

Ingegneria del Software: Concetti introduttivi

a cura di **Angelo Furfaro**

da “Ingegneria del Software, Fondamenti e Principi”
Ghezzi, Jazayeri, Mandrioli

Dipartimento di
Ingegneria Informatica, Elettronica, Modellistica e Sistemistica
Università della Calabria, 87036 Rende(CS) - Italy
Email: a.furfaro@unical.it
Web: <http://angelo.furfaro.dimes.unical.it>

Introduzione

- L'ingegneria del software si occupa della creazione di sistemi software complessi
- La loro realizzazione richiede il lavoro di uno o più squadre (team) di ingegneri
- Tali sistemi rimangono in servizio per molti anni e durante la loro *vita* subiscono numerose modifiche per:
 - eliminare difetti
 - migliorare alcune caratteristiche
 - introdurre nuove funzionalità
 - eliminare caratteristiche obsolete
 - adattarli ad operare in ambienti differenti

Definizioni di Ingegneria del Software

- Applicazione di un approccio sistematico, disciplinato e quantificabile nello sviluppo, funzionamento e manutenzione del software. [*IEEE 610.12-1990 Glossario standard della terminologia dell'ingegneria del software*]
- Creazione di software multiversione da parte di più operatori. [*Parnas, 1978*]

Programmazione e ingegneria del software

- La definizione di Parnas mette in luce le differenze tra programmazione e ingegneria del software
- Un programmatore scrive un programma completo
- Un ingegnere progetta e sviluppa un componente software che dovrà interagire con altri componenti scritti da altri ingegneri
- Un componente può essere modificato da altri ingegneri ed utilizzato per realizzare differenti versioni di un sistema o sistemi differenti
- La programmazione è un'attività individuale, l'ingegneria del software è un'attività di gruppo

Strumenti dell'ingegneria del software

- L'ingegneria del software ha cominciato ad assumere una forma ben definita verso la fine degli anni settanta
- Le difficoltà nella realizzazione di sistemi complessi (ad esempio sistemi operativi) hanno fatto nascere l'esigenza di un approccio ingegneristico alla produzione del software
- Nelle discipline ingegneristiche classiche, l'ingegnere ha a disposizione strumenti e competenze matematiche per descrivere le caratteristiche del prodotto distintamente da quelle del progetto
- Nell'ingegneria del software tali strumenti non sono ancora ben sviluppati: l'esperienza personale ha più peso rispetto alle tecniche matematiche.
- Tuttavia strumenti di analisi formale sono essenziali nella pratica

Il ruolo dell'ingegnere del software

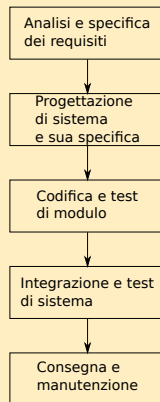
L'ingegnere informatico deve:

- essere un buon programmatore, con esperienza in uno o più linguaggi di programmazione e conoscere bene gli algoritmi e le strutture dati
- avere familiarità con più approcci di progettazione
- essere in grado di tradurre richieste vaghe in specifiche precise
- essere in grado di interagire con l'utente
- sapersi muovere attraverso i differenti livelli di astrazione nei diversi stadi del progetto
- saper costruire vari modelli e ragionare su di essi al fine di operare opportune scelte di compromesso (*trade-off*) nel processo di sviluppo
- avere capacità comunicative ed interpersonali che gli consentano di operare all'interno di un gruppo di lavoro

Il ciclo di vita del software

- Il software subisce uno sviluppo dall'idea iniziale di un possibile prodotto fino all'implementazione ed alla consegna al cliente (ed in seguito durante la sua manutenzione)
- Il ciclo di vita del software è costituito da varie fasi
- Nel modello tradizionale, detto “a cascata” (*waterfall*), ogni fase ha un inizio ed una fine ben definiti
- Ogni fase termina con la produzione di un artefatto che viene trasferito alla fase successiva

Modello Waterfall



Fasi del ciclo di vita

Analisi e specifica dei requisiti

- È la prima fase di un progetto per lo sviluppo di un software
- Viene effettuata a valle di uno studio di fattibilità per definire i costi ed i benefici della realizzazione del sistema
- Ha lo scopo di identificare e documentare i requisiti del sistema
- A tale fase partecipano: il cliente, specialisti di analisi di mercato, membri del team di sviluppo
- Nel caso in cui i requisiti non sono chiari, sono necessarie più interazioni tra cliente e sviluppatore
- I requisiti dovrebbero essere espressi in una notazione che sia al contempo comprensibile dall'utente finale e rigorosa nella loro formalizzazione
- Alcune metodologie ritengono che in questa fase debbano essere progettati i test a cui sottoporre il sistema finale

Progettazione di sistema e sua specifica

- Una volta definiti i requisiti, occorre progettare un sistema software che li soddisfi
- Questa fase è divisa in due sottofasi:
 - **Progettazione architetturale**
 - **Progettazione di dettaglio**
- Il progetto architetturale definisce l'organizzazione del sistema in termini di componenti di alto livello e delle loro interazioni
- I componenti sono quindi scomposti in moduli di basso livello con interfacce definite in modo accurato
- Ciascun livello di progettazione è descritto in opportuno documento di specifica in cui sono motivate le scelte progettuali effettuate

Fasi del ciclo di vita

Codifica e test di modulo

- In questa fase si produce il codice che implementa ciò che è stato progettato nella fase precedente
- Da questo codice si ottiene un sistema funzionante
- Anche per lo sviluppo di test occorre ottenere del codice, ma questo di solito è utilizzato solo dagli sviluppatori
- I singoli moduli creati in questa fase sono testati prima di passare a quella successiva

Integrazione e test di sistema

- Tutti i moduli sviluppati nella fase precedente sono assemblati, integrati e testati come un unico sistema

Consegna e manutenzione

- Il sistema viene consegnato al cliente ed entra nella fase di manutenzione in cui vengono incorporate le modifiche apportate al sistema dopo la prima consegna