

Misurazione del codice(13)

Misurazione: Processo che assegna numeri o simboli ad attributi di entità per descriverle secondo regole definite. **Misura:** Assegnazione empirica e oggettiva di un valore ad un'entità per caratterizzarne un attributo specifico. **Metrica:** insieme di regole per fissare entità da misurare gli attributi rilevanti, l'unità di misura. Secondo IEEE misura quantitativa del grado di possesso di uno specifico attributo da parte di un sistema, componente, processo.

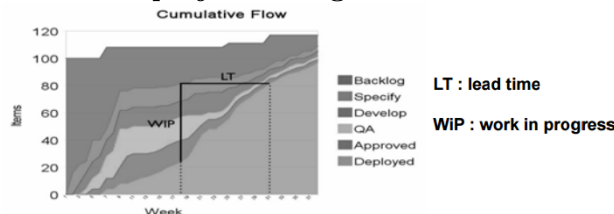
Ci sono anche gli **indicatori** che sono delle misure indirette ottenute tramite stime o precisazioni. Gli obbiettivi nella misurazione SW sono quelli di valutare lo stato di:

- Progetti, stime, preventivi e consuntivi di costo, tempo;
- Prodotti qualità;
- Processi qualità e miglioramenti;
- Risorse consumo;

ISO 15939 the information needs are based on goals, constraints, risk, and problems which originate from the technical and mgmt processes. Il CMMI suggerisce di determinarli nelle seguenti aree:

- Requirements Management, quando le pratiche in uso aderiscono a quelle di riferimento e i prodotti del progetto soddisfano i requisiti;
- Design & Implementation;
- Verification & Validation;
- Quality Assurance, come *requirements management*;
- Configuration Management, quale è l'impatto di una data modifica e qual'è il grado di integrità delle baseline di progetto;
- Project Management, quanto sono affidabili i preventivi e accurati i consuntivi, quanto sono efficaci le misure correttive e quanto sono concretamente adottate;
- Risk analysis & decision analysis, quali e quanto i rischi gravano attualmente sul progetto;
- Training.

Metrica di project management:



lead time= tempo che intercorre da quando un'azione è assegnata a quando ha avuto successo.

Metriche software:

- **SLOC:** linee di codice sorgente, viene usata per derivare informazioni di costo e produttività, è solitamente rapportata con la densità di commenti. Tuttavia dipende dalla sintassi del linguaggio e dallo stile di codifica.

- **Complessità ciclomatica:** è una metrica indipendente dal linguaggio, si svolge una misurazione astratta di un attributo significativo del software, misura la complessità del flusso di controllo e il grado di *fan-in fan-out* applicato ai dati può aiutare a stimare la complessità del flusso di dati, meglio comunque calcolarlo con strumenti automatici. L'unico difetto è che è molto onerosa da applicare prima della progettazione in dettaglio.

$$v(G) = e - n + p$$

- $v(G)$, numero di percorsi lineari in G ;
- e , numero degli archi (flusso);
- n , numero dei nodi (espressioni o comandi);
- p , numero delle componenti connesse ad ogni arco ($p=2$ per esecuzione sequenziale: 1 predecessore, 1 successore).

Le misure si dividono in due categorie, quelle interne (numero di requisiti, grado di coesione...) e quelle esterne (ISO/IEC 9126-1).

Per il software ad oggetti servono delle metriche:

- Metodi, si può usare McCabe come metrica base (*tiene conto di interfacce e variabili locali*)

$$MC = w_1 MIC + w_2 MLVC + w_3 MCC$$

- MIC, complessità dell'interfaccia del metodo;
- MLVC, complessità delle variabili locali;
- MCC, complessità ciclomatica del codice;
- w_i , peso determinato statisticamente.

$$CC = \omega_4 CCL + \omega_5 CCI$$

CCI complessità dell'interfaccia della classe

$CCL = \omega_6 ECCL + \omega_7 ICCL$ complessità locale

$ECCL = \sum_j MIC_j$ complessità locale esterna

$ICCL = \omega_8 CACL + \omega_9 CMCL$ complessità locale interna

$CACL = \sum_k CA_k$ complessità degli attributi locali

$CMCL = \sum_s MCC_s$ complessità dei metodi locali

CA costante o CC di classe attributo

ω_i peso determinato statisticamente

- Classe

$$SC = \sum_n CC_n$$

- Sistemi,

Bisogna in oltre prestare attenzione alla qualità delle metriche, qualità che si divide in precisione, aderenza alla realtà e accuratezza.