

Il ciclo di vita del software




Il ciclo di vita del software

IS 2001-5
Corso di Ingegneria del Software
V. Ambriola, G.A. Cignoni,
C. Montanero, L. Semini
Con aggiornamenti di: T. Vardanega (UniPD)



Dipartimento di Informatica, Università di Pisa 1/24



Il ciclo di vita del software

Contenuti

- Il ciclo di vita del software
- Il modello a spirale
- Il modello a cascata
- I modelli iterativi
- Seminario: i cicli di vita in ISO/IEC 12207

Dipartimento di Informatica, Università di Pisa 2/24




Il ciclo di vita del software

Il concetto di ciclo di vita

- Concezione → sviluppo → utilizzo → ritiro
- Identificazione delle attività
 - Modelli generici, indipendenti dal prodotto
 - Decomposizione delle fasi in attività (ricerca di coesione)
 - Adozione di terminologia consistente
- Organizzazione delle attività
 - Ordinamento delle attività
 - Definizione di criteri di completamento e di avanzamento

Dipartimento di Informatica, Università di Pisa 3/24




Il ciclo di vita del software

Obiettivi e limiti

- Propedeutico al processo definito
- Richiede controllo di qualità
 - Definizione del processo produttivo (standard aziendale)
 - Identificazione di attività, obiettivi, dipendenze
- Agevola la pianificazione e la gestione dei progetti
- È indipendente da metodi e strumenti di sviluppo software

Dipartimento di Informatica, Università di Pisa 4/24




Il ciclo di vita del software

Evoluzione dei modelli

- *Code-'n-Fix*: un "non-modello"
 - Consiste di attività casuali non organizzate
 - Dà luogo a progetti caotici non gestiti né gestibili
- Modelli organizzati
 - *Cascata* a rigide fasi sequenziali oppure sue varianti con prototipi e ritorni
 - *Incrementali* realizzazione in più passi
 - *Evolutivi* modelli ciclici con ripetute iterazioni interne
 - *Spirale* contesto allargato e modello astratto

Dipartimento di Informatica, Università di Pisa 5/24



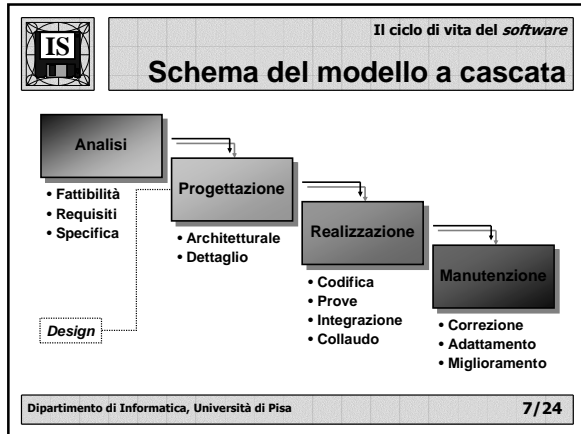
Il ciclo di vita del software

(1) Il modello a cascata

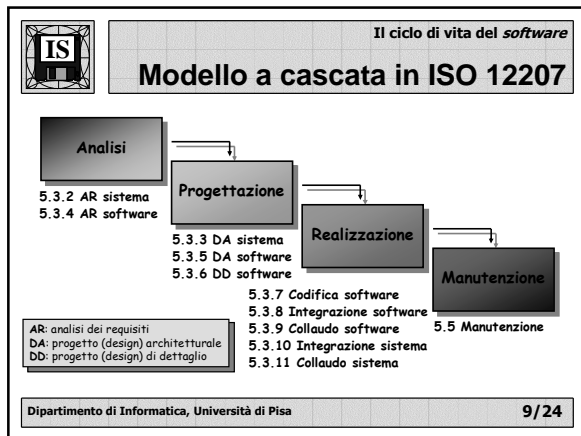
- Definito nel 1970 da Winston Royce
 - "Managing the development of large software systems: concepts and techniques"
- Una successione di fasi sequenziali
 - Non consente ritorno a fasi precedenti
 - Eventi eccezionali fanno ripartire dall'inizio
- Documentazione
 - Ogni fase produce "documenti" che la concretizzano
 - Emissione ed approvazione di documenti sono condizione necessaria per l'avvio della fase successiva
 - Modello "document-driven"

Dipartimento di Informatica, Università di Pisa 6/24

Il ciclo di vita del software



- Il ciclo di vita del *software*
- ## Caratteristiche delle fasi
- Le fasi sono descritte in termini di:
 - Attività previste e prodotti attesi
 - Contenuti e struttura dei documenti
 - Responsabilità e ruoli coinvolti
 - Scadenze di consegna dei documenti
 - Presentano dipendenze causali e temporali
 - Ciascuna fase determina le proprie attività
- Dipartimento di Informatica, Università di Pisa 8/24



- Il ciclo di vita del *software*
- ## Varianti del modello a cascata
- Difetto principale: eccessiva rigidità
 - Stretta sequenzialità tra fasi
 - Non ammette modifiche nei requisiti in corso d'opera
 - Richiede molta manutenzione
 - Esprime una visione burocratica e poco realistica
 - Variante 1: cascata con prototipazione
 - Prototipo "usa-e-getta", solo per capire meglio i requisiti
 - Variante 2: cascata con ritorni
 - Entrambe raggruppano sotto-sequenze di fasi
- Dipartimento di Informatica, Università di Pisa 10/24

- Il ciclo di vita del *software*
- ## Modelli iterativi
- Applicabili a qualsunque modello generale di ciclo di vita
 - Necessità di adattamento ai cambiamenti
 - Evoluzione di problemi, soluzioni possibili e tecnologie utilizzabili
 - Diversificazione dei requisiti del committente
 - Soluzione generale
 - Decomporre la realizzazione del sistema
 - Differire la realizzazione delle componenti critiche
 - Le iterazioni devono essere pianificate
- Dipartimento di Informatica, Università di Pisa 11/24

- Il ciclo di vita del *software*
- ## (2) Modello incrementale
- Analisi e progettazione sono definitive
 - I requisiti sono identificati completamente
 - L'architettura del sistema è identificata e fissata
 - I passi della realizzazione incrementale sono pianificati
 - La realizzazione è incrementale
 - Progettazione di dettaglio, codifica e prove
 - Integrazione e collaudo
 - Accettazione e rilascio di versioni parziali
 - Dai requisiti essenziali (prima) a quelli desiderabili (poi)
- Dipartimento di Informatica, Università di Pisa 12/24

Il ciclo di vita del software

Il ciclo di vita del software

Modello incrementale in ISO 12207

Analisi e Progettazione

- 5.3.1 Istanziamento del processo
- 5.3.2 AR sistema
- 5.3.3 DA sistema
- 5.3.4 AR software
- 5.3.5 DA software

Progettazione di dettaglio

- 5.3.6 DB software

Realizzazione

- 5.3.7 Codifica software
- 5.3.8 Integrazione software
- 5.3.10 Integrazione sistema

Accettazione di versione

- 5.3.9 Collaudo software
- 5.3.11 Collaudo sistema

Dipartimento di Informatica, Università di Pisa 13/24

Il ciclo di vita del software

(3) Modello evolutivo

- Analisi preliminare**
 - Per identificare i requisiti di massima
 - Per definire l'architettura di massima
 - Per pianificare i passi di analisi e realizzazione evolutiva
- Analisi e realizzazione di evoluzione**
 - Per raffinamento ed estensione dell'analisi
 - Per progettazione, codifica, prove e integrazione
- Rilascio di prototipi, accettazione finale**

Dipartimento di Informatica, Università di Pisa 14/24

Il ciclo di vita del software

Modello evolutivo in ISO 12207

Analisi preliminare

- 5.3.1 Istanziamento del processo
- 5.3.2 AR sistema
- 5.3.3 DA sistema
- 5.3.4 AR software
- 5.3.5 DA software

Analisi e Progettazione

- 5.3.3 DA sistema
- 5.3.4 AR software
- 5.3.5 DA software
- 5.3.6 DD software

Realizzazione

- 5.3.7 Codifica software
- 5.3.8 Integrazione software
- 5.3.10 Integrazione sistema

Prototipo oppure Accettazione finale

- 6.6 Revisione oppure
- 5.3.9 Collaudo software
- 5.3.11 Collaudo sistema

Il prototipo diventa visibile al committente

Dipartimento di Informatica, Università di Pisa 15/24

Il ciclo di vita del software

(4) Il modello a spirale

- Proposto da Barry W Boehm nel 1988**
- Prevede quattro attività principali**
 - Definizione degli obiettivi
 - Analisi dei rischi
 - Sviluppo e validazione
 - Pianificazione
- È un modello astratto: va specializzato**

Dipartimento di Informatica, Università di Pisa 16/24

Il ciclo di vita del software

Fasi del modello a spirale

- Definizione degli obiettivi**
 - Requisiti, identificazione dei rischi, piano di gestione
- Analisi dei rischi**
 - Studio delle conseguenze, valutazione delle alternative, con l'ausilio di prototipi e simulazioni
- Sviluppo e validazione**
 - Realizzazione del prodotto
- Pianificazione**
 - Decisione circa il proseguimento, pianificazione del ciclo di vita

Dipartimento di Informatica, Università di Pisa 17/24

Il ciclo di vita del software

Schema del modello a spirale

Dipartimento di Informatica, Università di Pisa 18/24

Il ciclo di vita del software

Il ciclo di vita del *software*

Aspetti del modello a spirale

- ❑ **Pone grande attenzione sugli aspetti gestionali**
 - Pianificazione delle fasi
 - Analisi dei rischi (modello "risk-driven")
- ❑ **Richiede forte interazione tra committente e fornitore**
 - Committente: definizioni degli obiettivi, pianificazione
 - Fornitore: sviluppo e validazione
 - Entrambi: analisi dei rischi
- ❑ **Applicabile ai cicli tradizionali**

Dipartimento di Informatica, Università di Pisa 19/24

Il ciclo di vita del *software*

Altri modelli

- ❑ **Extreme Programming (XP)**
<http://www.extremeprogramming.org>
 - Facilita il coinvolgimento (informale) del committente
 - Non solo nell'analisi, ma anche in progettazione e sviluppo (!)
 - Fortemente incrementali (a passo piccolo)
 - Tende (colpevolmente) a perdere rigore
- ❑ **Component-based**
 - Massima attenzione al riuso sistematico di componenti preesistenti proprie od "off-the-shelf"

Dipartimento di Informatica, Università di Pisa 20/24

Il ciclo di vita del *software*

Modello "component-based"

```

    graph TD
      A(Analisi dei Requisiti) --> B(Analisi delle Componenti)
      B --> C(Adattamento dei Requisiti)
      C --> D(Sviluppo ed Integrazione)
      D --> E(Validazione di Sistema)
      E --> F(Progettazione con Riuso)
      F --> A
  
```

Dipartimento di Informatica, Università di Pisa 21/24

Il ciclo di vita del *software*

Ripartizione dei costi sui modelli

- ❑ **In genere, il 60% dei costi va allo sviluppo ed il 40% alla qualifica**
- ❑ **I costi complessivi variano al variare del dominio e del tipo di sistema**
- ❑ **La ripartizione dei costi sulle fasi varia al variare del modello e del dominio**
 - Sistemi critici: 60% qualifica

Tratto da: Ian Sommerville, Software Engineering, 7th ed.

Dipartimento di Informatica, Università di Pisa 22/24

Il ciclo di vita del *software*

Riepilogo

- ❑ Il ciclo di vita del software
- ❑ Il modello a cascata
- ❑ I modelli iterativi
- ❑ Il modello a spirale
- ❑ Cicli di vita espressi mediante ISO/IEC 12207

Dipartimento di Informatica, Università di Pisa 23/24

Il ciclo di vita del *software*

Riferimenti

- ❑ B.W. Bohem, "A spiral model of software development and enhancement", IEEE Software, maggio 1998
- ❑ Center for Software Engineering, http://sunset.usc.edu/research/spiral_model/
- ❑ W.W. Royce, "Managing the development of large software systems: concepts and techniques", Atti della conferenza "Wescon '70", agosto 1970
- ❑ ISO/IEC TR 15271:1998, Information Technology - Guide for ISO/IEC 12207

Dipartimento di Informatica, Università di Pisa 24/24