

## METRICHE

L'approccio quantitativo è fondamentale in tutte i settori dell'Ingegneria.

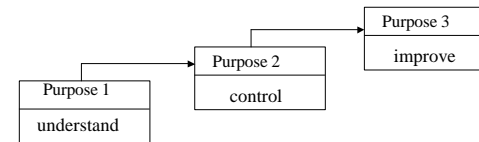
*"What is not measurable make measurable"*  
Galileo Galilei (1564-1642)

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

1

## SOFTWARE MEASUREMENT

- E' il processo di definizione, rilevazione, raccolta ed analisi di dati sullo sviluppo di processi software e sui loro prodotti con l'obiettivo di comprenderli, controllarli, avere informazioni utili al loro miglioramento



prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

2

## MISURA

- Le misure sono l'attività principale di un processo di misura

*Una misura è una empirica oggettiva assegnazione di un numero (o simbolo) ad una entità del mondo reale al fine di caratterizzarne uno specifico attributo*

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

3

## ..misura software

- Una misura software è una funzione che mappa un insieme di oggetti della IS in un insieme di oggetti matematici (numeri, vettori numerici,..)

$$M: (C,R) \rightarrow (N,P)$$

M funzione di mapping  
C entità osservate  
R insieme di relazioni fra le entità  
N oggetti matematici  
P relazioni

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

4

## ..scale..

- Una funzione di mapping può essere riferita a differenti scale
  - Una scala è (C,N,M)
- Una scala è determinata dalle relazioni (R,P) e può essere
  - Nominale, Ordinale, ad Intervalli, Ratio, Assoluta

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

5

## ..tipi di scale..

- **Nominale: Relazione di equivalenza**  
consente una semplice classificazione degli oggetti di partenza. In termini più analitici questo tipo di scala coincide con l'introduzione nell'insieme di partenza di *classi di equivalenza*.  
Ad esempio la classificazione degli errori di programmazione in lessicali, sintattici e semantici ci consente di suddividerli ma non di sapere se un errore sia più o meno grave di un altro;
- **Ordinale: Relazione d'ordine**  
introduce una *relazione d'ordine* fra gli oggetti, per cui è possibile determinare le posizioni relative degli oggetti; non siamo però in grado di quantificare la distanza fra gli oggetti stessi.  
Ad esempio in una scala di valutazione del tipo *sufficiente, buono, ottimo* possiamo dire che *sufficiente* è peggio di *buono* ma non di quanto.

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

6

- **Intervalli: distanza fra gli intervalli**

è possibile misurare la *distanza* fra gli oggetti del dominio e non solo posizionarli tra di loro come con le scale ordinali. La differenza si conserva ma non la moltiplicazione.

Con riferimento all'esempio precedente una scala 6, 8, 10. In Francia il sistema di voti è differente ... (trasformazioni affini)

- **Ratio: distanza fra entità**

rispetto al caso precedente si ha l'introduzione dello *zero*, ciò consente la classificazione di entità che hanno attributi nulli. Inoltre questa scala preserva anche la moltiplicazione.

Gli oggetti possono essere misurati in metri, pollici, .... Le conversioni sono possibili attraverso un fattore moltiplicativo.

- **Assoluto: conteggio**

in questo caso esiste una strategia di conteggio attraverso cui è possibile assegnare ad ogni oggetto un *numero* univocamente determinato.

Numero di test falliti durante il processo di testing.

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

7

## ..attributi ed entità..

- Un processo di misura richiede almeno due passi fondamentali:
  - Identificazione delle entità e degli attributi delle entità
  - Misura degli attributi delle entità

Esempio:

Altezza (**attributo**) degli animali (**entità**).

Dimensione (**attributo**) di sistemi C++ (**entità**).

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

8

## METRICHE

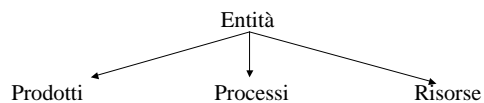
- Dai classici delle misure
  - Una misura diretta di un attributo è una misura che non dipende dalla misura di un altro attributo
  - Una misura indiretta di un attributo è una misura che coinvolge almeno la misura diretta di un altro attributo

Esempio: Lunghezza di un programma → *numero di istruzioni*  
 Affidabilità di un sistema → *tempo medio fra due "failures"*

- Misure dirette = Metriche
- Misure indirette = f(metriche)
- Attributi semplici → Metriche
- Attributi complessi → f(metriche)

Entità	attributo	metrica
Programma	Lunghezza	n.ro di pagine
//	//	n.ro di pollici
//	//	n.ro di linee di codice
//	//	n.ro di istruzioni
//	//	n.ro di bytes
//	//	n.ro di caratteri
//	//	.....

## ..classificazione entità...



Prodotti: programmi, test, documentazione,.....  
 Processi: processi, fasi, attività,.....  
 Risorse: uomini, tools, metodi, hardware, .....

## Attributi

**esterni:** attributi di un'entità che sono visibili e di interesse per l'utente (descrivono lo stato esterno di un'entità e il loro rapporto con l'ambiente in cui vengono usate, ad esempio *facilità d'uso, portabilità, efficienza...*);

**interni:** attributi di un'entità che sono visibili e di interesse del produttore (ad esempio *modularità, strutturazione, tracciabilità...*).

## ..classificazione di Metriche

- **METRICHE DI PRODOTTO**
- **METRICHE DI PROCESSO**
- **METRICHE DI RISORSE**
  
- **METRICHE FUNZIONALI**
- **METRICHE STRUTTURALI**
  - Metriche dimensionali

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

13

## Le Metriche di “base” più note

- **LOC (Lines of Code)**
- **Le metriche di Halstead (Software Science)**
- **Complessità Ciclomatica (McCabe)**
- **Function Points**

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

14

## ...le linee di codice :LOC

- Nasce con ..i programmi e con i linguaggi assembler ed evolve nel tempo
- È una metrica dimensionale, di lunghezza di unità di programma
- Viene molto usata come variabile indipendente in molte misure indirette

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

15

## ...le metriche di Halstead

- **i**

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

16

### ..rilevanti problemi di conteggio..

- Ben si adatta fino ai linguaggi di terza generazione
- Perde quasi senso nei linguaggi di 4<sup>a</sup> e 5<sup>a</sup> generazione (es. query languages)
- Conteggio a livello di linea, a livello di programma, a livello di produzione

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

17

### ..livello linea

- Contare le linee fisiche sullo schermo o su un tabulato
- Contare le linee come linee logiche (ad esempio: istruzioni, o linee delimitate da caratteri speciali come ;)

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

18

### ..livello programma..

- Contare solo le linee eseguibili (EXLOC)
- Contare solo le linee eseguibili e quelle di definizione dei dati  
( $ELOC = EXLOC + DDLOC$ )
- ....anche le linee di commento  
( $EXLOC + DDLOC + CLOC$ )
- ....anche le linee di JCL

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

19

### ..livello produzione..

- Contare solo le linee di nuova produzione
- Contare le linee nuove e quelle modificate
- .....anche quelle riusate (SI)
- Contare le linee consegnate (DSI)
- ...anche quelle del codice temporaneo eventualmente prodotto
- ....anche quelle del codice di supporto

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

20

Loc contate	exloc	Exloc+ ddloc	dsi	tempor anee
nuove	500	1000	2000	5000
riusate	3000	6000	8000	8000
base	5000	9000	10000	10000
Appare nte	8500	16000	20000	23000
durata	3 mesi	3 mesi	3 mesi	3 mesi

prof. A.Cimitile - Ingegneria del Software I - A.A. 2001/2002 21

### La definizione più usata

*‘Una linea di codice è ogni linea di testo di un programma che non sia bianca o un commento, indipendentemente dal numero di istruzioni o frammenti di istruzioni in essa inclusi. In particolare, tali linee di codice possono contenere intestazioni di unità di programma, dichiarazioni ed istruzioni esecutive.’*

Comunque, i commenti sono una parte importante del processo di sviluppo di un programma. Vanno prodotti in modo consistente e non inseriti all'ultimo solo per motivi di tariffazione.

prof. A.Cimitile - Ingegneria del Software I - A.A. 2001/2002 22

### ..nelle misure indirette...

Le LOC sono utilizzate per misurare indirettamente altri attributi, ad esempio:

- la probabilità di presenza di errori nel programma (cresce con la lunghezza);
- il tempo per produrlo;
- la produttività (o il compenso) di un programmatore;
- stima dei costi di un progetto SW

Esempi di misure derivate dalle LOC sono:

**densità di commento** = CLOC/LOC;

**effort** = 5,2\*KLOC<sup>0.91</sup>, misurato in mesi/uomo (KLOC = 1000LOC).

prof. A.Cimitile - Ingegneria del Software I - A.A. 2001/2002 23

### ...le metriche di Halstead

Le metriche della software science nascono dall'ipotesi che il numero di operandi e di operatori è direttamente correlato..”al numero di errori “ presenti in un programma

- n1 numero di operandi distinti
- n2 numero di operatori distinti
- N1 numero di operandi totali
- N2 numero di operatori totali

prof. A.Cimitile - Ingegneria del Software I - A.A. 2001/2002 24

...anche in questo caso..  
rilevanti problemi di conteggio

- Operandi: costanti e variabili usati nel programma
- Operatori: operatori aritmetici , di assegnazione, di lettura, di scrittura, .... , delimitatori, .....

prof. A.Cimitile - Ingegneria del Software I - A.A. 2001/2002

25

In effetti bisogna definire con riferimento alla sintassi di ogni linguaggio cos'è un operando e cos'è un operatore

- Tutte le classi sintattiche di un linguaggio debbono essere misurabili
- Tutte le ambiguità intrinseche del linguaggio debbono essere eliminate
- Tutte le ambiguità di programmazione debbono essere eliminate

Esempio: if R then A else B endif

1 operatore 2 operatori 3 operatori 4 operatori ?

Dipende anche dalla "grana" di misura scelta o necessaria.  
Strumenti di misura tarabili .

prof. A.Cimitile - Ingegneria del Software I - A.A. 2001/2002

26

Partendo da  
 $n_1$   $n_2$   $N_1$   $N_2$

- Vocabolario:  $n = n_1 + n_2$
- Lunghezza:  $N = N_1 + N_2$
- Volume:  $V = N * \log_2 n$

prof. A.Cimitile - Ingegneria del Software I - A.A. 2001/2002

27

Le ambizioni di Halstead: "la metafisica intuitiva"

- La lunghezza stimata una volta noti  $n_1$  ed  $n_2$ :  
 $N^* = n_1 \log_2 n_1 + n_2 \log_2 n_2$
- In un linguaggio ideale  $N_1 = n_1$   $N_2 = n_2$   $n_1 = 2$  e da qui
- Volume potenziale:  $V_p = (2 + n_2) \log_2 (2 + n_2)$

prof. A.Cimitile - Ingegneria del Software I - A.A. 2001/2002

28

...continuando

- Livello di implementazione:  $L=Vp/V$
- Difficoltà:  $D=1/L$
- Sforzo:  $E=V*D=V/L$
- Tempo di sviluppo:  $T=S/(Stround\ number)$
- Errori consegnati:  $V/3000$

Complessità Ciclomatica  
(McCabe)

Si applica al control flow di un programma

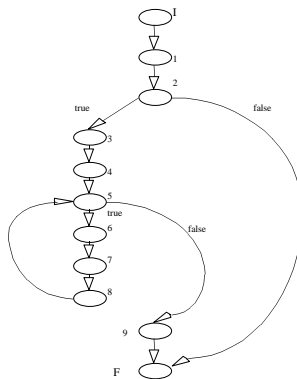
$$v=e-n+2p$$

n numero di nodi

e numero di archi

p numero di componenti connesse

e=12  
n=11  
p=1  
v=3



La metrica è derivata dalla teoria dei grafi dove , per un grafo orientato e fortemente connesso g , viene definito il numero ciclomatico

$$v(g)=e-n+1$$

e per un grafo composto da p grafi fortemente connessi

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

È opportuno ricordare che un grafo si dice fortemente connesso se comunque si prendono due suoi nodi a e b esiste almeno un cammino ab ed almeno un cammino ba

Dato un grafo fortemente connesso

Qualsiasi cammino può essere rappresentato da un vettore di e numeri interi, essendo e il numero di archi del grafo

(1,2,3,4) <111100>  
 (1,5,6,2,3,4,1) <211111>  
 (1,5,4) <100110>

Il componente i.mo del vettore rappresenta il n.ro di attraversamenti dell'arco i.mo

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002 33

## I FUNCTION POINTS

- I punti funzione (FP) sono una misura delle funzioni espletate da un sistema software e visibili all'utente ( entità logico funzionali che l'utente comprende facilmente)
- Nascono dall'esigenza di avere un riferimento a fatti oggettivi per discutere e valutare con l'utente piani, stime, e modifiche relative ad un sistema software
- Entità analizzate: "Requisiti di utente" o "Business Models" visti secondo l'attributo "funzionalità richieste"

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002 34

### ...i pregi

- È una misura che può essere fatta molto presto nel CVS
- Indipendenza dal linguaggio
- Indipendenza dalla tecnologia

Esempio:

	Assembler	Cobol
	30 FP	30 FP
	10 KLOC	3 KLOC
Non coding activities	5 p/m	5 p/m
Coding activities	5 p/m	3 p/m
Total project	10 p/m	8 p/m
FP per person month	3.0	3.75
Cost per function	\$ 1677	\$ 1333

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002 35

Esiste un organismo internazionale, International Function Point Users Group (IFPUG), fondato nel 1984 che si occupa delle problematiche connesse ai FP.

Esistono molte versioni dei FP

Albrecht and Gaffney 83  
 GUIDE 85  
 IFPUG 1986, 88, 90, 91, 92, **94**

**IFPUG produce ed aggiorna:**  
**Function Points Counting Practice Manual**

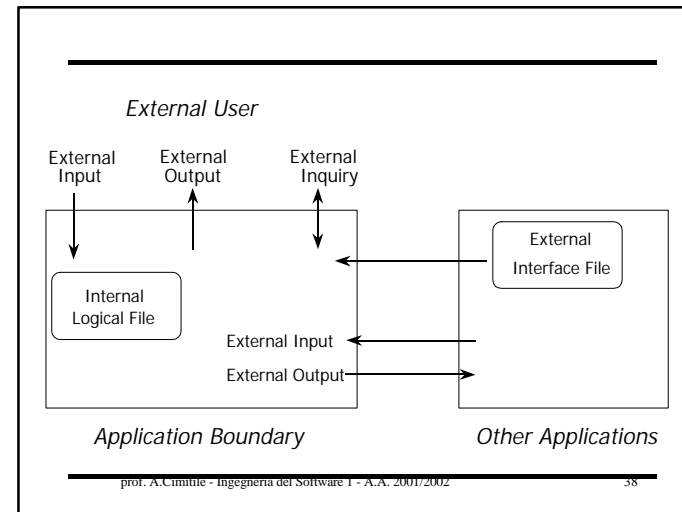
prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002 36

## ..entità e concetto di confine

- **Boundary**: è una linea chiusa che definisce il confine del sistema software rispetto agli altri sistemi con i quali interagisce
- Con riferimento al confine vengono definite 5 classi di entità:
  - **Internal logical files**
  - **External interfaces files**
  - **Inputs**
  - **Outputs**
  - **Inquires**

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

37



prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

38

## data function types

- Rappresentano entità relative a dati interni ed esterni all'applicazione

### INTERNAL LOGICAL FILE (ILF)

sono residenti dentro il confine del sistema software e gestiti dallo stesso

### EXTERNAL INTERFACE FILES (EIF)

sono residenti all'esterno del confine dell'applicazione e vengono gestiti da altre applicazioni

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

39

## transactional function types

- Rappresentano funzionalità di utente, tutte caratterizzate attraverso gli accessi ai files, a copertura dei requisiti

### External INPUTS (EI)

funzionalità di utente caratterizzate da updating di dati (add, change, delete) di un ILF

### External OUTPUTS (EO)

funzionalità di utente caratterizzate da outputs generati da ILF o EIF

### External INQUIRES (EQ)

funzionalità di utente caratterizzate da interrogazioni di ILF o EIF

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002

40

- INPUT
    - Dati o informazioni di controllo che attraversano il confine (..verso il sistema)
    - I dati modificano lo stato di un ILF
    - Le informazioni di controllo attivano e/o controllano una specifica funzione di utente
    - Un input è una elaborazione (transazione) unitaria per logica e formato
  - OUTPUT
    - Dati o informazioni di controllo che attraversano il confine (..verso l'esterno)
    - Un output è una elaborazione (transazione) unitaria per logica e formato
  - INQUIRY
    - È una combinazione di input/output (.. A domanda ..risposta immediata..)
    - Non confondere con EI (updating di ILF) seguito da EO (.. dati derivati o calcolati..)
    - Un inquiry è una elaborazione (transazione) unitaria per logica e formato
- prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002 41

## Misura dei FP

Scala ordinale (basso, medio, alto)

Matrice dei punti funzione  
 MFP (i: classe fp ; j: valore in scala)= n.ro di fp

Matrice dei pesi MPFP

Type	Low	Average	High
Ext Input	3	4	6
Ext Output	4	5	7
ELF	7	10	15
ILF	5	7	10
Ext Inquiry	3	4	6

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002 42

## Unadjusted function count ufc

Ufc=0  
 Per ogni i  
     per ogni j  
         ufc= ufc+ MFP(i,j)\*MPFP(i,j)

Infine si procede ad un aggiustamento finale attraverso la valutazione di un fattore di complessità tecnica TFC il cui valore Ricade nell'intervallo (0.65, 1.35)  
 .....è un aggiustamento contenuto in -35%, +35%

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002 43

## Calcolo TFC ed FP finale

Si valutano 14 fattori di complessità nella scala (0,1,2,3,4,5)  
*communications, distribute processing, performance objectives, operation configuration load, transaction rate, on-line data entry, end user efficiency, on-line update, complex processing logic, reusability, installation ease, operational ease, multiple sites desire to facilitate change.*

Detta f(i) la valutazione del fattore i-mo  
 TFC= 0.65+0.01(sommatoria f(i))

**MISURA FINALE**      FP=UFC\*TFC

prof. A.Cimitile - Ingegneria del Software 1 - A.A. 2001/2002 44

## ..il conteggio..

Anche in questo caso sono notevoli i problemi di definizione di una strategia di conteggio condivisa dalle parti interessate

Soggettività presente sia nella individuazione dei fp che nella assegnazione dei valori

L'importanza dello International Function Points User Group, che ha, fra l'altro, definito un manuale di riferimento (IFPUG 4.0) proprio per la definizione ed il conteggio dei FP

## ..per finire..

I Function Point sono stati introdotti per "misurare" i grandi sistemi transazionali. Interfacce utente, software real time e applicazioni scientifiche non trovano adeguate Strategie di conteggio.

Inoltre I sistemi Object Oriented pongono nuovi problemi. Ad esempio, in un sistema OO cosa significa ILI (Internal Logical File).