

# **Quaderno n° 16**

## **"Guida per l'utilizzo delle metriche nello sviluppo del software"**

Metriche Dimensionali

*La quantità del software*

# **Le metriche dimensionali**

# Argomenti

- Metriche Tecniche vs. Metriche Funzionali
  - LOC
  - Function Point
  - Cosmic Full Function Point
- Metodi di stima
  - Early & Quick Function Point
- Confronti e Conclusioni

## Perché misurare le dimensioni del SW

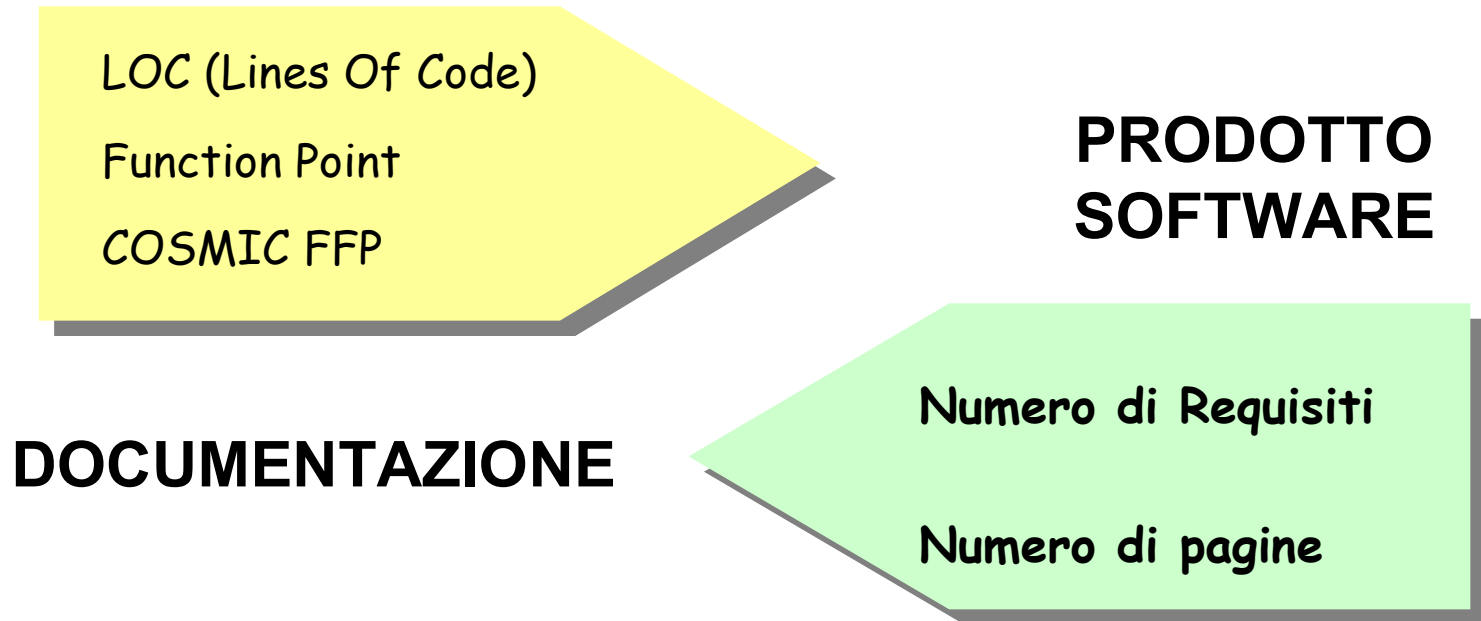
- Confronto di "prestazioni" e "caratteristiche" in forma normalizzata e oggettiva, ad esempio:
  - $Produttività = Dimensione / Impegno\_lavorativo$  (o inv.)
  - $Densità\_difetti = N\_difetti / Dimensione$
- Stima economica di nuovi progetti
  - Attività: sviluppo e manutenzione del SW
  - Dimensione: quantità di software impattato come *driver primario* dei costi

# ***Misurare le dimensioni del SW per..***

- monitorare lo stato di avanzamento del progetto rispetto alle attese
- valutare l'entità del patrimonio in esercizio
- misurare la produttività della manutenzione 'di presidio' (correzione di errori su applicazioni già in esercizio)
- misurare la Change Request in corso d'opera

# ***Dimensioni del SW: quantità di funzioni o quantità di codice?***

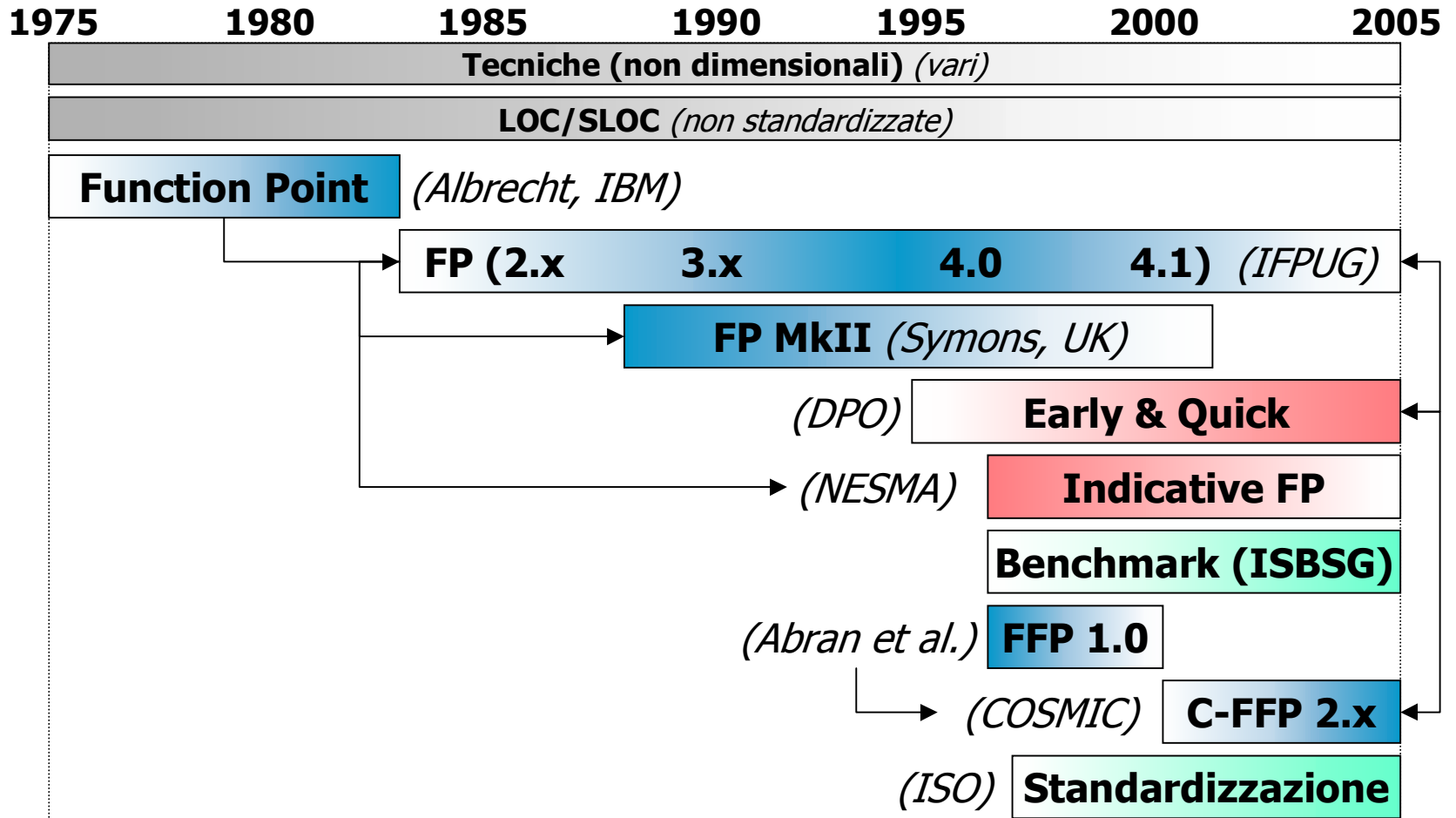
Le dimensioni del software possono essere misurate secondo una prospettiva funzionale o da un punto di vista puramente tecnico



# Misura **FUNZIONALE** vs. **TECNICA**

<b>Categoria</b>	<b>Metriche fisico-tecniche (oggetti)</b>	<b>Metriche logico-funzionali (concetti)</b>
<b>Oggetto</b>	Dimensione fisica ( <i>Lines Of Code</i> )	Dimensione logica ( <i>Funzioni</i> )
<b>Fonti</b>	Codice, Moduli fisici, Interfacce, Modelli fisici	Requisiti funzionali utente, Modelli concettuali
<b>Supporto</b>	Elevato	Limitato
<b>Utilità</b>	A posteriori (consuntivazione)	A priori e a posteriori (prev. e cons.)
<b>Vantaggi</b>	<ul style="list-style-type: none"> <li>- Oggettive</li> <li>- Automatizzabili</li> </ul>	<ul style="list-style-type: none"> <li>- Ottenibili in fasi alte</li> <li>- Condivisibili con l'utente</li> <li>- Indipendenti dal linguaggio e piattaforma</li> </ul>
<b>Svantaggi</b>	<ul style="list-style-type: none"> <li>- Richiedono impostazione tecnica adeguata</li> <li>- Mescolano qualità e quantità</li> <li>- Previsione difficile</li> <li>- Scarsa valenza per l'utente</li> <li>- Non significative o non applicabili per tutti i linguaggi</li> </ul>	<ul style="list-style-type: none"> <li>- Richiedono visione non tecnica (impatto culturale)</li> <li>- Non automatizzabili</li> </ul>

# Evoluzione storica



# Linee di Codice

- Altre denominazioni utilizzate in letteratura:
- DSI - Delivered Source Instructions
- LOC - Lines of Code
- SLOC - Source Line of Code

***non esiste uno standard internazionale  
per LOC che comprenda tutti i linguaggi  
procedurali***

## LOC: esempi di definizione...

- **"una linea di codice è una linea di un programma che non è un commento o una linea bianca. Di conseguenza la linea di codice include tutte le linee che contengono le intestazioni del programma, le dichiarazioni e le istruzioni eseguibili e non eseguibili". [E. Fenton ]**
- **LOC è ogni linea del testo di un programma che non è di commento o una linea bianca, riguardante il numero di frasi o frammenti di frase sulla linea. Esse specificatamente includono tutte le linee di intestazione, dichiarazioni, e frasi eseguibili e non eseguibili. [Conte, Dunsmore e Shen ]**

*Vengono incluse anche "altre cose" oltre le istruzioni*

## **SLOC - Source Line of Code...**

- Gli SLOC sono state definite con lo scopo di **ESCLUDERE** tutto ciò che non riguardi l'esecuzione del programma. Esempio di definizione tramite la checklist del SEI (Software Engineering Institute)
- Approccio utilizzato nel modello COCOMO [COCOMO II Boem pp. 15]

*Vengono incluse solo  
istruzioni ma...*

## **...SLOC - Source Line of Code**

... ma nondimeno vi può essere la necessità di tenere in considerazione anche lo "sforzo" profuso nel produrre altre "linee" come quelle di commento, che in questo caso andrebbero contate [vedi Boehm 1981, pp. 58-59]

# La checklist del SEI

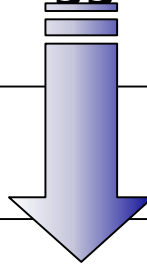
Definition Checklist for Source Statement Counts						
<u>Physical Source Lines of Code</u> <i>(basic definition)</i>					Date:	<u>7/9/92</u>
						<u>SEI</u>
Measurement unit:		Physical source lines	<input checked="" type="checkbox"/>	Logical source statements	<input type="checkbox"/>	
Statement type	Definition	<input checked="" type="checkbox"/>	Data array	<input type="checkbox"/>	Includes	Excludes
<i>When a line or statement contains more than one type, classify it as the type with the highest precedence.</i>						
1 Executable	Order of precedence ->	1	<input checked="" type="checkbox"/>			
2 Nonexecutable						
3 Declarations		2	<input checked="" type="checkbox"/>			
4 Compiler directives		3	<input checked="" type="checkbox"/>			
5 Comments						
6 On their own lines		4				<input checked="" type="checkbox"/>
7		5				<input checked="" type="checkbox"/>
8 Banners and nonblank spacers		6				<input checked="" type="checkbox"/>
9 Blank (empty) comments		7				<input checked="" type="checkbox"/>
10 Blank lines		8				<input checked="" type="checkbox"/>
12						
How produced	Definition	<input checked="" type="checkbox"/>	Data array	<input type="checkbox"/>	Includes	Excludes
1 Programmed					<input checked="" type="checkbox"/>	
2 Generated with source code generators					<input checked="" type="checkbox"/>	
3 Converted with automated translators					<input checked="" type="checkbox"/>	
4 Copied or reused without change					<input checked="" type="checkbox"/>	
5 Modified					<input checked="" type="checkbox"/>	
6 Removed						<input checked="" type="checkbox"/>
7						
8						

Figure 2: Portion of SEI Checklist for Defining Lines of Code

[www.sei.cmu.edu/pub/documents/articles/pdf/using.data.asm.pdf](http://www.sei.cmu.edu/pub/documents/articles/pdf/using.data.asm.pdf)



**Non esiste uno standard internazionale per LOC che comprenda tutti i linguaggi procedurali**



- **misure anche molto diverse di LOC sullo stesso programma, in funzione della definizione adottata**
- **misure diverse di LOC sullo stesso programma scritto con linguaggi diversi**



- **LOC perdono significato quando sono applicate alla misurazione di programmi scritti con formalismi di IV e V generazione (spreadsheet, query-languages, etc)**
- **La misura di produttività è inversamente proporzionale al livello del linguaggio utilizzato; risulta penalizzante usare linguaggi come ADA, APL, C++ .**

# Functional Size Measurement – FSM

ISO/IEC/JTC1/SC7 Standard #14143

- **“Functional Size:** A size of software derived by quantifying the functional user requirements”
- **“Functional User Requirements:** a sub-set of the user requirements. The Functional User Requirements represent the user practices and procedures that the software must perform to fulfil the users’ needs. They exclude Quality Requirements and any Technical Requirements.”
- **“User:** Any person, physical device, item of software, etc which interacts with the software being measured”

# Metriche funzionali: le più diffuse

- **Function Point IFPUG**
  - Varianti: NESMA, Mk II (UKSMA)
- **Full Function Point COSMIC**
- **Tecniche di stima anticipata**
  - Indicative & Estimated Function Point
  - Early & Quick (Full) Function Point (IFPUG & COSMIC)

# Metriche Funzionali: organismi coinvolti

- **IFPUG (International Function Point Users Group)**
  - Comitati di sviluppo (CPC, CC, ...)
  - Conferenze
  - Certificazione (CFPS, Tool, Materiali)
- **GUFPI-ISMA (Gruppo Utenti FP Italia – Italian SW Metrics Association)**
  - Comitati di ricerca
  - Seminari
- **NESMA, UKSMA, DASMA, JFPUG, ecc.**
- **COSMIC (Common Software Measurement International Consortium)**
  - Comitati di sviluppo
  - Partecipazioni di ricerca
- **ISBSG (International Software Benchmarking Standards Group)**
  - Raccolta dati e analisi statistica
- **ISO/IEC (International Organization for Standardization & International Electrotechnical Commission)**
  - Pubblicazione di Standard

# Standard ISO-IEC pubblicati

- 14143-1:1998      **"Information technology – Software measurement – Functional size measurement – Part 1: Definition of concepts"**
- 14143-2:2002      **"Id. – Part 2: Conformity evaluation of software size measurement methods to ISO/IEC 14143-1:1998"**
- 14143-3:2003      **"Id. – Part 3: Verification of functional size measurement methods"**
- 14143-4:2002      **"Id. – Part 4: Reference model"**
  
- 15939:2002        **"Software engineering – Software measurement process"**
  
- 19761:2003        **"Software engineering – COSMIC-FFP – A functional size measurement method"**
- 20926:2003        **"Software engineering – IFPUG 4.1 Unadjusted functional size measurement method – Counting practices manual"**
- 20968:2002        **"Software engineering – Mk II Function Point Analysis – Counting Practices Manual"**
  
- 14143-5 (*nd*)      **"Determination of Functional Domains for use with Functional Size Measurement"**

# I Function Point: un'analogia efficace

- Dimensione di un appartamento ( $m^2 \Leftrightarrow FP$ )
- Indipendente da:
  - **Chi** costruisce
    - o Quantità, Tipo, Esperienza risorse
  - **Quando** è misurata
    - o Sul Progetto, Schema, Planimetria
    - o Sull'Edificio Iniziato (fondamenta, fase iniziale)
    - o Sull'Edificio Consegnato (definitivo, fase finale)
  - **Come** è costruito
    - o Metodo, Processo, Sequenza/Fasi, Qualità dei Materiali, Utilizzo di Moduli Prefabbricati (riuso)

## Obiettivi della FPA

- Misurare le funzionalità che l'utente richiede e riceve
- Misurare lo sviluppo e la manutenzione del software indipendentemente dalla tecnologia utilizzata



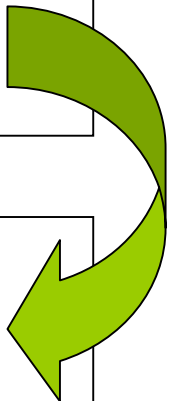
# Benefici della FPA



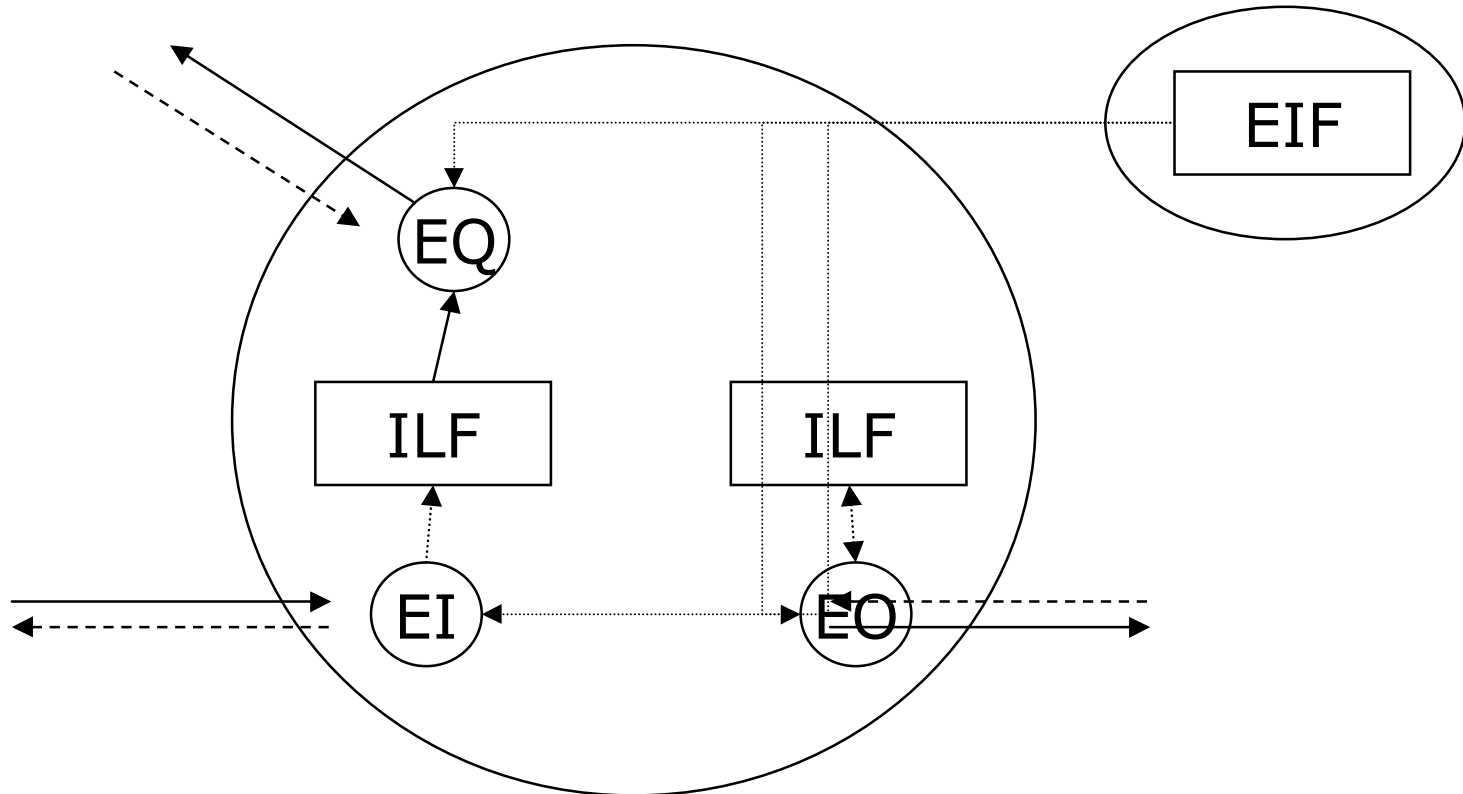
Uno strumento per:

- determinare la grandezza di un pacchetto applicativo acquistato attraverso la quantificazione di tutte le sue funzionalità.
- Misurare l'insieme di un prodotto sw a sostegno di analisi sulla qualità e sulla produttività.
- stimare costi e risorse necessarie per lo sviluppo e la manutenzione del software.
- Un fattore di normalizzazione per effettuare confronti tra software

- Misura le funzionalità dal punto di vista utente
- È indipendente dal linguaggio e dalla piattaforma
- È disponibile fin dalle prime fasi del ciclo di vita del software



# FP: Modello di riferimento



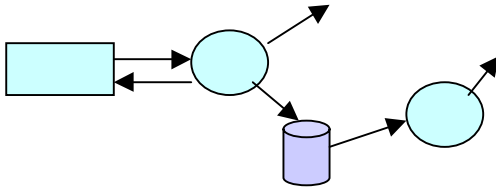
**Dominio utente (umani + sistemi)**

# FP: Basi del Conteggio

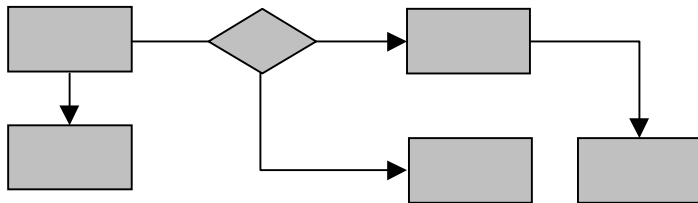
Requisiti utente



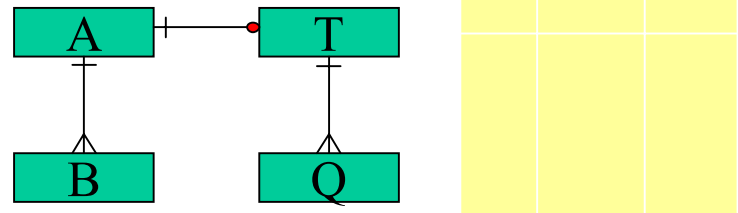
Modello dei processi



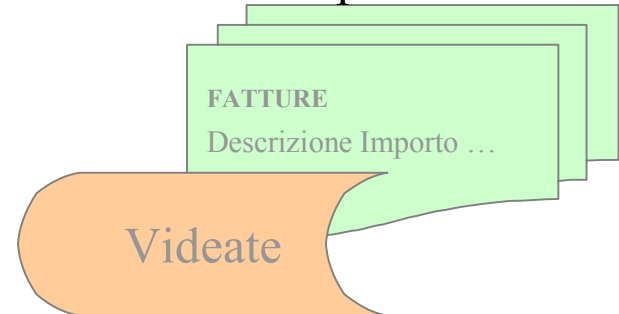
Diagrammi di flusso logico / fisico

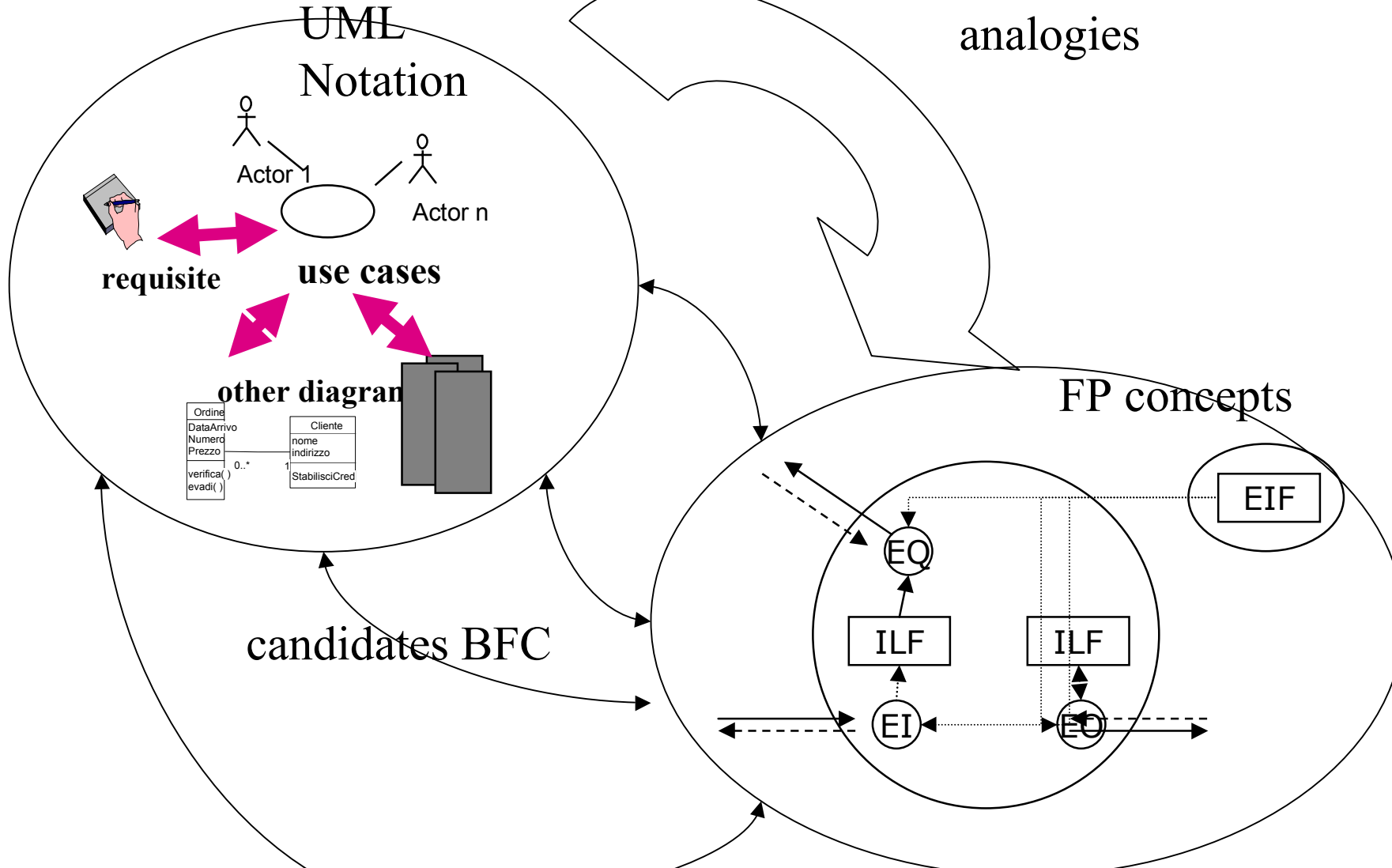


Modelli concettuali, logici, fisici di dati

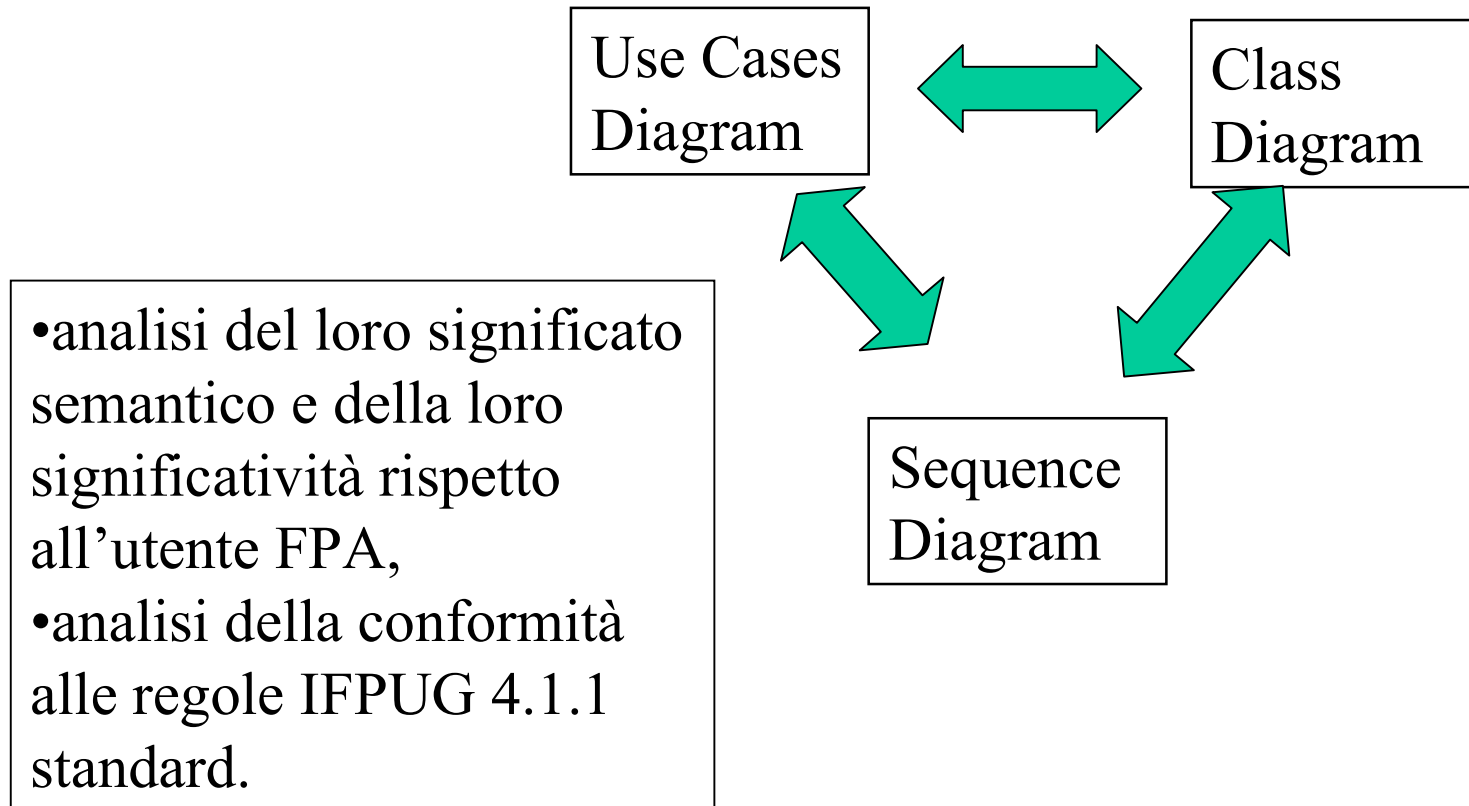


Prospetti e videate

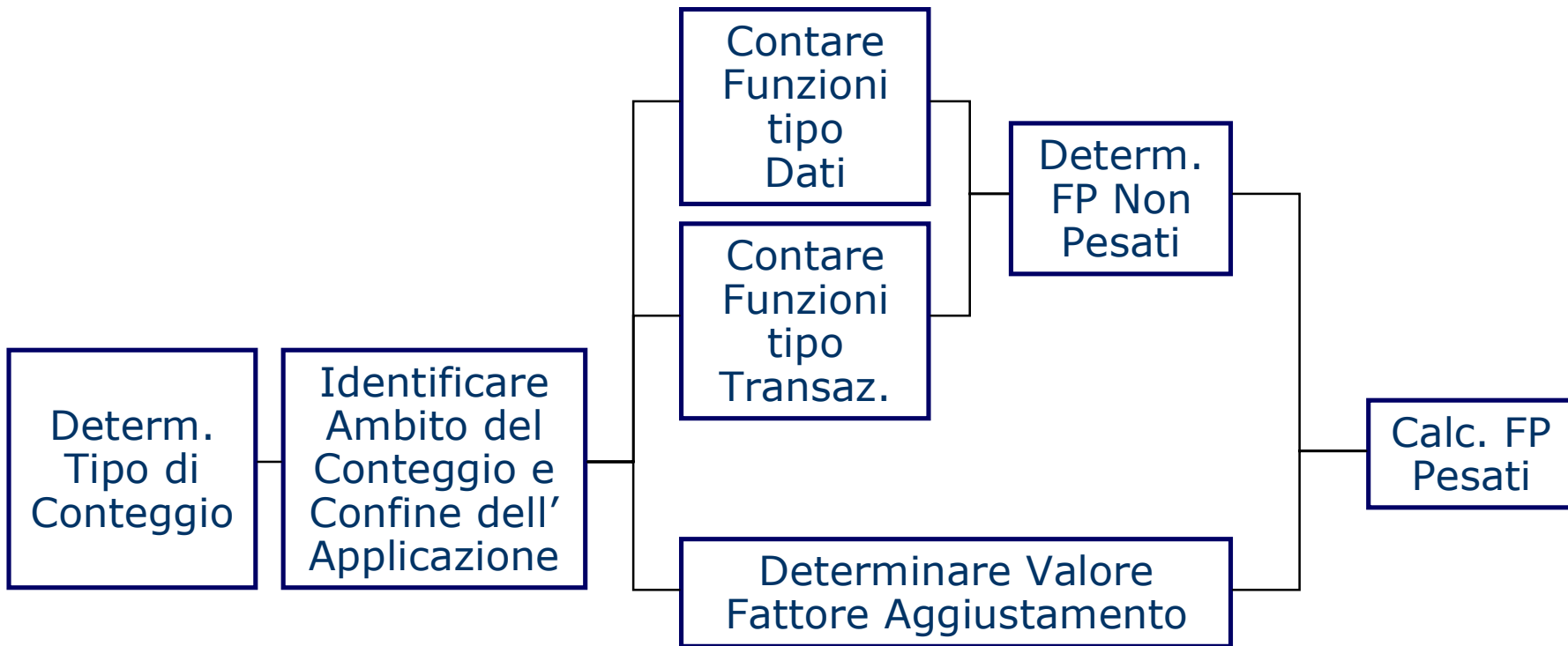




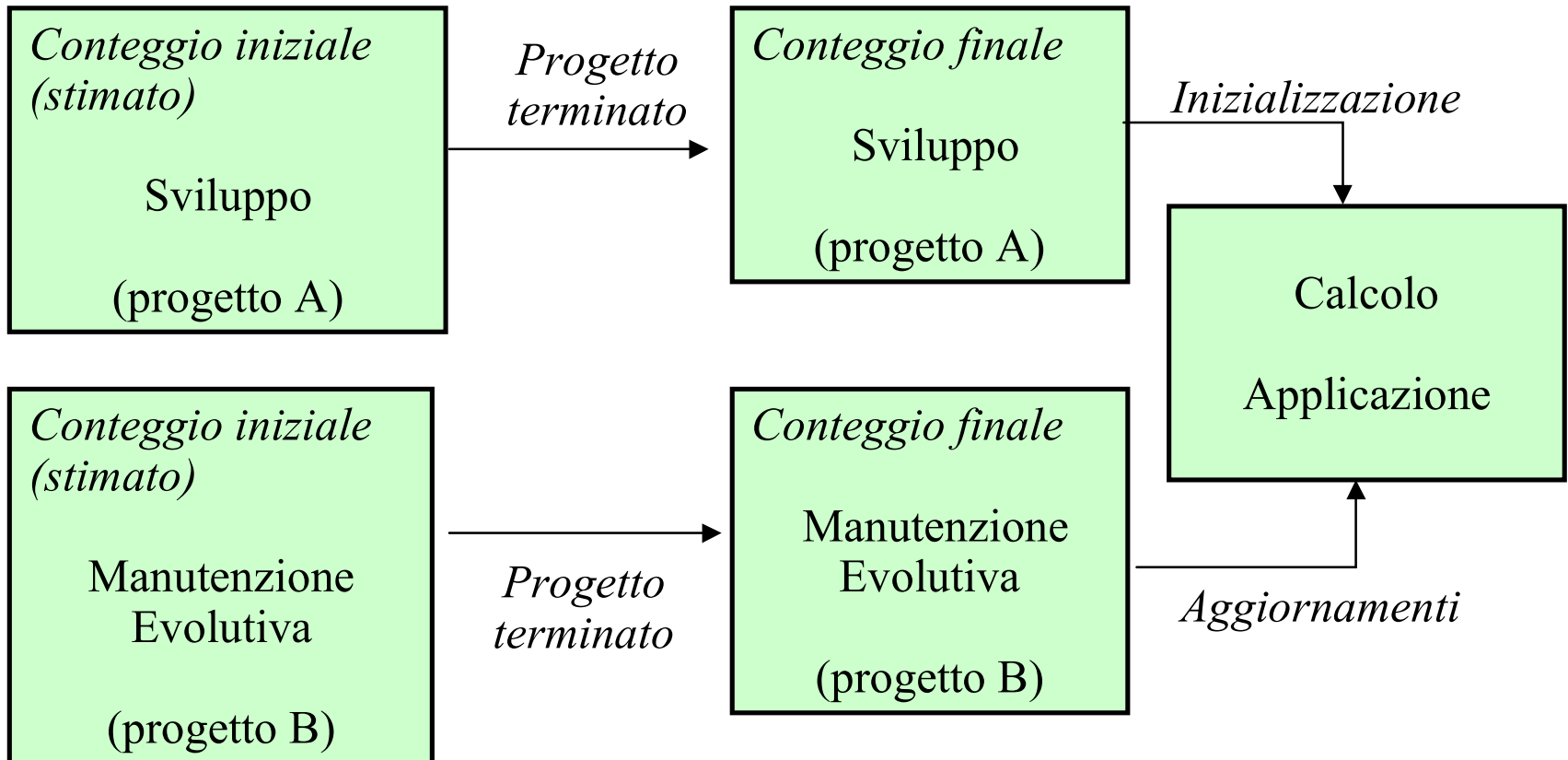
# Validazione dei candidati BFC:



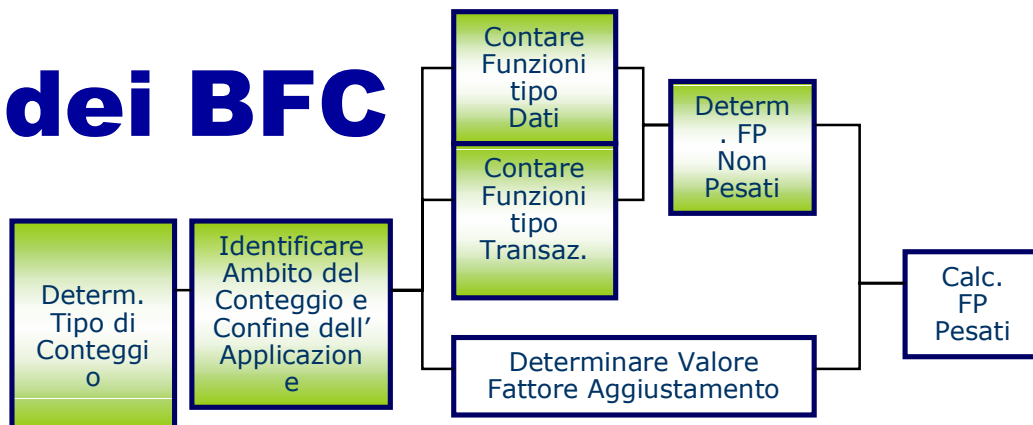
# FP: Procedura per il Conteggio



# Relazioni fra Tipi di Conteggio



# Complessità dei BFC

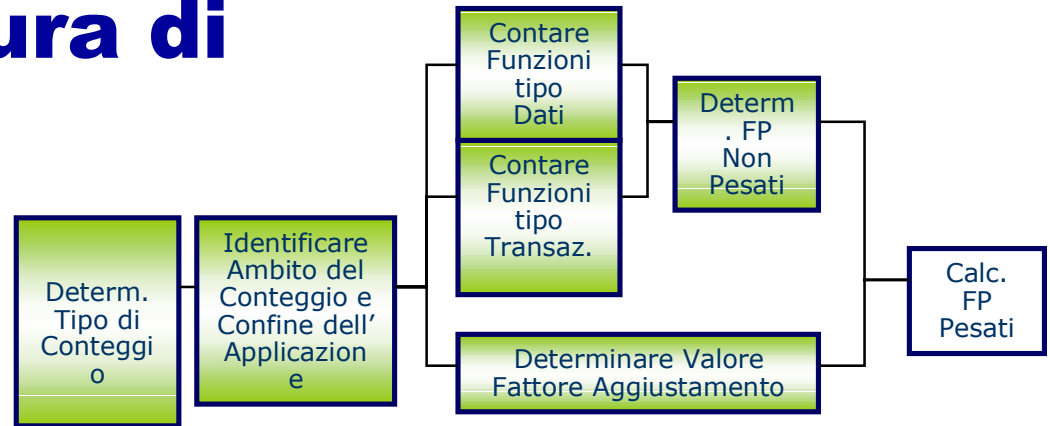


Tipo di funzione	Bassa	Media	Alta
EI	x 3	x 4	x 6
EO	x 4	x 5	x 7
EQ	x 3	x 4	x 6
ILF	x 7	x 10	x 15
EIF	x 5	x 7	x 10

# Le 14 GSC

- ⇒ **comunicazione dati**
- ⇒ **distribuzione dell'elaborazione**
- ⇒ **prestazioni**
- ⇒ **utilizzo intensivo della configurazione**
- ⇒ **frequenza delle transazioni**
- ⇒ **inserimento dati interattivo**
- ⇒ **efficienza per l'utente finale**
- ⇒ **aggiornamento interattivo**
- ⇒ **complessità elaborativa**
- ⇒ **riusabilità**
- ⇒ **facilità di installazione**
- ⇒ **facilità di gestione operativa**
- ⇒ **molteplicità di siti**
- ⇒ **facilità di modifica**

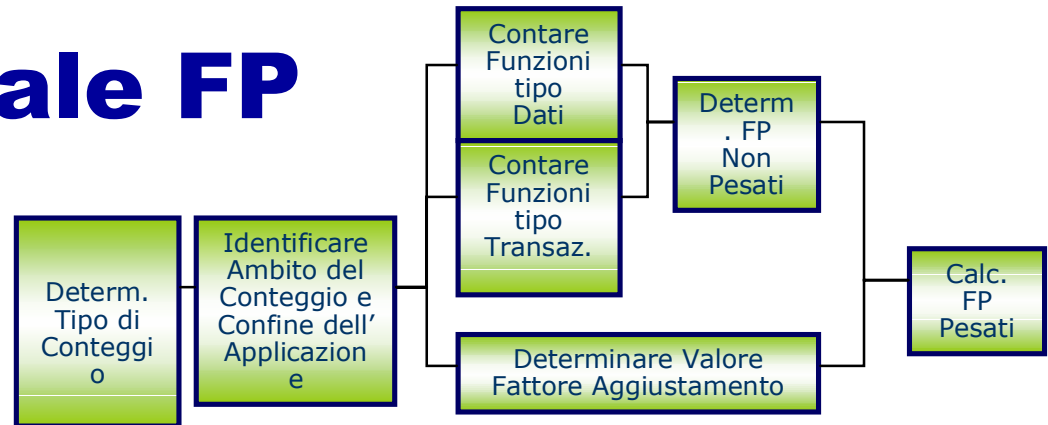
# VAF: Procedura di calcolo



- Valutare il DI (Degree of Influence) delle 14 GSC
- Calcolare il Total DI
- Calcolare il VAF:

$$\begin{aligned} \mathbf{VAF} &= \mathbf{(TDI \times 0,01) + 0,65} \\ &= \mathbf{(\sum GSC / 100) + 0,65} \end{aligned}$$

# Calcolo finale FP

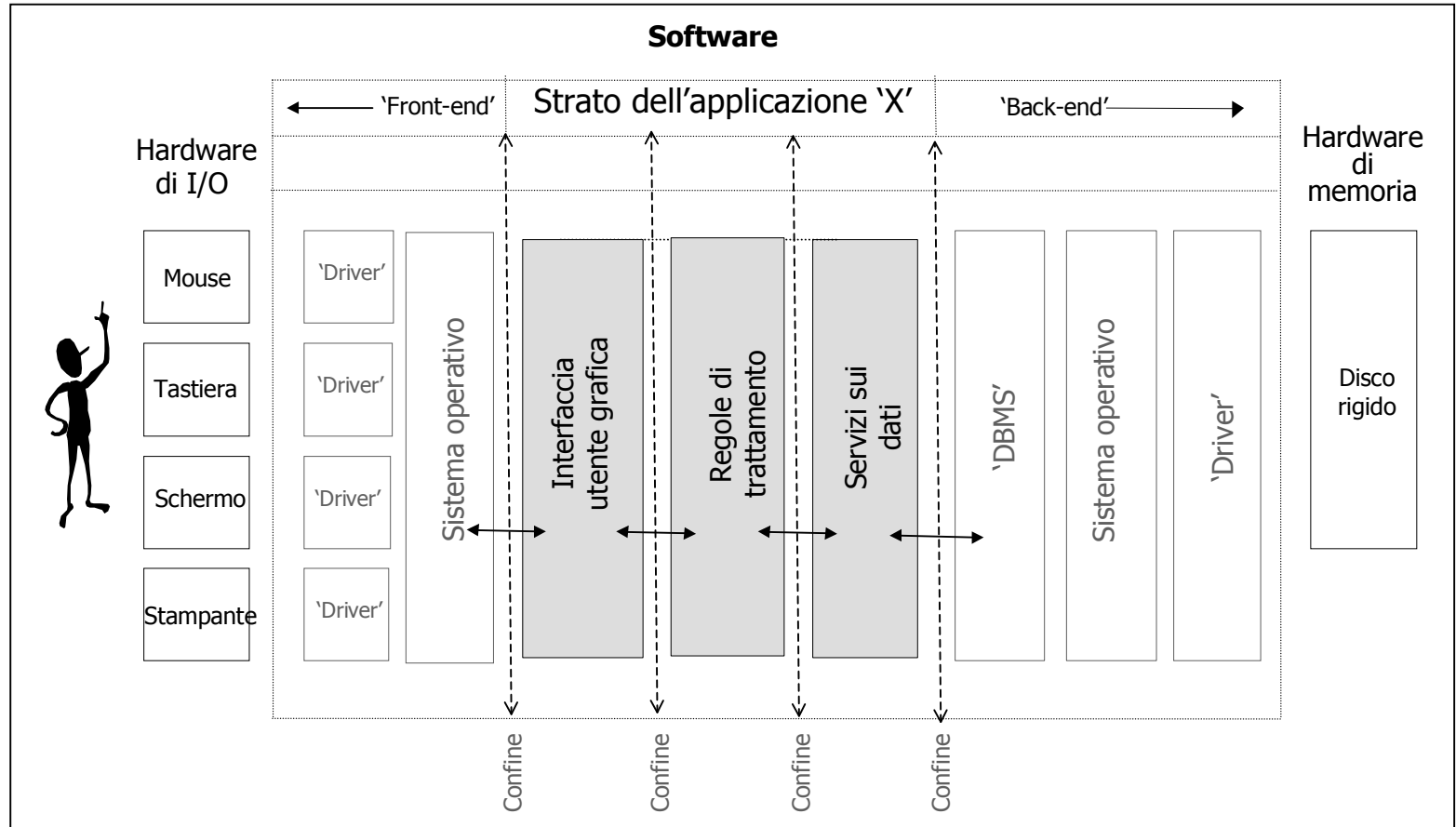


Applicazione: conteggio iniziale	<b>AFP = ADD x VAF</b>
Progetto di Sviluppo	<b>DFP = (UFP + CFP) x VAF</b>
Progetto di Manutenzione Evolutiva	<b>EFP = [(ADD+CHGA+ CFP) x VAFA] + (DEL x VAFB)</b>
Applicazione: dopo la MEV	<b>AFP = [(UFPB+ADD+CHGA) - (CHGB+DEL)] x VAFA</b>

# Una nuova metrica funzionale



# Modello del contesto COSMIC (esempio)



# Procedura COSMIC

## Fase di Mappatura

- Individuazione degli Strati
- Individuazione dei Confini
- Individuazione dei Processi Funzionali
- Individuazione dei Gruppi di Dati
- Individuazione degli Attributi dei Dati

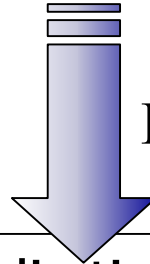
## Fase di Misurazione

- Identificazione dei Movimenti di Dati (E, X, R & W)
- Applicazione della Funzione di Misurazione
  - Standard di Misurazione (1 CFSU  $\forall$  E, W, R & W)
- Aggregazione dei Risultati
  - Totali distinti per strato

# Early & Quick Function Point Analysis



**FPA: dettaglio richiesto troppo alto per le previsioni**



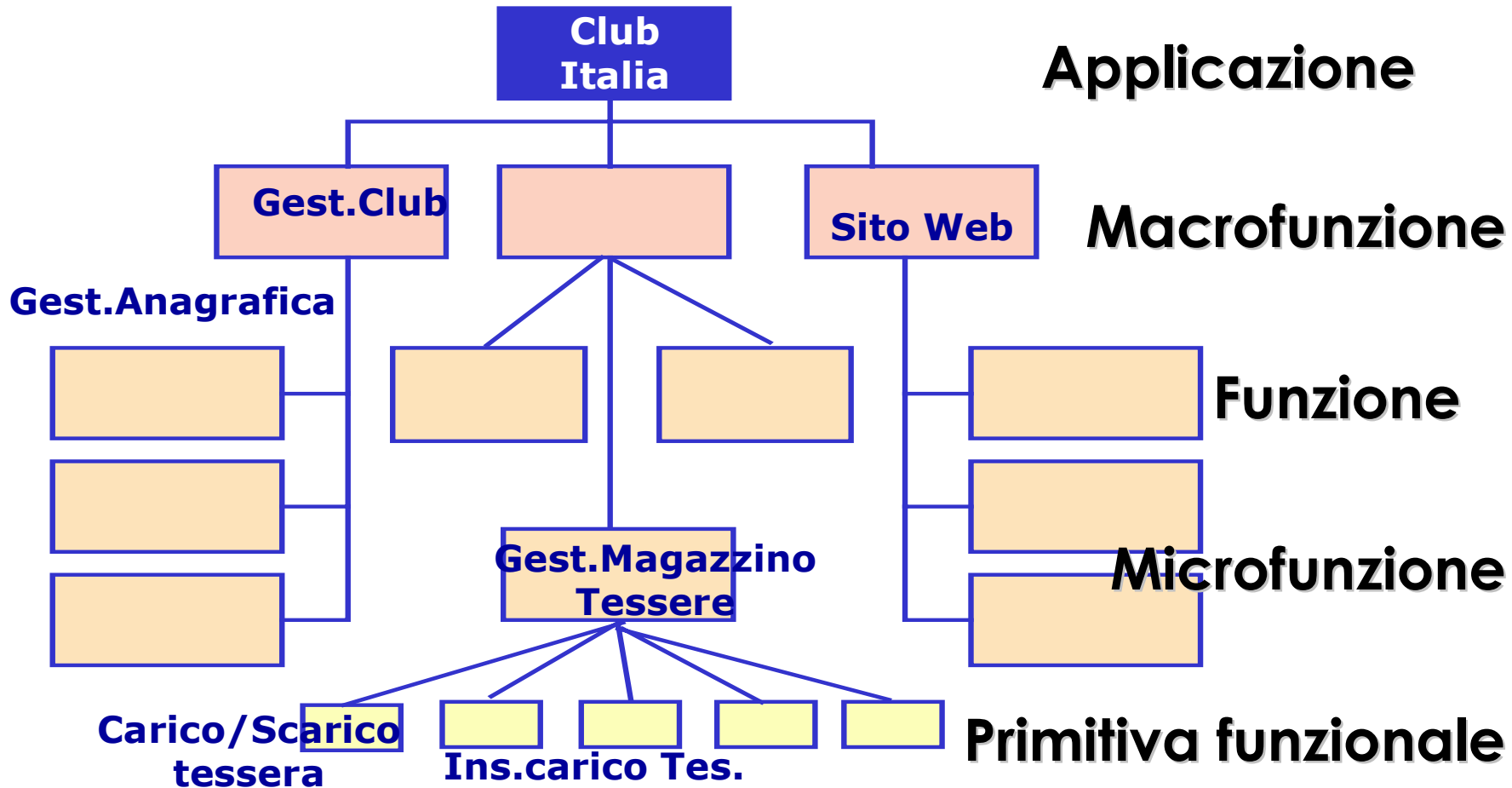
**EFP: una soluzione per:**

- una via più precoce di stima dei FP
- una via più veloce di stima dei FP
- una via più economica di stima dei FP



- compatibilità con le regole standard emanate dall'IFPUG

## EFP: L'aggregazione strutturata

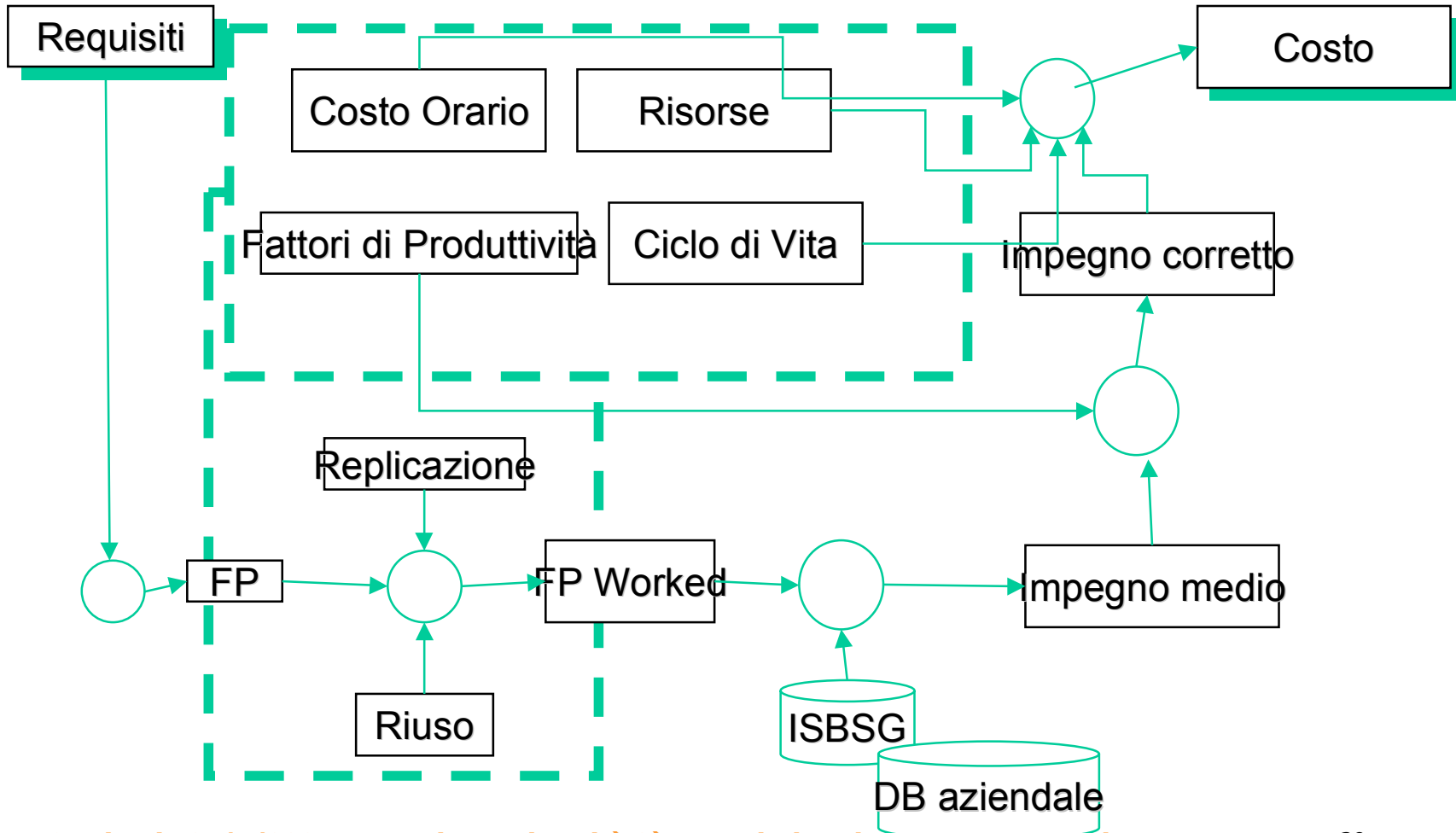


# EFP: Risultati incoraggianti

- $(UEFP - USFP) / USFP < 10\%$

- $Pred(0.25) = 100\%$

# Dai Requisiti al Costo



# Equazioni dell'impegno ISBSG

<b>Equazione media ISBSG</b>	$\text{Impegno} = a + b (\text{FP})^c$
<b>Tutti i progetti</b>	$\text{Impegno} = 11.79 * \text{FPsize}^{0.898}$
<b>Solo 3GL</b>	$\text{Impegno} = 5.76 * \text{FPsize}^{1.062}$
<b>Solo 4GL</b>	$\text{Impegno} = 9.32 * \text{FPsize}^{0.912}$
...	...

\* unità di impegno : **ore persona**

# Fattori correttivi

Fattori		Valorizzazione					
		Molto basso	Basso	Normale	Alto	Molto alto	Altissimo
FC 1	affidabilità richiesta al prodotto	0.75	0.88	1.00	1.15	1.40	
FC 2	dimensione della base di dati		0.94	1.00	1.08	1.16	
FC 3	complessità del prodotto	0.70	0.85	1.00	1.15	1.30	1.65
FC 4	vincoli tecnologici		0.87	1.00	1.20	1.64	2.20
FC 5	volatilità dell'ambiente hw/sw		0.87	1.00	1.15	1.30	
FC 6	uso di tecniche moderne di programmazione	1.24	1.10	1.00	0.91	0.82	
FC 7	uso di strumenti software avanzati	1.24	1.10	1.00	0.91	0.83	
FC 8	capacità del gruppo di analisi	1.46	1.19	1.00	0.86	0.71	
FC 9	esperienza sull'applicazione	1.29	1.13	1.00	0.91	0.82	
FC 10	capacità del gruppo di programmazione	1.42	1.17	1.00	0.86	0.70	
FC 11	esperienza sull'ambiente hw/sw	1.21	1.10	1.00	0.90		
FC 12	esperienza nel linguaggio di programmazione	1.14	1.07	1.00	0.95		

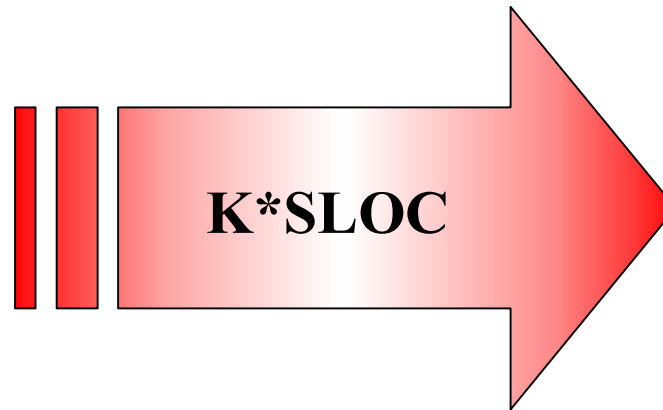
# Analogie e differenze

<b>Metrica</b>	<b>LOC</b>	<b>FP IFPUG</b>	<b>E&amp;Q FP</b>	<b>FFP COSMIC</b>
<b>Tipo</b>	Tecnico	Funzionale	Funzionale	Funzionale
<b>Oggetto</b>	Codice	FUR	FUR	FUR
<b>Applicabilità</b>	Non omogenea (dipende dal linguaggio)	<ul style="list-style-type: none"> <li>- Fasi avanzate del ciclo di vita</li> <li>- Indipendente dalla tecnologia</li> </ul>	<ul style="list-style-type: none"> <li>- Tutte le fasi del ciclo di vita</li> <li>- Indipendente dalla tecnologia</li> </ul>	<ul style="list-style-type: none"> <li>- Fasi avanzate del ciclo di vita</li> <li>- Indipendente dalla tecnologia</li> </ul>
<b>Confrontabilità DB Benchmark</b>	-	Benchmark ISBSG (≈ 2000 progetti DEV/MEV al 2003)	-	-
<b>Livello di dettaglio della misurazione</b>	dipendente dalla def. di LOC	Un solo strato "applicativo" definito in maniera standard dal modello IFPUG	Possibilità di definire uno o più livelli di dettaglio in base alle esigenze e cooscenze	Possibilità di definire un livello di dettaglio in base alle esigenze



# metodi di “backfiring” e tabelle di conversione LOC/FP

(USA, Capers Kones, anni '90)



**FP?**

**Statistiche valide (forse) su realtà locali, ma soprattutto...**

**Oggetti di misura differenti!**



- Uno stesso modello di business può essere rappresentato con diversi modelli UML alternativi tra loro, e i diversi modelli alternativi, anche se riferiti alla stessa applicazione, possono portare a conteggi diversi.
  - Non risulta conveniente "forzare" la tecnica UML con delle regole apposite, per rendere meno incerto il passaggio dagli oggetti UML agli oggetti FP.
- 
- **Si rivelano invece utili delle linee Guida per la redazione e l'interpretazione dei diagrammi UML...**
  - **L'attività di conteggio può diventare un ottimo pretesto per verificare e migliorare la documentazione di progetto**





- **Prezzo fisso a FP (Italia, anni '90, contratti quadro)**  
**Totale[€] = FP × Prezzo\_unitario[€]**  
**con Prezzo\_unitario[€] fissato per qualsiasi progetto a contratto**
- **L'approccio del prezzo fisso a FP si è rivelato un boomerang per la diffusione delle pratiche di misurazione nei contratti "multiprogetto" o aperti (pratica del backfiring dai costi)**
- **Soluzioni**
  - **Scorporo dei fattori d'impatto dalla metrica funzionale reale**
  - **Utilizzo di framework contrattuali**

# Conclusioni: presente e futuro

- **Evoluzione dei metodi di misurazione (IFPUG, COSMIC)**
- **Ricerca sull'utilizzo previsionale**
  - **Modelli COCOMO-like, ISBSG-based, ...**
- **Ricerca sull'utilizzo contrattuale**
  - **Futuro documento GUFPI/CNIPA**
- **Iniziative metriche particolari**
  - **Patrimonio SW & Patrimonio ICT**
  - **Benchmarking e individuazione esatta dei fattori d'impatto**
  - **Linee guida per specifici ambienti, dominî, tipologie: Data Warehouse, Middleware, UML, Web, ...**

# Documentazione e link

- **Counting Practices Manual IFPUG (4.1.1, 2000)**
- **Measurement Manual COSMIC (2.2, 2003)**
- **Publicazioni IFPUG ([www.ifpug.org](http://www.ifpug.org))**
  - **Case studies & papers**
- **Publicazioni GUFPI-ISMA ([www.gufpi.org](http://www.gufpi.org))**
  - **Linee Guida Italiane per il conteggio dei FP IFPUG**
- **Publicazioni COSMIC ([www.cosmicon.com](http://www.cosmicon.com))**
  - **Case studies & papers**
- **Publicazioni ISBSG ([www.isbsg.org](http://www.isbsg.org)[.au])**
  - **Benchmark produttività/dimensione**
- **Tecnica di stima Early & Quick FP ([www.dpo.it](http://www.dpo.it))**
  - **FP e UML**
  - **FFP COSMIC**
- **Software Measurement European Forum ([www.dpo.it/smef2004](http://www.dpo.it/smef2004))**

Grazie per l'attenzione  
**Grazie per l'attenzione**

**[tommaso.iorio@dpo.it](mailto:tommaso.iorio@dpo.it)**