

▪ **Lezione 2 - (10 / 10 / 2024)**

Costruire un Prodotto Software ha un costo. Questo costo lo si può applicare riguardo dimensione (size), repliche e ampiezza di mercato.

Per fare due prodotti di dimensione $S/2$ (due mezzi prodotti) costa meno che farne uno di dimensione S , poiché il costo della dimensione verrà calcolato come $C = a S^2$

$$\left(\frac{S}{2}\right)^2 + \left(\frac{S}{2}\right)^2 = \left(\frac{S}{2}\right)^2$$

Immaginiamo che realizzi un prodotto Software, che mi è costato x , produrre una replica di esso costa 0. Infatti per i Prodotti Software produrre una replica non costa niente, invece, per i Prodotti Hardware invece ci sta un costo di replica.

Un prodotto di dimensione (size) doppia richiede un prezzo quattro volte superiore o un mercato quattro volte più ampio.

Introduciamo alcuni termini:

- Con la dicitura *Prodotto Software* intendiamo non soltanto il prodotto finale di un codice, ma tutta la documentazione che viene prodotta dall'inizio del ciclo di vita del software (Documento di specifica + Codice).
- Un *Artefatto* può essere un documento requisiti, un documento di specifica o un documento di progetto.
- *Codice* è il prodotto software finale
- *Sistema Software* è un insieme organizzato di prodotti Software (ES: Word, PPT, ecc...), ma anche un insieme di software e di hardware
- *Cliente*: colui che ordina o commissiona e che paga per lo sviluppo e la manutenzione del prodotto Software
- *Sviluppatore*: è il soggetto al cui viene commissionato questo Prodotto Software (solitamente è un'azienda)
- *Utente*: colui che usa il prodotto Software

Abbiamo due tipologie di Software:

- *Software a contratto*: cliente e sviluppatore sono soggetti differenti – lo sviluppatore farà il prodotto software su richiesta del cliente
- *Software Interno*: è un sottotipo del software a contratto, qui il cliente e sviluppatore coincidono – ovvero cliente e sviluppatore fanno parte della stessa organizzazione/azienda

Un aspetto importante del prodotto Software è l'affidabilità

L'affidabilità la possiamo definire attraverso una definizione informale come la credibilità del prodotto software. Possiamo definirla pure dal punto di vista formale come la probabilità che il prodotto software lavora correttamente in un intervallo temporale.

Il Funzionamento Corretto di un prodotto è legato al concetto di Guasto (*failure*), ovvero il comportamento anomalo del prodotto Software dovuto alla presenza di un difetto.

Un difetto (detto *Defect* o *Bug*) è un'anomalia presente nel prodotto software.

Questi difetti nel prodotto Software ci finiscono per Errore, ovvero un'azione errata di chi per ignoranza o distrazione introduce un difetto nel prodotto Software.

In un prodotto Software quando ho un Difetto non per forza deve verificarsi un Guasto. Un prodotto software con molti difetti è poco affidabile. Infatti l'affidabilità del prodotto migliora via via che si riduce il numero di difetti.

Tra l'affidabilità osservata e il numero di difetti latenti c'è una relazione non-semplificata.

L'eliminare difetti da parti del prodotto usate raramente, non comporta grandi effetti sull'affidabilità osservata.

Esempio: su word uso determinate funzioni di più (finestra Edit, il Correttore, la Funzione Stampa), poi esistono altre funzioni che usiamo raramente (modificare il layout inserendo colonne).

Esiste una regola 10-90 la quale evidenzia che, in programmi di grandi dimensioni, la maggior parte del tempo di esecuzione (il 90%) è spesa eseguendo una piccola porzione di istruzioni (il 10%). Questo 10% delle istruzioni, che richiede la maggior parte del tempo di calcolo, è chiamato core o nucleo del programma. Quindi, il core del programma rappresenta quella parte cruciale del codice che è più "pesante" in termini di esecuzione e quindi merita un'ottimizzazione particolare per migliorare le prestazioni complessive del software.

Il miglioramento dell'affidabilità di un software dipende dall'eliminazione dei difetti e, soprattutto, dalla loro posizione: è più significativo se i difetti si trovano nel nucleo del programma.

Tutto ciò definisce il profilo Operativo, detto anche *operational profile*.

I difetti in un prodotto Software sono unici e soggettivi, poiché non tutti gli utenti usano il Software allo stesso modo. Infatti l'affidabilità di un prodotto Software dipende dall'utente.

Confronto tra Affidabilità Software e Hardware:

- I guasti in un prodotto Software sono dovuti alla presenza di difetti nei programmi. Negli anni i software non si “consumano”. Questi difetti sono latenti / nascosti.
- I guasti in un prodotto Hardware sono quasi sempre dovuti al consumo o al deterioramento dei componenti, ciò comporta che qualche componente non si comporta più come dovrebbe o si rompe.

Una volta corretti questi difetti, dopo la riparazione:

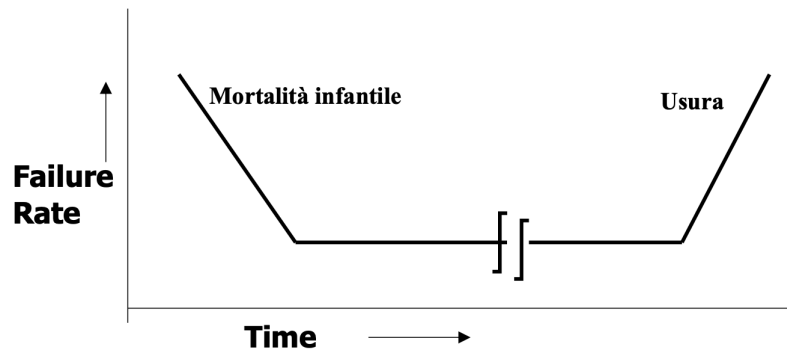
- l'affidabilità dell'Hardware torna come era prima
- l'affidabilità del Software può aumentare o diminuire

L'obiettivo dell'affidabilità per l'hardware è mantenere stabile la frequenza di guasto, mentre per il software è ridurre la frequenza di guasto nel tempo, aumentando così l'affidabilità.

Esempio:

- vogliamo che un computer portatile non inizi a rompersi di più con l'uso, ma mantenga la sua affidabilità costante.
- vogliamo che il programma diventi sempre più stabile e con meno errori man mano che viene aggiornato.

- Andamento frequenza di guasto Hardware



Possiamo definire questo andamento come “andamento vasca da bagno”. Questo andamento ci dice che:

È noto che i prodotti una volta usciti dalla fabbrica tendono ad avere una mortalità infantile. Bisognerebbe tenere la frequenza di guasto costante. Ma dopo un lungo periodo arriveremo all'usura del prodotto.

- Andamento frequenza di guasto Software

