

Corso di Laurea in Informatica - Ingegneria del Software 2 1



Ingegneria del Software 2

5. Metodiche standard di sviluppo industriale

Docente: Tullio Vardanega
tullio.vardanega@math.unipd.it

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 2



Gradi di libertà

- ◆ Segmento di ciclo di vita
 - ◆ Determinato dall'intervallo temporale tra la prima e l'ultima revisione *esterna*
 - ◆ Nel nostro caso, l'intervallo tra **RRS** (termine gara d'appalto) ed **RA** (fine del corso)
- ◆ Modello di ciclo di vita *interno*
 - ◆ Adottato dal fornitore *entro* tale segmento
 - ◆ Determina il piano d'uso delle risorse disponibili
 - ◆ Persone, capacità, strumenti

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 3



Pianificazione - 1

- ◆ Relazione tra compiti, persone e sforzo richiesto
 - ◆ *The Mythical Man-Month*, Frederick P Brooks, Jr (1975)
- ◆ Analisi delle componenti *non riducibili*
 - ◆ Compiti non partizionabili
 - ◆ Per necessità: p.es., un solo ambiente di test
 - ◆ Per scelta: p.es., per preservare *integrità concettuale*
 - ◆ Compiti che richiedono comunicazione ed interazione tra membri del progetto
 - ◆ Ri-coordinarsi sistematicamente costa molto sforzo
 - ◆ Verifiche a livello sistema
 - ◆ Il sistema è uno ed è disponibile solo alla fine dello sviluppo

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 4



Pianificazione - 2

- ◆ Aggiungere persone ad un gruppo di progetto
 - ◆ Aggiunge complessità
 - ◆ A tutti i compiti (*task*) che richiedono comunicazione ed interazione
 - ◆ Aggiunge inefficienza
 - ◆ A tutti i compiti non partizionabili
- ◆ Una buona pianificazione (comprensiva di analisi e disegno di sistema)
 - ◆ Costa sforzo a monte
 - ◆ Quando c'è tempo
 - ◆ Risparmia sforzo a valle
 - ◆ Quando le risorse cominciano a scarseggiare

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 5



Modelli di ciclo di vita interno - 1

- ◆ Non tutti i modelli si adattano allo stesso modo agli adempimenti esterni richiesti dal progetto
- ◆ La scelta interna è libera, ma comporta oneri variabili
- ◆ La scelta è spesso per prodotto (per progetto)
 - ◆ *Indipendente* dall'organizzazione di appartenenza

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

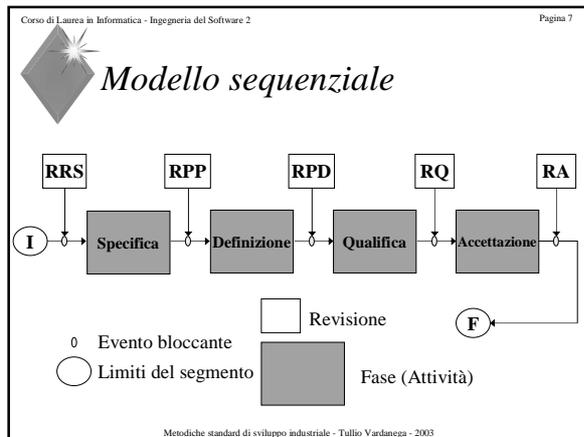
Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 6



Modelli di ciclo di vita interno - 2

- ◆ Sequenziale
 - ◆ **Vantaggi**
 - ◆ Impone disciplina al progetto
 - ◆ Comporta verifica sul completamento di ogni fase
 - ◆ **Svantaggi**
 - ◆ Richiede un notevole sforzo di documentazione
 - ◆ Allontana la percezione del prodotto (analisi e disegno) dalla sua realizzazione (codifica e qualifica)
 - ◆ **Si presta bene a**
 - ◆ Progetti a rischio contenuto, con poche dipendenze dall'esterno e limitati impatti sull'esterno

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

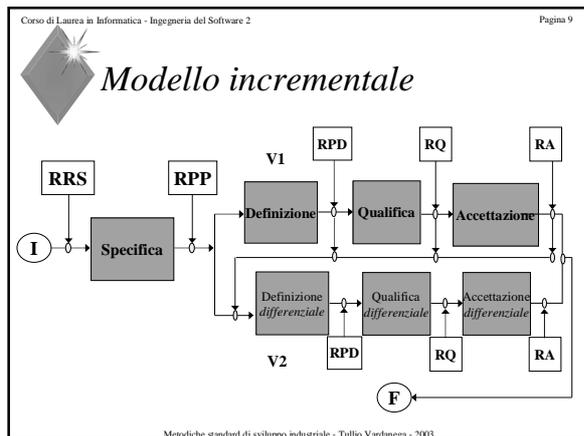


Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 8

Modelli di ciclo di vita interno - 3

- ◆ **Incrementale**
 - ◆ **Vantaggi**
 - ◆ Agevola il dialogo cliente-fornitore approssimando progressivamente il prodotto finale
 - ◆ Tollera una limitata fluttuazione (e maturazione) di requisiti
 - ◆ **Svantaggi**
 - ◆ Degenera facilmente nel costoso *circolo vizioso* "rilascia-ed-aggiusta" di versioni preliminari e premature
 - ◆ Difficile decidere *quante* versioni intermedie sono utili o necessarie
 - ◆ **Si presta bene a**
 - ◆ Progetti nei quali l'integrazione del prodotto finale ha elementi di incertezza tecnica e finanziaria

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

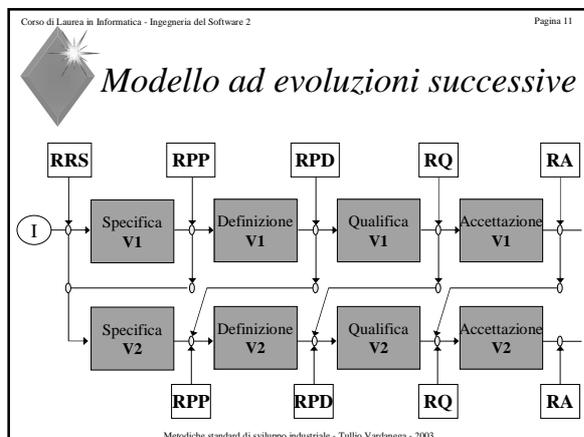


Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 10

Modelli di ciclo di vita interno - 4

- ◆ **Per evoluzioni successive**
 - ◆ **Vantaggi**
 - ◆ Quelli del modello incrementale, ma in un contesto *più formale*
 - ◆ **Svantaggi**
 - ◆ Aggiornamento dei documenti di specifica e definizione (SRS, ST, DP) per *ogni* versione successiva
 - ◆ **Si presta bene a**
 - ◆ Progetti nei quali la definizione dei bisogni è particolarmente laboriosa

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003



Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 12

Modelli di ciclo di vita interno - 5

- ◆ **A spirale**
 - ◆ **Vantaggi**
 - ◆ Favorisce l'esplorazione di alternative e promuove pratiche di riuso
 - ◆ **Svantaggi**
 - ◆ L'analisi delle alternative e dei rischi ed il contenimento dei costi di prototipazione richiedono esperienza
 - ◆ **Si presta bene a**
 - ◆ Progetti nei quali l'analisi dei rischi beneficia e coinvolge anche il cliente

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 13



Disegno software - 1

- ◆ Comprende *specificazione e definizione* del prodotto
 - ◆ Definizione IEEE
 - ◆ Il *processo* di definizione dell'architettura, delle componenti, delle interfacce e delle altre caratteristiche di un sistema o di una sua componente
 - ◆ Il *prodotto* di tale processo

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 14



Disegno software - 2

- ◆ Processo interno al processo di sviluppo
 - ◆ Procede dall'analisi dei requisiti
 - ◆ Produce una descrizione della *struttura interna* e dell'*organizzazione* del sistema
 - ◆ Fornisce la base della realizzazione
 - ◆ Architettura software
 - ◆ Decomposizione ed organizzazione del sistema in componenti ed interfacce tra esse
 - ◆ Il livello di dettaglio deve essere sufficiente a guidarne la realizzazione

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 15



Disegno software - 3

- ◆ Attività 1 : Disegno architetturale
 - ◆ Definisce la struttura e l'organizzazione del sistema secondo una visione *ad alto livello*
 - ◆ Identifica le componenti
 - ◆ Entità *funzionalmente coese* e suscettibili di implementazione mediante ulteriore *decomposizione*
- ◆ Attività 2 : Disegno di dettaglio
 - ◆ Ciascuna componente è descritta ad un livello sufficiente per *determinarne* la codifica

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 16



Nozioni fondamentali

- ◆ Limiti *intrinseci* del disegno
 - ◆ Non tutti i problemi hanno soluzione
 - ◆ Occorre comprendere
 - ◆ Obiettivi
 - ◆ Vincoli
 - ◆ Alternative
 - ◆ Rappresentazioni del problema e delle sue soluzioni
- ◆ *Contesto* del disegno
 - ◆ Fattibilità e verificabilità
 - ◆ Tecnica ed economica

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 17



Tecniche abilitanti - 1

- ◆ Astrazione
 - ◆ Dimenticare informazione (attributi specifici) per applicare operazioni *uguali* ad entità *diverse*
 - ◆ P.es., calcolare l'area di una figura piana
 - ◆ La base di una gerarchia di classi *astraerispetto* alle classi più specializzate
 - ◆ Ad una astrazione corrisponde una *concretizzazione*
 - ◆ Mediante parametri
 - ◆ Una *generic unit* in Ada; un *template* in C++
 - ◆ Mediante specializzazione di specifica
 - ◆ Una *classe* in Java

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 18



Tecniche abilitanti - 2

- ◆ Grado di accoppiamento e di coesione
 - ◆ L'accoppiamento è misura dell'intensità della relazione *tra moduli* (inter)
 - ◆ La modifica di uno comporta la modifica dell'altro
 - ◆ Indesiderabile effetto domino
 - ◆ Forte accoppiamento segnala scarsa *modularità*
 - ◆ La coesione è misura dell'intensità della relazione *tra i costituenti di un modulo* (intra)
 - ◆ Forte coesione denota buona *caratterizzazione*

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 19



Tecniche abilitanti - 3

- ◆ Decomposizione e modularità
 - ◆ Una buona decomposizione architetturale identifica componenti tra loro indipendenti
 - ◆ Disaccoppiati
 - ◆ Autosufficienti (funzionalmente coesi)
- ◆ Incapsulazione (*information hiding*)
 - ◆ Separare l'astrazione dal dettaglio implementativo
 - ◆ L'una è *pubblica* (specifica di interfaccia)
 - ◆ L'altro è noto solo all'autore

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 20



Tecniche abilitanti - 4

- ◆ Sufficienza
 - ◆ La definizione dell'astrazione è sufficiente a caratterizzare l'entità desiderata
- ◆ Completezza
 - ◆ L'astrazione esibisce tutte le caratteristiche richieste
- ◆ Atomicità
 - ◆ La definizione dell'astrazione non può essere convenientemente decomposta in astrazioni più primitive

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 21



Aspetti cruciali - 1

- ◆ Concorrenza
 - ◆ Se e come decomporre il sistema in entità attive concorrenti (processo, *task*, *thread*) assicurando
 - ◆ Efficienza di esecuzione
 - ◆ Atomicità di azione
 - ◆ Integrità di dati condivisi
 - ◆ Comunicazione e sincronizzazione
 - ◆ Predicibilità di ordinamento (*scheduling*)

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 22



Aspetti cruciali - 2

- ◆ Controllo e gestione degli eventi
 - ◆ **Evento**
 - ◆ Relativo al *flusso dei dati*
 - ◆ La disponibilità di un dato (dall'interno o dall'esterno)
 - ◆ Relativo al *flusso di controllo*
 - ◆ L'ingresso del sistema (o di una sua componente) in un particolare stato
 - ◆ Relativo al *trascorrere del tempo*
 - ◆ Come organizzare il flusso di dati e di controllo
 - ◆ Come trattare gli eventi temporali

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 23



Aspetti cruciali - 3

- ◆ Distribuzione
 - ◆ Se e come componenti software sono disseminate su più nodi di elaborazione
 - ◆ Come tali componenti comunicano fra loro
- ◆ Gestione degli errori e delle eccezioni
 - ◆ Come prevenire, trattare e tollerare eventi anomali (guasti, difetti interni, errori d'uso)

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 24



Integrità concettuale - 1

- ◆ Facilmente riconoscibile in una architettura fisica (p.es., un edificio)
 - ◆ Suggestisce uno stile uniforme, coerentemente applicato a tutte le *parti* del sistema ed alle loro *interazioni*
 - ◆ Bilancia capacità funzionale con semplicità d'uso e di concezione
 - ◆ L'una cresce senza penalizzare le altre
- ◆ Desiderabile in *ogni* architettura di sistema

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003

Corso di Laurea in Informatica - Ingegneria del Software 2 Pagina 25



Integrità concettuale - 2

- ◆ **Procede da una definizione unitaria**
 - ◆ *Non unilaterale*, perché passa al vaglio dei membri del progetto
 - ◆ Richiede *osservanza* (ai costruttori) e *vigilanza* (all'architetto)
 - ◆ Nozione *aristocratica* piuttosto che democratica
- ◆ **È separata dall'implementazione**
 - ◆ Consente più percorsi implementativi
 - ◆ Permette parallelismo con l'implementazione

Metodiche standard di sviluppo industriale - Tullio Vardanega - 2003