

Modelli di processo

Ciclo di vita di un prodotto sw = tutte le fasi che accompagnano tale prodotto dal concepimento dell'idea dello stesso fino al suo ritiro

Modello di processo = tentativo di organizzare il ciclo di vita del sw

- definendo le attività coinvolte nella produzione del sw
- ordinando tali attività e le loro relazioni (stabilendo cioè il flusso di esecuzione)

“determinare l'ordine delle fasi coinvolte nello sviluppo e nell'evoluzione del sw e stabilire i criteri di passaggio da una fase alla successiva. Ciò comprende criteri di completamento della fase corrente più criteri di ingresso nella successiva. Quindi un modello di processo affronta le seguenti domande relative a un progetto sw:

Che facciamo adesso?

Fino a quando continuiamo a farlo?”

(Boehm, 1988)

Obiettivi di un modello di processo

- introdurre disciplina
- standardizzare
- dominare la complessità
- migliorare verificabilità, manutenibilità, riusabilità, comprensibilità, produttività, visibilità e tempestività del processo
- aumentare la capacità di prevedere tempi e costi
- rendere il processo più facilmente automatizzabile
- migliorare la qualità dei prodotti

Code&Fix

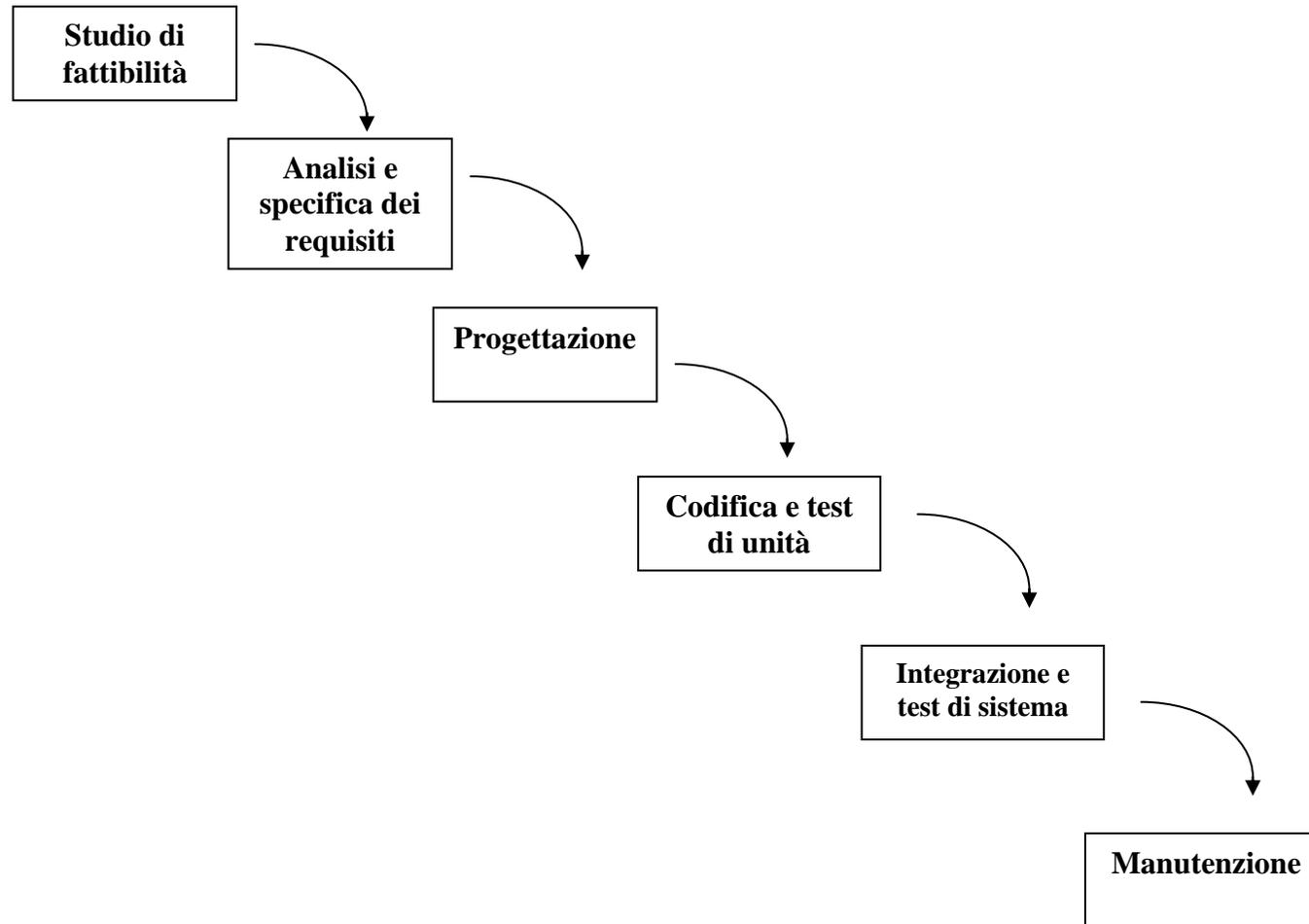
È l'approccio più antico, secondo cui

- si scrive il codice
- lo si aggiusta per eliminare gli errori che sono stati scoperti, per migliorare le funzionalità esistenti e/o per aggiungere nuove caratteristiche

Carenze:

- impossibile fare previsioni
- processo ingestibile

Modello a cascata (Royce 1970)



Modello a cascata: fasi

Inventato nei tardi anni '50 per grandi sistemi di difesa aerea, divenuto popolare negli anni '70

Fasi:

- 1) Studio di fattibilità: definizione preliminare del problema, valutazione a priori di costi e benefici; obiettivo: stabilire se lo sviluppo debba essere avviato, evidenziare le risorse disponibili per il progetto, elencare e comparare le alternative
- 2) Analisi e specifica dei requisiti: analisi completa del problema dell'utente e della sua realtà applicativa al fine di specificare le caratteristiche di qualità e i requisiti funzionali dell'applicazione (*cosa* il sistema deve fornire, non *come*). I risultati di tale analisi (spesso incompleti/inconsistenti/ambigui) devono essere sottoscritti dal committente
- 3) Progettazione: definizione dell'architettura sw

Modello a cascata: fasi (cont.)

Fasi:

- 4) Codifica e test di unità: programmazione (distinzione sfumata rispetto alla progettazione) + test per verificare il soddisfacimento delle specifiche di progetto
- 5) Integrazione e test di sistema: collaudo dell'intero sistema + (opzionalmente) alfa test (rilascio entro l'organizzazione del produttore) e beta test (rilascio a pochi utenti selezionati)
- 6) Manutenzione

Modello a cascata: pro ...

Se si adotta il modello a cascata, è necessario definire con precisione contenuti e struttura dei semilavorati e, in fase di pianificazione, le scadenze entro cui devono essere prodotti e superare i controlli di qualità

Pro:

- Le fasi indirizzano l'attività del progettista e consentono il controllo dello svolgimento del progetto
- Rimanda l'implementazione a dopo che gli obiettivi sono stati compresi

... e contro

Contro:

- Congela i requisiti
- Le fasi non sono formalmente definite, né passibili di svolgimento o controllo automatici
- I ricicli (retroazioni) sono nascosti
- È difficile raccogliere i requisiti una volta per tutte
- Una sola data di consegna
- Non include la gestione dei cambiamenti → manutenzione ed evoluzione sono di difficile previsione ed attuazione
- Stati bloccanti

Modello a cascata: rischi

Rischi:

- Individuare scelte ottimali solo per l'applicazione attuale, senza pensare all'evoluzione futura dell'applicazione
- Non pianificare l'attività di manutenzione ed eseguirla solo sul codice, non sugli altri semilavorati

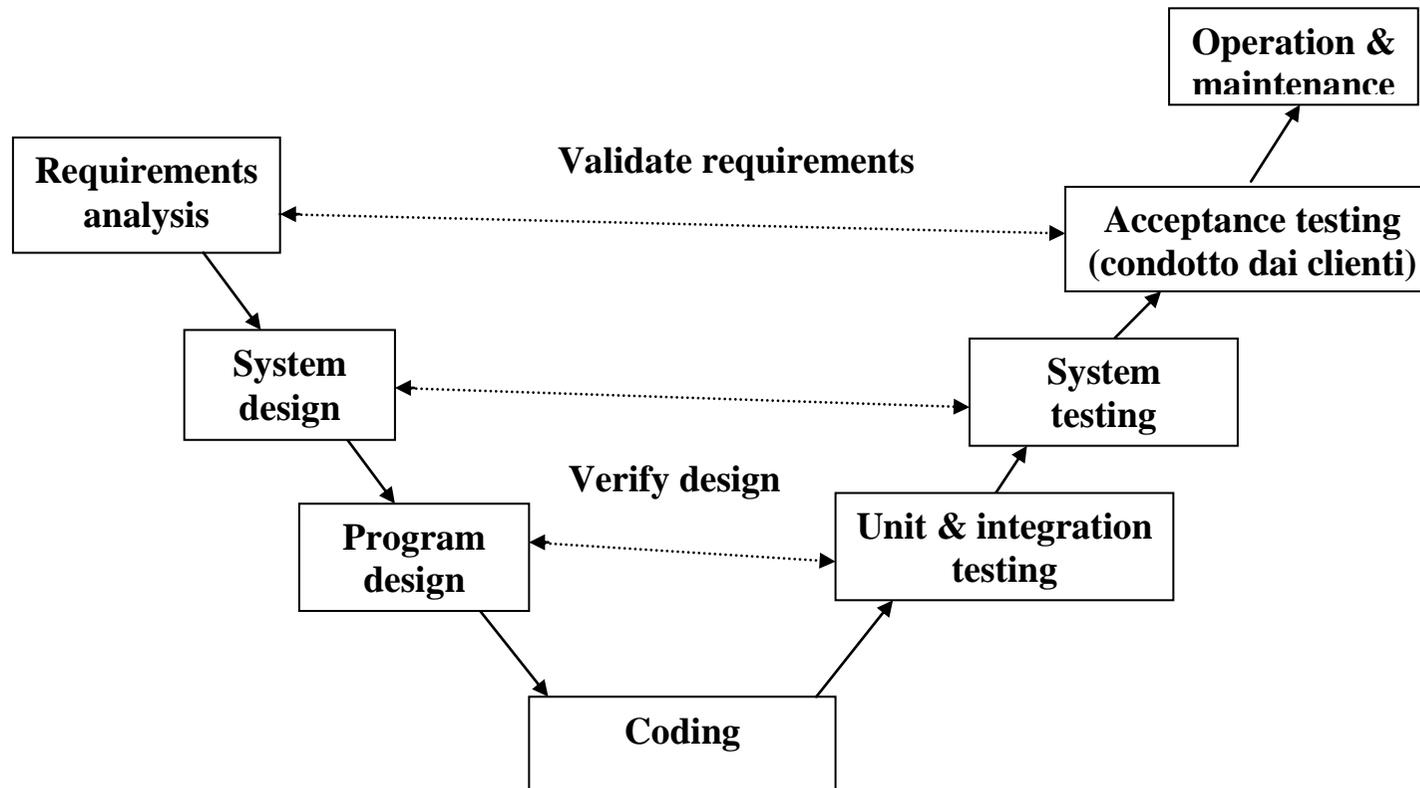
Tutto ciò è miope e si paga a caro prezzo →

Reingegnerizzazione: riportare sw destrutturato e non documentato in uno stato dal quale si possa ripartire per una manutenzione sistematica

Conclusione:

se il maggiore rischio è l'affidabilità dell'applicazione, mentre i requisiti sono estremamente stabili e ben noti, è il modello più ragionevole, con rigorosi controlli per il passaggio da una fase all'altra

Modello a V (Ministero della difesa tedesco, 1992)



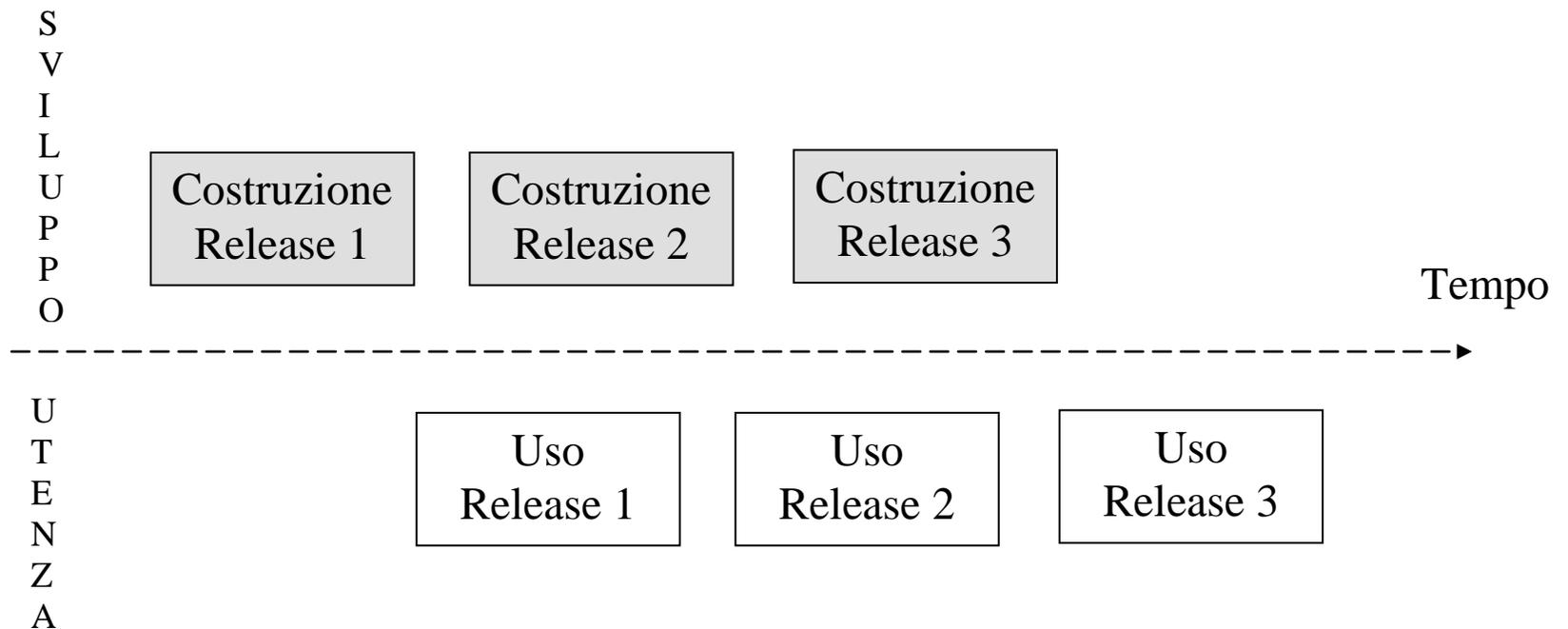
Modello a V (cont.)

- È una variante del modello a cascata che rende esplicita la necessità di effettuare iterazioni
- Mostra come le attività di testing siano collegate ad analisi dei requisiti e progetto
- I test di unità e di integrazione, oltre a occuparsi della correttezza dei programmi, possono essere usati per assicurarsi che tutti gli aspetti del progetto del programma siano stati implementati correttamente
- Il test di accettazione convalida l'aderenza ai requisiti associando un passo del test a ciascun elemento della specifica
- I problemi scoperti sul lato destro della V determinano la riesecuzione di attività sul lato sinistro

Modelli incrementali / iterativi (detti anche evolutivi)

Sono una risposta alla necessaria evoluzione del sw, alternativa interessante soprattutto quando i requisiti sono imperfetti o instabili: il sistema evolve man mano che i requisiti vengono progressivamente compresi

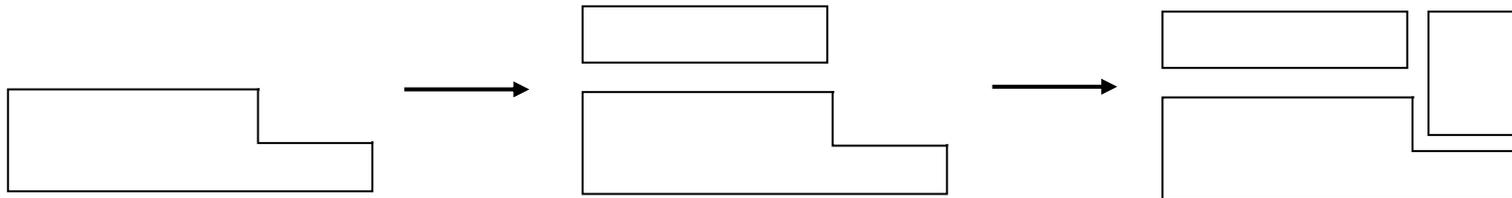
Sviluppo a stadi: mentre è operativa la release n del sistema, si lavora alla release $n+1$



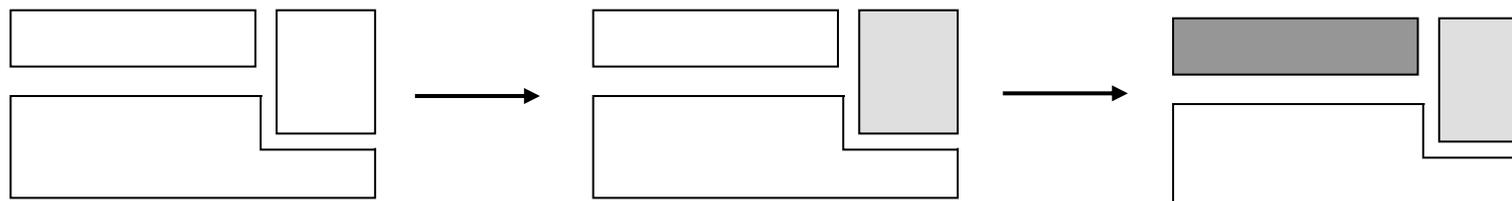
Modelli incrementali / iterativi (cont.)

Lo sviluppo a stadi è supportato da due approcci:

- sviluppo incrementale



- sviluppo iterativo

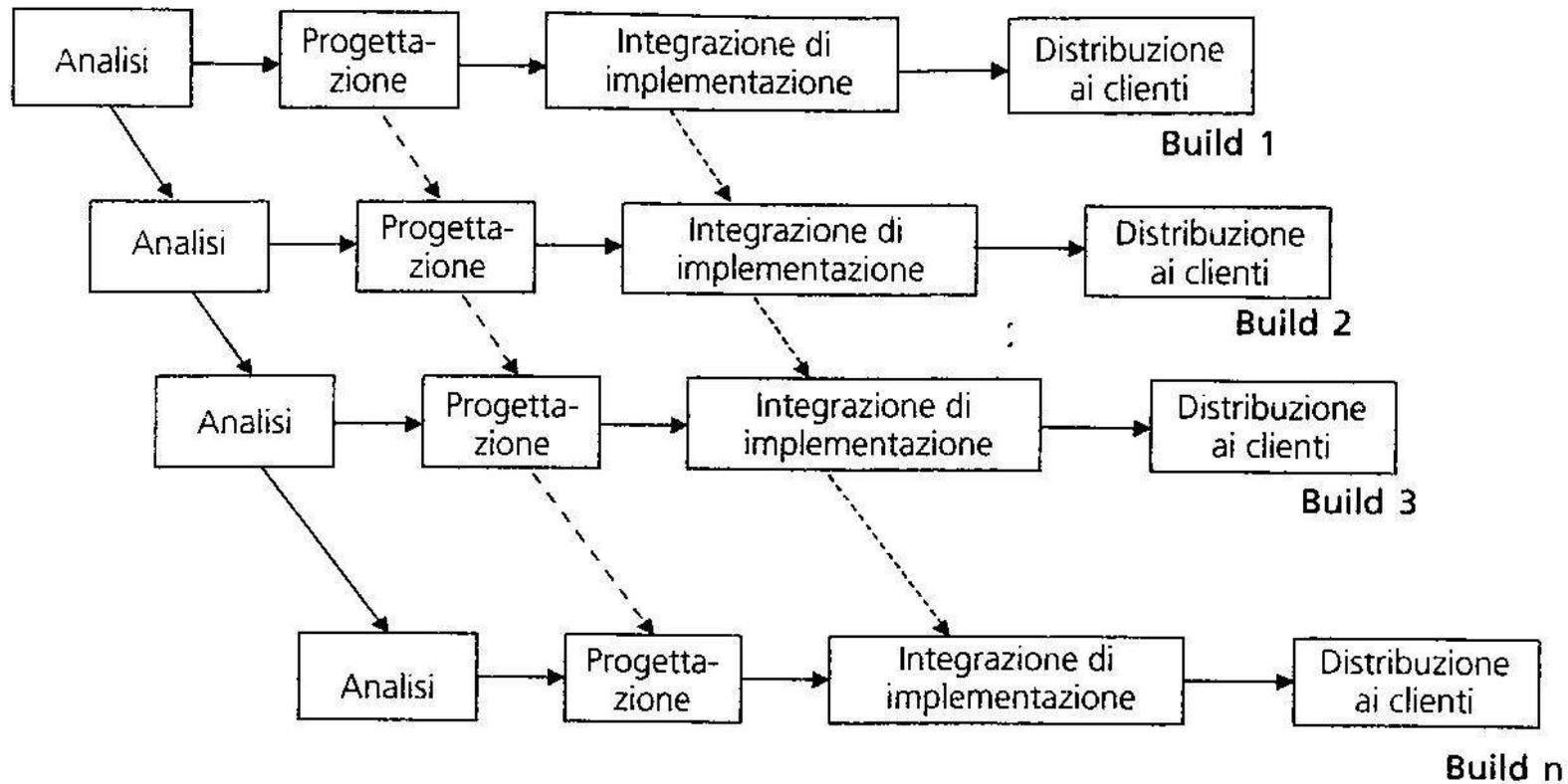


Modelli incrementali / iterativi (cont.)

- Le funzionalità individuate nell'analisi dei requisiti sono allocate a iterazioni diverse; ad ogni release segue la consegna di una nuova versione operativa con qualità e/o funzionalità aumentate
- Dall'incremento corrente si devono trarre indicazioni su come effettuare il successivo attraverso modifiche semplici e affidabili → necessità di metodi di progettazione opportuni
- Uso della prototipazione
Prototipo (= modello operativo dell'applicazione)
 - ✓ Usa e getta (ad es.
 - da mostrare al (potenziale) committente (prototipo dimostrativo)
 - per migliorare la comprensione da parte degli sviluppatori
 - per la convalida dei requisiti da parte di utenti/clienti)
 - ✓ Evolutivo (= primo incremento)

Conclusione: sono i modelli preferibili quando i rischi maggiori risiedono nell'instabilità e incertezza dei requisiti

Ingegneria concorrente



Modello a spirale (cont.)

- È un meta-modello dei processi sw
- Dati gli obiettivi di ciascuna iterazione, se ne valutano in dettaglio i rischi e si decide di conseguenza che alternativa adottare
- Quattro volute i cui prodotti sono (dalla più interna alla più esterna):
 - ✓ concept of operation (descrizione di alto livello di come il sistema dovrebbe funzionare)
 - ✓ requisiti
 - ✓ progetto
 - ✓ codifica e testing
- In ogni voluta è previsto l'uso di prototipi per valutare la fattibilità o desiderabilità di un'alternativa
- Raggio della spirale = costo accumulato nel progetto