

## Processo – parte I

Leggere Sez. 7.1 e 7.2 Ghezzi et al.



università di ferrara  
DA SEICENTO ANNI GUARDIAMO AVANTI.

## Processo software

- Organizzazione delle attività che portano alla definizione, costruzione, consegna ed evoluzione di un prodotto software, dalla sua ideazione al suo ritiro.
- Lo studio dei processi software comprende:
  - Definizione degli aspetti pratici delle attività coinvolte
  - Relazioni fra attività

Processo 1



università di ferrara  
DA SEICENTO ANNI GUARDIAMO AVANTI.

2

## Ciclo di vita del software

- Analisi e specifica dei requisiti
- Specifica e progetto architetturale
- Codifica e verifica
- Consegna e installazione
- Manutenzione ed evoluzione
- Ritiro

Processo 1



università di ferrara  
DA SEICENTO ANNI GUARDIAMO AVANTI.

3

## Modelli di processi software

- Tentativo di organizzare il ciclo di vita del software
  - Individuando le attività coinvolte nella produzione del software
  - Definendo le relazioni fra esse (temporali e non)
- Obiettivi:
  - standardizzazione, prevedibilità, produttività, qualità del prodotto, allocazione corretta di risorse finanziarie e di tempo

Processo 1



università di ferrara  
DA SEICENTO ANNI GUARDIAMO AVANTI.

4

## Code & fix

- Primo modello utilizzato.
- Consistente in due fasi:
  1. Codifica
  2. Modifica del codice per rimozione degli errori, miglioramenti, modifiche dei requisiti
- Soddisfacente in un'epoca in cui:
  - il software era prodotto da un solo programmatore
  - spesso l'utente era la stessa persona
  - il problema da risolvere era ben compreso

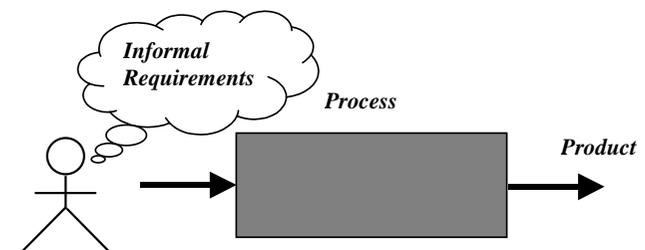
## Code & fix: problemi

- Venendo meno le sue premesse, questo modello portò a:
  - Mancata soddisfazione dei requisiti utente
  - Bassa qualità del prodotto
  - Mancato rispetto dei vincoli di tempo e costo
- Crisi del software (anni Sessanta)
- Le dimensioni economiche del fenomeno portarono alla ricerca di modelli più efficaci

## Obiettivi dei modelli di processo software (Boehm, 1988)

Determinare l'ordine degli stadi coinvolti nello sviluppo e nell'evoluzione del software e stabilire criteri di transizione per progredire da uno stadio al successivo. Questi includono criteri di completamento per lo stadio corrente e criteri per la scelta e l'ingresso nello stadio successivo.

## Processo come scatola nera

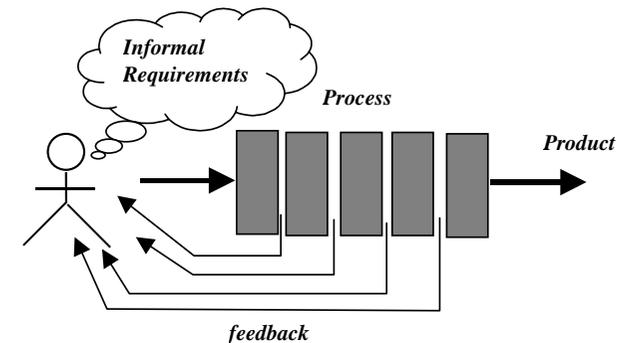


- Da requisiti informali al prodotto finale, attraverso un processo non visibile nei suoi stadi intermedi

## Problemi

- Sono visibili solo i requisiti e il prodotto finale
- I requisiti spesso non sono ben definiti
  - Il cliente non sa cosa vuole
  - Il progettista non conosce il dominio
  - I requisiti possono cambiare nel tempo
- Processo non adatto a reagire a modifiche dei requisiti
- Difficoltà della valutazione della qualità

## Processo a scatola bianca



- Gli stadi intermedi del processo sono visibili e documentati

## Vantaggi

- La trasparenza del processo permette di valutare gli artefatti intermedi durante lo sviluppo
- In seguito a una modifica dei requisiti, si può modificare il prodotto partendo dall'artefatto toccato dalla modifica
- La valutazione della qualità del prodotto è più agevole
- Aumenta la fiducia dello sviluppatore sulla qualità del prodotto