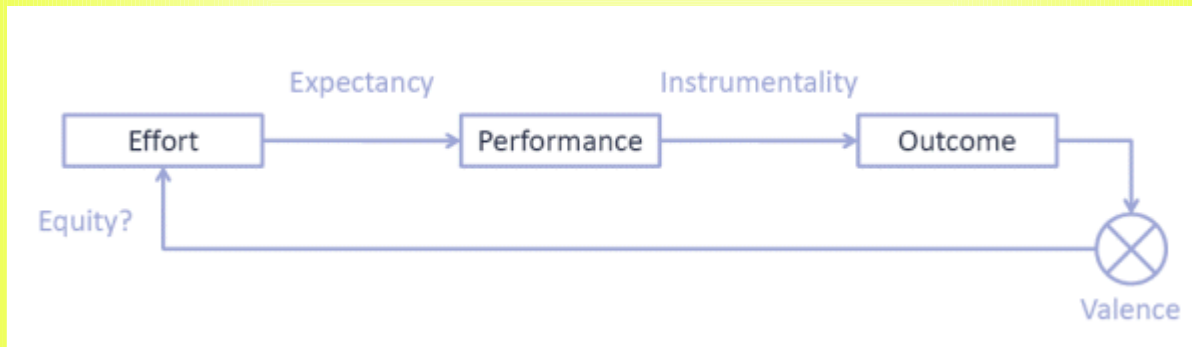
# Igiene-motivazione di Herzberg

- Bastone e carota funzionano ma a breve
- Eliminare prima insoddisfazione
- Critica
  - Non e' detto che la realizzazione implichi alta produttivita'
- Che ne dite?

# Teoria delle aspettative (Vroom)

- L'individuo si comporta in base alle aspettative ambientali se portano ricompense
- Per motivare, le promesse di ricompensa vanno mantenute (?!)
- I managers dovrebbero
  - Creare alte aspettative
  - Premiarle apertamente

# Teoria delle aspettative (Vroom)



# Teoria delle aspettative (Vroom)

- Expectancy (se lavoro di piu' le cose andranno meglio)
- Dipende dalla disponibilita' di
  - Risorse
  - Abilita'
  - Supporto (ambiente/informazioni)

# Teoria delle aspettative

(Vroom)

- Instrumentality (se lavoro meglio avro' una ricompensa)
- Dipende da
  - Capire le regole del gioco
  - Fiducia in chi premia
  - Trasparenza processo



# Teoria delle aspettative (Vroom)

- Valence: importanza attribuita al risultato
- Se ad es mi interessa piu' tempo libero potrei non volere piu' soldi
- Le persone modificano I loro sforzi in base alla forza della relazioni che vedono tra questi punti

# Teoria delle aspettative (Vroom)

- Se una sola e' debole c'e' scarsa motivazione
  - Il mio aumento di sforzo non aumenta le mie prestazioni
  - Un aumento di prestazioni non aumenta il mio premio
  - Non sono interessato ai premi in palio

# Teoria delle aspettative (Vroom)

- Funziona sulla percezione:
- Anche se la ditta ha fatto il possibile, qualcuno potrebbe pensare diversamente
- Si puo' applicare anche a contesti estranei:
  - Piu' mi sforzo a riciclare piu' riciclo
  - Piu' riciclo meno risorse si usano
  - Meno risorse si usano piu' sono contento

# Gestione rischi

- Un progetto puo' essere avanti come sviluppo ma indietro come rischio
- Strategie reattive e proattive
- Affrontare il rischio
- Identificazione
- Valutazione
- Gestione

# Identificazione rischi

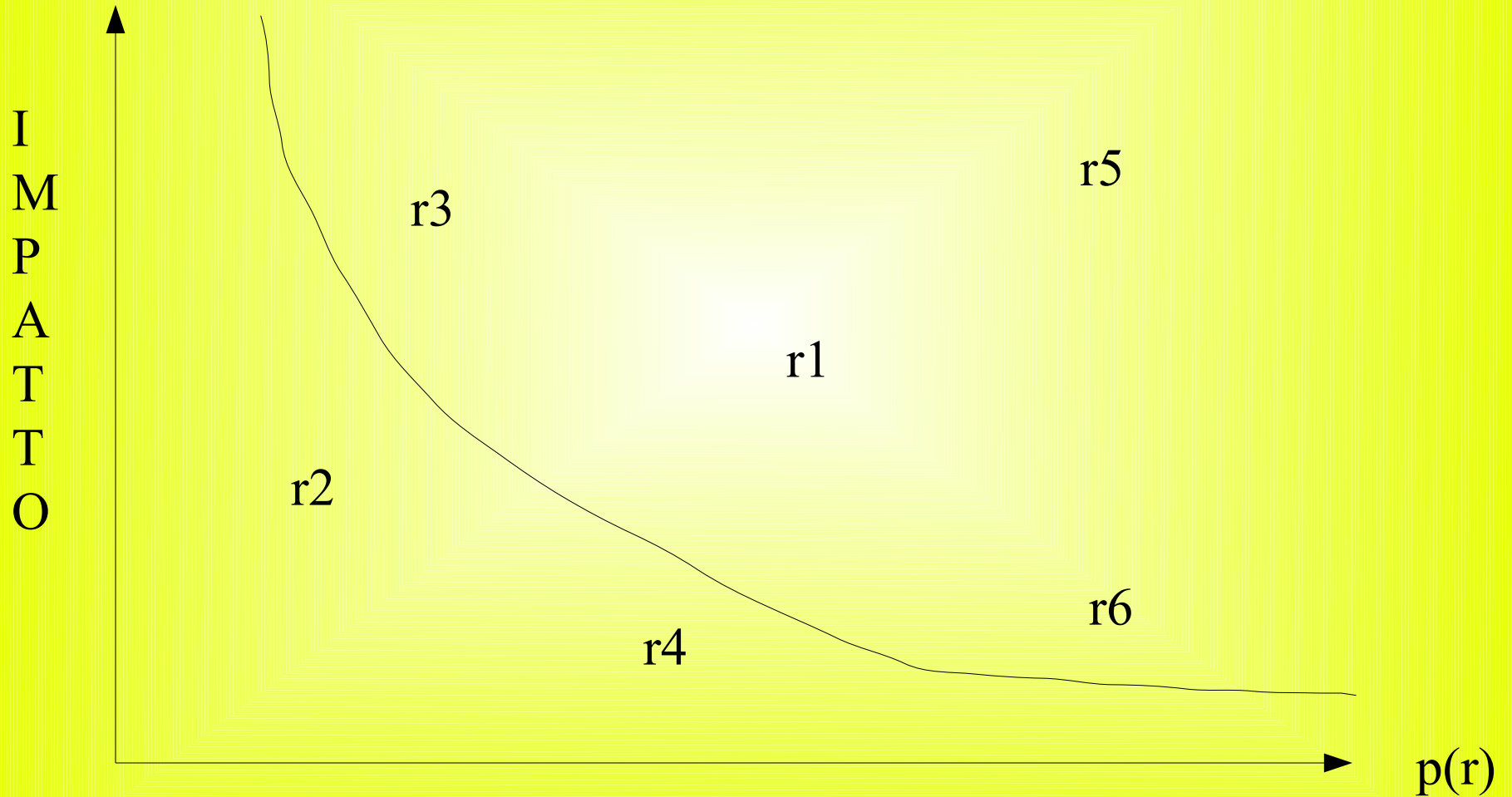
- Prodotto
- Processo
- Persone
- Business
- Organizzazione
- Tecnologia
- Strumenti
- Clienti
- Requisiti
- Altri progetti
- Fattori esterni

# Valutazione rischi

- Detta analisi, stima, proiezione
- Ordinati (guarda caso) per:
  - Probabilità'
  - Impatto (catastrofico,, marginale) rispetto:
    - Tempo
    - Costo
    - Prestazioni
    - Supportabilità'

# Valutazione rischi

- Curva di eliminazione



# Valutazione rischi

- Creazione di livelli di rischio per stabilire
  - Punti di terminazione del progetto
- Decisione politica con gravi impatti
- Valutazione soggettiva, derivante anche dai dati in input ai metodi di calcolo del rischio



# Gestione della qualita'

- Team di gestione qualita'
- Indipendente con propri canali di comunicazione
- Piano di gestione
- Standard di qualita'
-

# Qualita'

- Correttezza
- Affidabilita'
- Robustezza
- Prestazioni
- Usabilita'
- Comprensibilita'
- Manutenibilita'
- Scalabilita'
- Riusabilita'
- Portabilita'
- Interoperabilita'
- Produttivita'
- Tempistica
- Visibilita'

# Qualita' come

- Checklist
  - Lista di domande a cui rispondere
  - Serve anche per certificazioni
  - Puo' servire di supporto ad altre procedure
  - Aiuta a costruire cultura della qualita'

# Qualita'

- Review: riunione formale
  - Walkthrough (tecnici)
  - Ispezione (con managers)
- Guidata da un documento
- Poche persone
- Registrazione dei problemi
- Riunione successiva di controllo

# Qualita'

- Audit: simile ai normali audit
- Documentazione e studi sul sistema
- Interviste, ispezioni
- Conclusioni, valutazioni
- Ambito ampio, includendo:
  - Gestione processo
  - Considerazioni legali

# Testare la qualita'

- Test delle funzionalita'
- Test dei vincoli
- Documentazione di supporto
- Gestione dei cambiamenti
- Tracciabilita'
- 
- Vedi serie di slide bruegge

# Gestione dei cambiamenti

- Difetti:
  - trattati direttamente dall' engineering
- Requisiti:
  - Approvazione tramite processo formale
    - Proposta di cambiamento
    - Comitato preposto all' approvazione
    - Richiesta di cambiamento
  - Relativo cambiamento dei test

# Gestione dei cambiamenti

- I test rivelano dei difetti
- I difetti richiedono cambiamenti di due tipi
  - Errori da correggere
  - Miglioramenti da apportare
- Entrambi
  - Evolvono attraverso stati (es errori: aperto, non riproducibile, individuato ,riparato ...)
  - Hanno prioritá'
  - Hanno un responsabile
  - Sono ricondotti ai documenti di progetto
- Gestione nei gruppi di progetto
  - Controllo versioni



# Richiesta dei cambiamenti (1)

- Procedura meno discrezionale (piu' automatizzabile) per gli errori
- Sottomissione di un difetto:
- se il difetto e' segnalato da un utente , se ne verifica preliminarmente l'esistenza:
  - scarsita' di addestramento
- Altrimenti viene inserito con associato **caso d'uso** , **priorita' e gravita'** e notificato al responsabile del team di sviluppo presumibilmente coinvolto. L'errore puo' appartenere anche alla documentazione
- Il responsabile individua un tecnico a cui assegnare la riparazione

# Richiesta dei cambiamenti (2)

- Se il difetto non appartiene al componente inizialmente imputato, inizia una fase di investigativa per l'attribuzione corretta
- La priorit  puo' essere modificata dal manager ma non dal tecnico
- Difetti non riproducibili vengono generalmente ignorati
- Sistemi di gestione tengono traccia del processo e forniscono statistiche (tipiche di un RDB)
  - Numero di difetti non assegnati
  - Numero di difetti irrisolti
  - Rapporto scoperti/risolti (se peggiora?)

# Tracciabilita'

- Analisi costi/benefici
  - Piu' si traccia e piu' costa
  - Fallimento del progetto
  - A lungo termine: quantificabilita' del progetto
- Livelli
  - Semplice : dai difetti ai casi d'uso
  - Migliore: dai difetti ai casi d'uso ai test necessari per evitare regressioni
- Il sistema di gestione del progetto puo' includere la gestione degli errori e propagare a cascata le modifiche implicate dalla soluzione dei difetti a tutta la documentazione

# Baseline e release

- Baseline ogni volta che una configurazione viene congelata ed immessa nello spazio di lavoro condiviso
- Il manager puo' decidere di promuovere a release una baseline
  - Molto piu' lavoro: codice di altri componenti, manuale (insomma un prodotto)

# Metriche del SW

- Insieme di numeri per misurare le varie caratteristiche
- Approssimati e soggettivi meglio di niente
- CK (Chidamber + Kemerer)
  - Weighted methods per class (WMC)
  - Depth of inheritance tree (DIT)
  - Number of children (NOC)
  - Coupling between object classes (CBO)
  - Response for a Class (RFC)
  - Lack of cohesion in methods (LCOM)

# Weighted methods per class (WMC)

- Somma pesata delle complessità dei metodi di una classe
- Può stimare lo sforzo nella sua implementazione
- Valore alto provoca alta dipendenza di ereditarietà se riusata come superclasse
- È molto specifica alla applicazione, difficile da riusare

# Depth of inheritance tree (DIT)

- lunghezza cammino di ereditarietà di una classe
- Classe con numero elevato eredita molti metodi, può essere difficile da comprendere
- Numero alto in tante classi indica complessità di progetto
- Numero alto indica che la classe sfrutta molto ereditarietà

# Number of children (NOC)

- 
- 
- Numero di sottoclassi di una classe
- Numero alto puo' indicare scadente astrazione nella classe



# Coupling Between Object classes (CBO)

- bidirezionale dipendenza tra due classi
- Es metodi invocati
- Introduce dipendenza (ovviamente!) e complica il progetto

## Lack of cohesion in methods (LCOM)

- Misurato ad es tramite variabili accedute in comune da metodi diversi
- Design piu' complesso
- La classe dovrebbe essere spezzata

# Response for a class (RFC)

- Numero di messaggi potenzialmente richiesti per rispondere ad un messaggio inviato da una classe
- Classe complessa
- Da' buona stima, essendo caso pessimo, dello sforzo di testing

# Pianificazione e Controllo del progetto

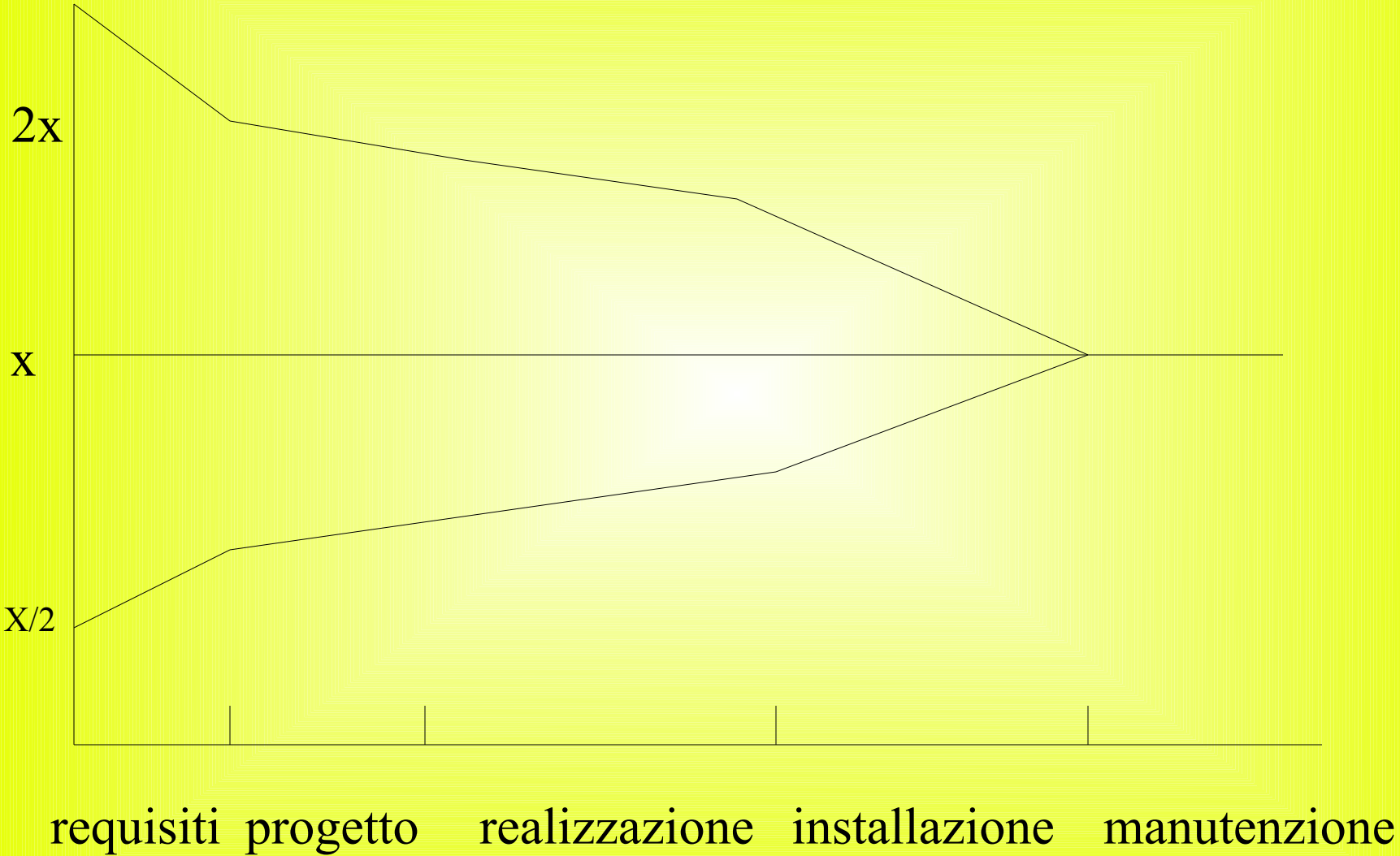
# Sommario

- Pianificazione
- Scheduling
- Stima costi
- Controllo
  - Scheduling
  - Costi

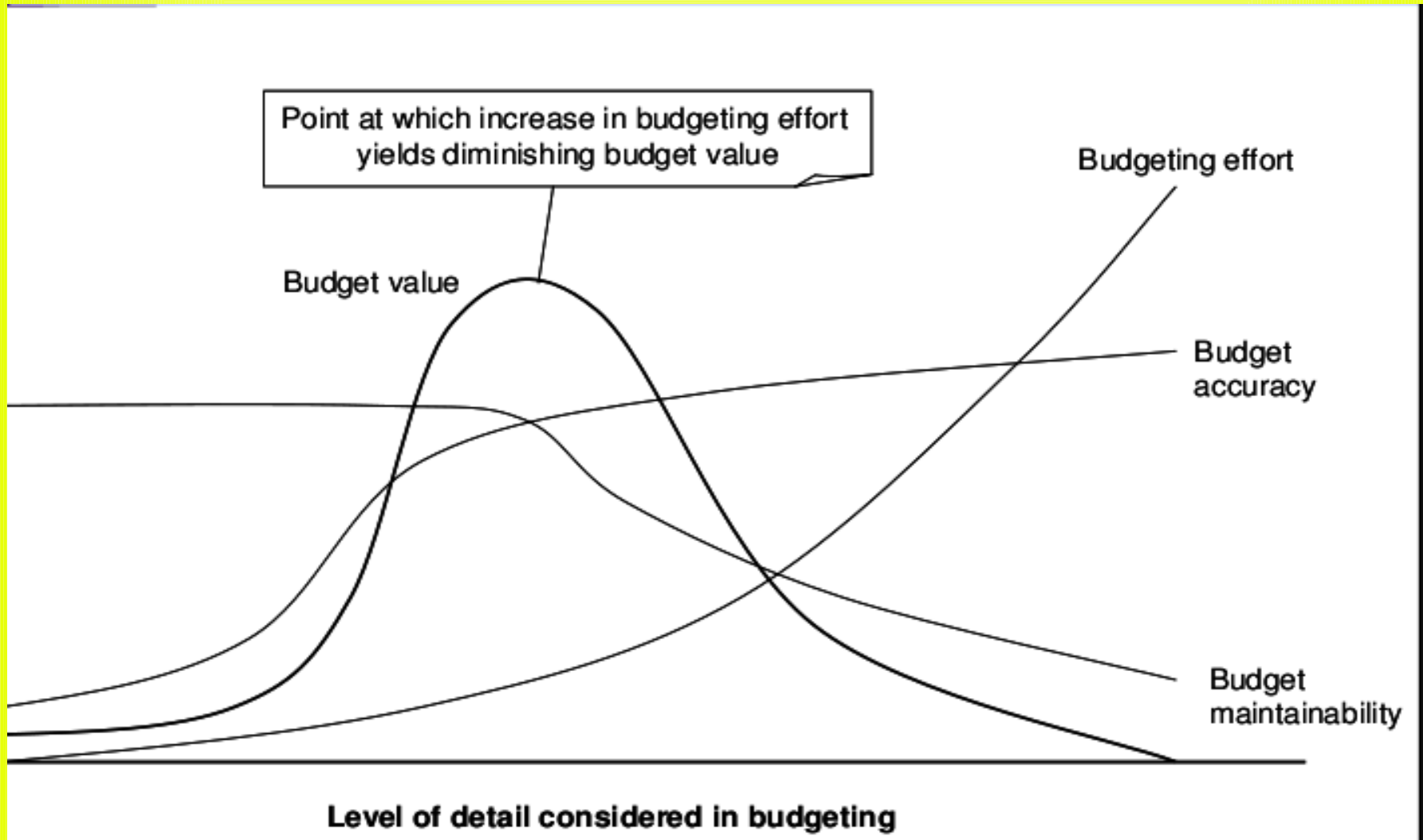
# Management

- Ovviamente, include piu' o meno tutto:
  - Scheduling
  - Costi
  - Rischi
  - Qualita'
  - Personale

# stima costo tempo



# Stima budget





# Stima budget

- Giudizio esperto
- Per analogia
- Guidata dallo schedule
- Algoritmica

# Budget derivato da schedule

- Sembra facile:
  - Assegno risorse ai task
  - Assegno costo alle risorse
  - Aggiungo costi fissi
- Ma:
  - Le risorse contribuiscono in modo variabile
  - Contabilita' , etc.

# Budget basato su algoritmo

- Stima empirica costo risorse umane basato su dimensione progetto
- Sforzo =  $C * \text{dimensione}^k$
- $K$  rappresenta l'incremento di complessita' (tra 1 e 1.5)
- $C$  basato su difficulta' progetto (qualita' etc)
- COnstructive COst Model (COCOMO)
  - Misura in migliaia linee di codice, mesi uomo
  - Regressione statistica su dati storici

# COCOMO

- 3 tipi di progetto:
- A) Piccoli team esperti, vincoli non rigidi
- B) Team medi, medio esperti, vincoli piu' rigidi
- C) Vincoli rigidi (di tutti i tipi)

# COCOMO

- Coefficienti  $a_1$ ,  $b_1$ ,  $c_1$ ,  $d_1$
- Sforzo =  $a_1 * KLOC^{b_1}$  (mesi uomo)
- Tempo =  $c_1 * Sforzo^{d_1}$  (mesi)
- Personale = Sforzo / Tempo

# COCOMO

| progetto      | a1  | b1   | c1  | d1   |
|---------------|-----|------|-----|------|
| Organico      | 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 |

# Esempio COCOMO

Progetto da 10K linee di codice

- Sforzo organico :  $2.4 * 10^{**} 1.05 = \sim 27$ 
  - $\sim 27$  mesi uomo
- Sforzo embedded:  $3.6 * 10^{**} 1.2 = \sim 57$ 
  - $\sim 57$  mesi uomo

# COCOMO intermedio

- Complessita' prodotto
- Capacita' analista
- Capacita' programmatore
- Affidabilita' SW
- Vincoli velocita'
- Conoscenza dell'applicazione
- Conoscenza del linguaggio



# COCOMO intermedio

- Tiene anche conto della relativa importanza delle diverse fasi

# Esempi di altri numeri

- Si possono creare modelli diversi, in diverse fasi del progetto
- Es numero di interfacce, videate, componenti (Object Points)
- Calcolo produttività/efficienza della ditta
- Sforzo = OP/ produttività'

# Controllo budget

- Tutti i casi possibili
  - sopra/sotto budget
  - Sopra sotto/schedule
- Metodologia Earned Value Analysis
- Non ne parliamo