

Function Point Analysis (IFPUG 4.1.1)

Ing. Thimoty Barbieri

settembre 2002

Indice

INTRODUZIONE	5
I FUNCTION POINT	6
OBIETTIVI E PRINCIPI CARDINE DELLA FPA.....	8
PUNTO DI VISTA DELL'UTENTE.....	9
APPROSSIMAZIONE E MISURAZIONE.....	10
IL CONTEGGIO DEI FUNCTION POINT	11
TIPO DI CONTEGGIO.....	12
IDENTIFICAZIONE DELL'AMBITO DI CONTEGGIO E DEL CONFINE APPLICATIVO.....	13
UNADJUSTED FUNCTION POINT: I FUNCTION TYPE.....	14
ESEMPIO.....	16
CARATTERISTICHE GENERALI.....	17
INDIVIDUAZIONE E CALCOLO DEI FUNCTION TYPE	19
CONTEGGIO DELLE FUNZIONI DATI.....	20
INTERNAL LOGICAL FILE	21
REGOLE DI IDENTIFICAZIONE DEL PROCESSO ELEMENTARE	23
CRITERI PER IL CONTEGGIO.....	24
EXTERNAL INTERFACE FILE	25
CRITERI PER IL CONTEGGIO.....	26
VALUTAZIONE DELLA COMPLESSITÀ	27
REGOLE DI CONTEGGIO DET/RET	28
ESEMPIO DI CONTEGGIO ILF/EIF	29
CONTEGGIO DELLE FUNZIONI TRANSAZIONALI	32
EXTERNAL INPUT	33
CARATTERISTICHE DEL TRATTAMENTO LOGICO.....	35
CRITERI PER IL CONTEGGIO.....	36

VALUTAZIONE DELLA COMPLESSITÀ.....	38
REGOLE PER CONTEGGIO DET/RTF DI EI	39
ESERCIZIO DI CONTEGGIO EI	41
EXTERNAL OUTPUT.....	42
REGOLE IFPUG PER IL CONTEGGIO DI EO	43
CRITERI PER IL CONTEGGIO.....	44
VALUTAZIONE DELLA COMPLESSITÀ.....	45
REGOLE PER CONTEGGIO DET/RTF DI EO	46
ESEMPIO DI CONTEGGIO EO	48
EXTERNAL INQUIRY.....	49
REGOLE IFPUG PER CONTEGGIO EQ	50
CRITERI PER IL CONTEGGIO.....	51
VALUTAZIONE DELLA COMPLESSITÀ.....	52
REGOLE CONTEGGIO DET/RTF PER EQ.....	53
ESEMPIO DI CONTEGGIO EQ	55
VALUTAZIONE DELLE CARATTERISTICHE GENERALI DEL SISTEMA.....	56
COMUNICAZIONE DATI.....	57
DISTRIBUZIONE DELL'ELABORAZIONE.....	58
PRESTAZIONI	59
UTILIZZO INTENSIVO CONFIGURAZIONE	60
FREQUENZA DELLE TRANSAZIONI.....	61
INSERIMENTO DATI INTERATTIVO	62
EFFICIENZA PER L'UTENTE FINALE.....	63
AGGIORNAMENTO INTERATTIVO	65
COMPLESSITÀ ELABORATIVA	66
RIUSABILITÀ.....	67
FACILITÀ DI INSTALLAZIONE.....	68
FACILITÀ DI GESTIONE OPERATIVA	69
MOLTEPLICITÀ DI SITI.....	70
FACILITÀ DI MODIFICA.....	71
CALCOLO DEI FUNCTION POINT.....	73
TIPI DI CALCOLO FINALE.....	74
ESERCIZIO DI CONTEGGIO.....	76
ESECUZIONE CONTEGGIO – SCHEMA	77
ESECUZIONE CONTEGGIO.....	78

CASO DI STUDIO – GESTIONE CONTATTI.....	80
CASO DI STUDIO – PROJECT MANAGEMENT.....	83
CONTEGGIO A PARTIRE DA GUI.....	91
IL CONTEGGIO NEL PARADIGMA OOP	105
CONTEGGIO DI WEB APPLICATION.....	107
DALLA STIMA DELLA DIMENSIONE ALLA STIMA DELLO SFORZO: COCOMO II.....	109
FORMULE BASE DI COCOMO II.....	110
DA FP A KSLOC.....	112
CONFIGURAZIONE DEL MODELLO COCOMO II.....	113
TEMA D’ESAME PER LA CERTIFICAZIONE CFPS..	114
BIBLIOGRAFIA E LINK UTILI	115

Introduzione

I Function Point (FP) misurano il software quantificando le funzionalità fornite verso l'esterno, basandosi principalmente sul disegno logico del sistema. Tenendo conto di tale fatto si può ricordare che gli obiettivi del conteggio con i FP sono:

- ?? Valutare ciò che viene chiesto dall'utente e ciò che gli viene fornito
- ?? Misurare un software indipendente dalla tecnologia utilizzata per lo sviluppo
- ?? Fornire una metrica dimensionale tale da supportare studi di qualità e di produttività
- ?? Fornire una base per studi relativi alla preventivazione e stima
- ?? Fornire un fattore di normalizzazione che permetta il raffronto tra sistemi di organizzazioni diverse

Per soddisfare i requisiti sopra citati, il processo di misurazione dovrebbe essere:

- ?? Abbastanza semplice, per diminuire il sovraccarico del processo di misurazione
- ?? Conciso, per garantire la congruenza tra tempo, progetti e gestione del personale applicativo.

I Function Point

Cenni sul conteggio dei Function Point

Premessa

L'introduzione delle prime metriche per i prodotti software risale a più di 20 anni fa, grazie ai lavori di ricerca del prof. M. Halstead, il quale sosteneva che le caratteristiche di un sistema o di parte di esso possono essere predette contando gli oggetti che ne compongono il codice sorgente. Negli anni '70 McCabe aggiunse a tale metrica dimensionale una metrica strutturale basata sulla caratterizzazione del flusso di controllo di un programma o di una subroutine in termini di un numero, detto cicomatico, che ne esprime la complessità. Mentre la metrica strutturale viene per lo più utilizzata per misurare la manutenibilità dei programmi, la metrica dimensionale è utilizzabile secondo il modello Co.Co.Mo. di Bohem che, partendo dalla stima di quanti statement COBOL compongono una procedura, ne predice impegno e tempo di sviluppo richiesti.

Tale metodo ha due punti di debolezza. Il primo è che non è assolutamente facile stimare gli statement di cui si comporrà un'applicazione nelle fasi alte dell'analisi; il secondo è che le stime sono fortemente condizionate dal modo di sviluppare tipico dei linguaggi procedurali e che quindi tale modello non è agevolmente utilizzabile per procedure sviluppate con linguaggi superiori a quelli di terza generazione.

Per ovviare a questi inconvenienti è necessario quindi

adottare una metrica che:

- ?? sia il più possibile indipendente dalla tecnologia hardware e software utilizzata;
- ?? sia facilmente utilizzabile a partire dalle fasi "alte" del processo di sviluppo software, ai fini di una previsione degli impegni di realizzazione e di un controllo del processo stesso.

Una metrica che risponde a queste caratteristiche è quella dei cosiddetti "Function Point" o "Punti Funzione", introdotta nel 1984 in ambito IBM da Allan J. Albrecht. Tale metrica è infatti svincolata dal sistema di sviluppo ed è possibile associarvi un modello di stima simile a Co.Co.Mo. verificabile praticamente in ogni fase del progetto.

I Function Point misurano le funzionalità di cui si compone un'applicazione in termini di elementi oggettivi ed enumerabili anche dall'utente finale (transazioni di input e output, tabulati, inquiries e entità fondamentali) e rappresentano uno standard che si va sempre più affermando a livello professionale.

A conferma di ciò, l'International Function Point User Group (IFPUG) ha pubblicato la release 4.1.1 del "Function Counting Practices Manual" che rappresenta il punto di riferimento per la definizione formale ed il conteggio dei FP.

Tale misurazione consente, tra l'altro, di osservare le eventuali modifiche che incidono sulla quantità di oggetti da produrre e di avere quindi un quadro più completo degli scostamenti dal piano di sviluppo originale e delle loro motivazioni.

Obiettivi e Principi Cardine della FPA

L'analisi a Function Point è un metodo standard per misurare lo sviluppo del software dal punto di vista dell'utente.

Essa quantifica le funzionalità del software basandosi principalmente sulla progettazione logica, con l'obiettivo di:

- misurare le funzionalità che l'utente richiede e riceve
- misurare la manutenzione e lo sviluppo del software in maniera indipendente dal tipo di tecnologia utilizzata (host monolitico, CS, Web...)
- Il conteggio deve essere coerente tra tutti i progetti e tra tutte le organizzazioni che utilizzano questa metodologia (in modo da consentire la creazione di repositories di conteggio tra categorie, raffronti tra le prestazioni di varie aziende, etc.)

Punto di vista dell'Utente

La Vista Utente (User View) rappresenta una descrizione formale delle esigenze applicative dell'utente, espressa nel linguaggio dell'utente.

- è una descrizione delle funzioni "business"
- è approvata dall'utente
- può essere utilizzata per il conteggio degli FP
- ha varie forme (cartacea, requisiti, specifiche interne, manuali procedure interne...)

Le specifiche dei requisiti hanno fasi di vita: Requisiti Utenti Iniziali, Requisiti Tecnici Iniziali, Requisiti Funzionali Finali. Inizialmente i requisiti sono espressi con una terminologia non compatibile con gli FP, e contenente descrizioni non corrette dal punto di vista tecnico. Successivamente i tecnici revisionano questi requisiti, producendo un documento troppo legato alla tecnologia, e con un linguaggio non familiare agli utenti. L'incontro tra tecnici e utenti produce il documento finale, che contiene terminologia comune, descrive in modo integrato tutti i requisiti utente, ed è sufficientemente preciso per un conteggio FPA.

Approssimazione e Misurazione

Il conteggio può essere

- 1) Approssimato o può essere una vera e propria
- 2) Misurazione

Nel ciclo di vita del software, a livello di Proposta (l'utente esprime le proprie intenzioni) è possibile approssimare gli FP, ma non effettuare un vero e proprio conteggio.

Agli altri livelli è possibile fare sia una rapida approssimazione, che eseguire un conteggio preciso:

- Requisiti, dove gli utenti e gli sviluppatori sono in accordo sull'espressione delle esigenze applicative
- Progettazione, dove gli sviluppatori includono gli elementi implementativi, non necessari al conteggio FP
- Costruzione
- Consegna
- Manutenzione Evolutiva

Il conteggio dei Function Point

La metrica dei FP misura un'applicazione basandosi su due aree di valutazione.

La prima è basata sul valore degli Unadjusted Function Point (UFP), che riflette le funzionalità fornite all'utente dell'applicazione; la seconda valutazione fornisce il valore dei Fattori di Aggiustamento, che valorizza la complessità delle funzionalità generali fornite all'utente.

Affinando il conteggio ottenuto dalla prima fase con il Fattore del Valore di Aggiustamento ottenuto dalla seconda fase, viene prodotto il conteggio dei Function Point finale.

La procedura generale da seguire per il conteggio è la seguente:

1. Determinare il Tipo di Conteggio
2. Identificare l'Ambito di Conteggio e i Confini Applicativi
3. Identificare tutte le funzioni dati
4. Identificare tutte le funzioni transazionali
5. Eseguire il conteggio unadjusted
6. Determinare il valore del fattore correttivo attraverso le Caratteristiche Generali del Sistema
7. Finalizzare il calcolo usando il fattore correttivo

Tipo di Conteggio

Il primo passo consiste nel determinare il tipo di conteggio funzionale, legato allo stato in cui l'applicazione conteggiata si trova.

1. Conteggio per Sviluppo di Progetto (Development Project)
 - a. Misura le funzionalità fornite agli utenti finali alla prima installazione del sistema. Questo conteggio va aggiornato al procedere dell'analisi del sistema, con l'aumento delle funzionalità (scope creep). Si conteggiano anche le funzionalità di conversione.
2. Conteggio Manutenzione Evolutiva (Enhancement)
 - a. Misura le modifiche ad un'applicazione esistente, e comprende le funzionalità fornite agli utenti con l'aggiunta di nuove funzioni, cancellazione di vecchie funzioni, e variazione di funzioni esistenti. Le funzionalità di conversione esistono anche nei progetti di manutenzione evolutiva.
3. Conteggio Applicativo (Applicazione)
 - a. Misura un'applicazione già installata. Questo conteggio, detto anche baseline, valuta le funzionalità attualmente fornite all'utente.

Identificazione dell'Ambito di Conteggio e del Confine Applicativo

L'ambito di conteggio è determinato dallo scopo del conteggio: esso identifica i sistemi, le applicazioni o i sottoinsiemi dell'applicazione che deve essere conteggiata.

Per confini dell'applicazione si intende la linea di divisione tra l'applicazione o il progetto che si vuole misurare e le altre applicazioni, e tra l'applicazione misurata e l'utente. Una tale divisione è importante al fine di individuare quali processi sono proprietari delle informazioni e dati che verranno trattati durante il conteggio.

Al fine di individuare i confini del progetto o applicazione si deve guardare al sistema dal punto di vista utente, considerando le funzionalità che l'utente può capire e descrivere dell'applicazione in esame; tali funzionalità verranno valutate in termini di FP. Tale informazione può essere reperita sia dalle specifiche funzionali del sistema che da un diagramma di contesto al più alto livello.

Alcune regole specifiche IFPUG:

- **il confine si basa sulla vista utente** (il confine è definito nei termini del business e del linguaggio dell'utente)
- **il confine si basa su considerazioni legate all'ambito del business più che a considerazioni tecnologiche**

Unadjusted Function Point: i Function Type

Nel conteggio degli UFP le funzionalità vengono valorizzate considerando il "cosa" viene consegnato all'utente e non il "come"; vengono, infatti, conteggiati solo i componenti richiesti dall'utente e a questo visibili. Tali componenti vengono distinti in Function Type, tali Function Type possono essere Dati o Transazioni.

Dati:

- ?? Internal Logical File (ILF) – aggregati logici di dati gestiti internamente al sistema
- ?? External Interface File (EIF) – aggregati logici di dati scambiati o condivisi dall'applicazione con altre applicazioni

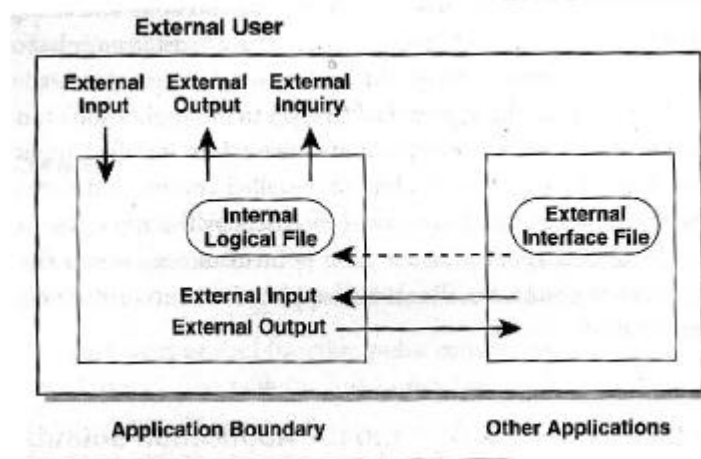
Transazioni:

- ?? External Input (EI) – informazioni distinte fornite dall'utente e usate dal sistema come dati in ingresso
- ?? External Output (EO) – output distinti che il sistema ritorna all'utente come risultato delle proprie elaborazioni
- ?? External Inquiry (EQ) – interrogazioni in linea che producono una risposta immediata dal sistema

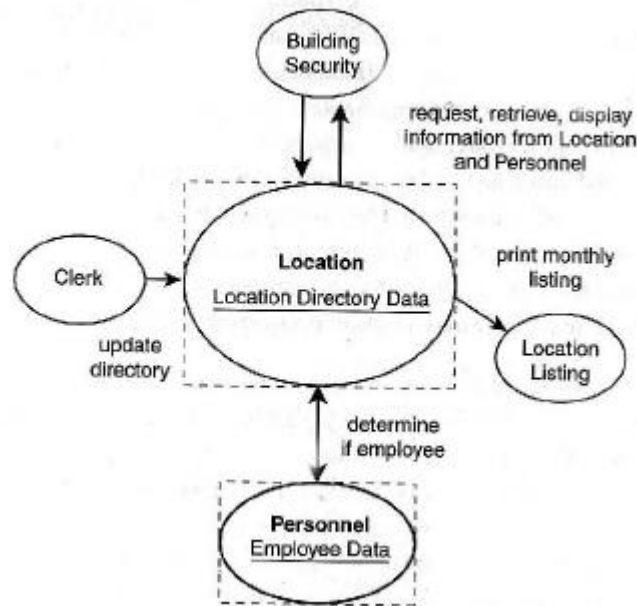
La complessità di ogni Function Type è valutata su una scala nominale di tre punti: Basso, Medio, Alto.

La sommatoria dei Function Type (il cui valore può variare da 3 a 15, in dipendenza della loro complessità) porta alla valorizzazione degli Unadjusted Function Point.

La seguente figura mostra una schematizzazione delle funzioni dati e delle funzioni transazionali



Esempio



Nell'applicazione per la gestione del database delle località:

Funzioni Dati

ILF: il database delle località, EIF: il database del personale

Funzioni Transazionali

EI: aggiornamento dell'impiegato del database località

EO: Lista stampata mensile delle località

EQ: Richiesta caratteristiche località dal rep. Sicurezza; sincronizzazione del database del personale

Caratteristiche generali

Il value Adjustment Factor consta di 14 sotto-fattori, detti General System Characteristic (GSC), che vengono utilizzati al fine di valutare la complessità generale del sistema. Le 14 caratteristiche generali sono:

1. Comunicazione Dati
2. Distribuzione dell'elaborazione
3. Prestazioni
4. Utilizzo intensivo configurazione
5. Frequenza delle transazioni
6. Inserimento dati interattivo
7. Efficienza per l'utente finale
8. Aggiornamento interattivo
9. Complessità elaborativa
10. Riusabilità
11. Facilità di Installazione
12. Facilità di gestione operativa
13. Molteplicità di Siti
14. Facilità di Modifica

Tali caratteristiche vengono valutate utilizzando una scala di sei valori, o scala del grado di influenza (DI):

0. non presente, o nessuna influenza
1. influenza poco significativa
2. influenza moderata
3. influenza media
4. influenza significativa
5. influenza massima (fattore determinante)

Per alcune delle GSC il metodo di valutazione del DI è diverso (invece di quotare l'influenza si assegna il punteggio per somma di presenze di alcune sotto-caratteristiche)

Per una descrizione più dettagliata della valutazione di tali fattori si rimanda alla sezione Valutazione delle caratteristiche generali del sistema.

Il computo delle GSC consente di cambiare il valore del conteggio dei function points non pesato (unadjusted) del +/- 35%.

Individuazione e calcolo dei Function Type

L'individuazione precisa e definitiva dei Function Points può avvenire soltanto nel momento in cui l'applicazione software è stata completata.

Rispetto alle linee di codice, però, la natura stessa dei Function Point permette di effettuarne una stima del numero già nelle fasi "alte" del processo di sviluppo. In particolare, al termine della Fase di Requisiti Funzionali è possibile derivare i Function Point tramite l'analisi di alcuni oggetti del documento di Specifiche funzionali. Vedremo in dettaglio come è possibile effettuare questa operazione per ogni tipologia di Function point.

Conteggio delle Funzioni Dati

Le funzioni dati sono ILF e EIF. Entrambi sono gruppi logici di informazione, e vanno conteggiati secondo le loro caratteristiche logiche e non l'implementazione fisica (n. di tabelle, etc.). Gli ILF sono sempre mantenuti e aggiornati all'interno del confine applicativo conteggiato. Gli EIF possono solo essere letti o referenziati dall'interno del confine applicativo. Gli EIF sono sempre mantenuti/aggiornati al di fuori del boundary.

Solitamente la prima fase prevede il dimensionamento delle funzioni dati (ILF ed EIF) per i seguenti motivi:

- Occorre sapere quali ILF ed EIF sono mantenuti o referenziati dalle funzioni transazionali per assegnare loro una valutazione di complessità, in base alle matrici standard.
- Identificando prima le basi dati, quando si procede a valutare le transazionali è possibile convalidare la scelta di classificazione come ILF o EIF effettuata

Internal logical file

E' detto Internal Logical File (ILF) ogni raggruppamento logico identificabile dall'utente di dati o di informazioni controllo che viene mantenuto (generato, usato e gestito) all'interno dei confini dell'applicazione in esame. L'intento primario di un ILF è di tenere aggiornati i dati mediante uno o più processi elementari dell'applicazione conteggiato.

Glossario:

identificabile dall'utente = si riferisce ai requisiti definiti dall'utente insieme allo sviluppatore e dall'utente riconosciuti come tali

raggruppamento logico = ogni raggruppamento deve essere coeso rispetto alla propria definizione.

Dati = l'insieme di fatti e di cifre mantenuti all'interno dell'applicazione

Informazioni di controllo = dati usati dall'applicazione per influenzare un processo elementare dell'applicazione conteggiata

Mantenuto = i dati sono modificati usando un processo elementare dell'applicazione

Processo Elementare = è la più piccola unità di attività significativa per l'utente

Un Internal Logical File è assimilabile ad una "entità", cioè ad ognuno degli oggetti trattati dall'applicazione. Un internal logical file è sempre accessibile all'utente attraverso un external input, output o inquiry.

Tipicamente sono riconducibili a uno o più ILF in dipendenza della visione utente:

?? data base o parti di essi

?? tabelle create e gestite dall'utente dell'applicazione

?? file/tabelle di messaggi

Regole IFPUG

Perché dei dati o delle informazioni di controllo siano contate come ILF deve valere:

- **I dati costituiscono un raggruppamento logico identificabile dall'utente**
- **I dati costituiscono un raggruppamento mantenuto attraverso un processo elementare all'interno del confine applicativo**

Regole di identificazione del processo elementare

L'individuazione di un processo elementare e delle sue caratteristiche è uno dei criteri per capire se alcuni punti funzioni vanno contati più di una volta o meno.

In generale:

- Un processo elementare è la più piccola unità di attività significativa per l'utente

Occorre cioè considerare il processo percepito come "macrofase" di specifica, e non come istruzione o metodo a livello di programma.

- Il processo deve essere autonomo e lasciare l'applicazione misurata in uno stato di coerenza funzionale

Significa che non è possibile considerare processo elementare un processo che viene forzatamente innescato da un altro processo per il suo completamento, o un processo che non è completato logicamente perché deve chiamare ulteriori operazioni.

Criteri per il conteggio

Sono da considerare come Internal Logical File

- ?? una entità dal punto di vista utente
- ?? ogni file logico generato o gestito dall'applicazione
- ?? Dati di sicurezza/password mantenuti nell'applicazione
- ?? Dati legati ad help utente se non già built-in nel sistema operativo
- ?? Dati per gestione di parametri dell'applicazione
- ?? File di errori con relative descrizioni

NON sono da considerare ILF:

- File temporanei e di appoggio (per calcolo o ordinamenti), questi non sono mai contati in FPA
- Copie ridondanti dello stesso file
- Indici/Join/Viste alternative (implementativo)
- Dati di audit o storici
- Dati di backup a livello corporate per scopo di recovery
- File sospesi per transazioni non completate (tecnico)

External interface file

E' detto External Interface File (EIF) ogni raggruppamento logico identificabile dall'utente di dati o di informazioni di controllo utilizzati (referenziati) dall'applicazione in esame ma mantenuti (gestiti) da altre applicazioni. L'intento primario di un EIF è di mantenere dati referenziati attraverso uno o più processi elementari nel confine applicativo. Un EIF contato per una applicazione deve essere un ILF contato in un'altra applicazione.

Regole IFPUG

- **il raggruppamento è logico e secondo utente**
- **il raggruppamento è esterno e referenziato dall'applicazione**
- **il raggruppamento non è mantenuto dall'applicazione conteggiata**
- **il raggruppamento è mantenuto in quanto ILF da un'altra applicazione**

Criteri per il conteggio

Sono da considerare come External Interface File

- ?? ogni file referenziato da un'altra applicazione
- ?? messaggi di help mantenuti fuori dall'applicazione
- ?? messaggi di errore mantenuti fuori dall'applicazione

Non sono da considerare come External Interface File

- ?? dati ricevuti da un'altra applicazione che aggiungono, cambiano o cancellano dati su un ILF
- ?? dati gestiti dall'applicazione in esame ed utilizzati da altre applicazioni (questi sono EIF dell'applicazione utilizzante e non in esame)
- ?? file temporanei/di appoggio/di sorting/join/view/indice esterni, questi non sono mai conteggiati in FPA perché derivano da esigenze implementative-tecnologiche

Valutazione della complessità

La valutazione della complessità funzionale dei dati si fonda su due elementi: i Record Element Type (RET) e i Data Element Type (DET).

?? (RET) numero di record logico/fisici da cui è costituito un dato, in modo unico, non ripetibile, e riconoscibile dall'utente

?? (DET) sottogruppi riconoscibili dall'utente di elementi di dati contenuti in un ILF o EIF. I sottogruppi sono tipicamente rappresentati in un diagramma ER come sottotipi di entità o entità attributive.

Nel seguito è riportata la tabella per la valutazione del livello di complessità di un ILF/EIF.

	Data Element Types		
RET	1-19	20-50	>=51
1	Basso	Basso	Medio
2-5	Basso	Medio	Alto
>5	Medio	Alto	Alto

Regole di conteggio DET/RET

Regole IFPUG DET

1. contare un DET per ogni campo unico, riconoscibile dall'utente, non ripetuto in ILF o EIF
2. se due o più applicazioni mantengono/referenziano lo stesso ILF/EIF, contare solo i DET usati dall'applicazione per dimensionare l'ILF/EIF.
3. Contare un DET per ogni dato necessario all'utente per stabilire una relazione con un altro ILF/EIF. (Foreign Key)
4. Campi che compaiono più di una volta per motivi implementativi/tecnologici ma logicamente identici, vanno contati una sola volta
5. Campi ripetuti per consentire l'immissione di più valori della stessa voce sono contati come un solo DET

Regole IFPUG RET

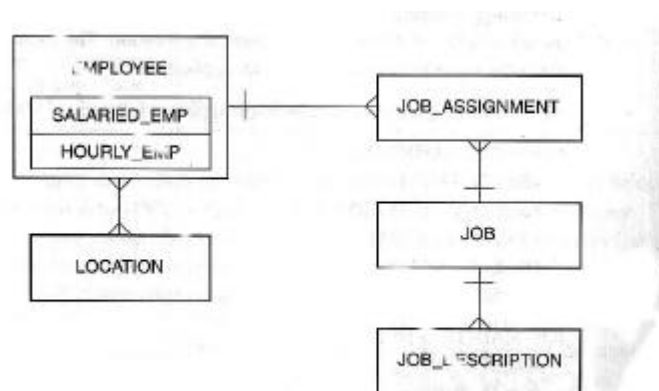
1. Contare un RET per ogni sottogruppo opzionale o obbligatorio dell'ILF o dell'EIF. Non contare RET la cui esistenza è legata ad esigenze implementative (headers, file di passaggio, ecc.).
2. Se non ci sono sottogruppi, contare l'ILF/EIF come 1 RET.

Esempio di conteggio ILF/EIF

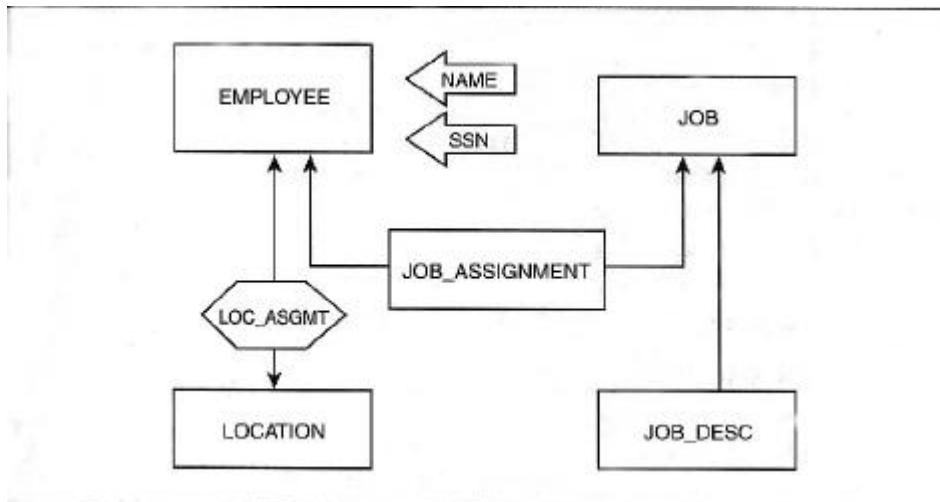
Si consideri la seguente applicazione di esempio:

Un gestionale personale che consenta di:

- mantenere, interrogare e riportare informazioni sul personale (employee). Il rapporto comprende i dati di locazione di determinato personale prelevato da file esterno
- mantenere, interrogare e riportare informazioni su lavori, con descrizione dei lavori
- mantenere, interrogare e riportare sugli assegnamenti dei lavori agli impiegati
- interrogare e riportare sulle diverse locazioni presenti in azienda, compresa una lista di impiegati in una data locazione. I dati di locazione sono in sola lettura e mantenuti da un'altra applicazione. Modello E/R:



Schema della struttura del database:



Tracciato logico del database:

EMPLOYEE entity type (non si conta come RET perché esistono già due RET) - ILF

Employee_Name	DET 1
Social_Security_Number	DET 2
Nbr_Dependents	DET 3
Type_Code (Salaried or Hourly)	DET 4
Location_Name (Foreign Key)	DET 5

SALARIED_EMPLOYEE entity subtype	RET 1
Supervisory_Level	DET 6

HOURLY_EMPLOYEE entity subtype	RET 2
Standard_Hourly_Rate	DET 7
Collective_Bargaining_Unit_Number	DET 8

JOB entity type RET 1 - ILF

Job_Name	DET 1
Job_Number	DET 2
Pay_Grade	DET 3

JOB_DESCRIPTION entity type (staccato per ragioni implementazione, parte di JOB)

Job_Number (Foreign Key) (già contato nel DET 2)	
Line_Number (solo per implementazione, scartare)	
Description_Line	DET 4

JOB_ASSIGNMENT entity type	RET 1 - ILF
Effective_Date	DET 1
Salary	DET 2
Performance_Rating	DET 3
Job_Number (Foreign Key)	DET 4
Employee_SSN (Foreign Key)	DET 5
LOCATION entity type	RET 1 - EIF
Location_Name	DET 1
Address	DET 2
Interoffice_Code	DET 3

Elenco delle funzioni dell'applicazione:

EMPLOYEE-MAINTENANCE

CREATE-EMPLOYEE
 EMPLOYEE-INQUIRY
 UPDATE-EMPLOYEE
 DELETE-EMPLOYEE
 EMPLOYEE-REPORT

JOB-MAINTENANCE

CREATE-JOB
 JOB-INQUIRY
 UPDATE-JOB
 DELETE-JOB
 JOB-REPORT

JOB-ASSIGNMENT-MAINTENANCE

ASSIGN-EMPLOYEE-TO-JOB
 JOB-ASSIGNMENT-INQUIRY
 TRANSFER-EMPLOYEE
 EVALUATE-EMPLOYEE
 DELETE-ASSIGNMENT
 JOB-ASSIGNMENT-REPORT

LOCATION-REPORTING

LOCATION-INQUIRY
 LOCATION-REPORT

Conteggio delle Funzioni Transazionali

Le funzioni transazionali si distinguono in tre categorie:

- External Input (EI): si occupano di elaborare i dati in ingresso (per mantenere gli ILF) oppure le informazioni di controllo (per modificare il comportamento del sistema).
Intento Primario: mantenere uno o più ILF, o alterare il comportamento del sistema.
- External Inquiries (EQ): inviano dati al di fuori dell'applicazione recuperando dati o informazioni di controllo dagli ILF / EIF, senza alterare il comportamento del sistema o mantenere ILF, o eseguire calcoli.
Intento Primario: Presentare informazioni ad un utente
- External Output (EO): inviano dati al di fuori dell'applicazione per elaborazioni logiche diverse o in aggiunta rispetto al semplice recupero di dati o di informazioni di controllo.
Intento Primario: Presentare informazioni ad un utente, può anche eventualmente mantenere un ILF o alterare il comportamento del sistema.

External Input

E' detto External Input (EI) un processo elementare dell'applicazione che elabora dati o informazioni di controllo che entrano dall'esterno dei confini dell'applicazione in esame. I dati elaborati agiscono su uno o più ILF, le informazioni di controllo possono mantenere o non mantenere un ILF. L'intento primario di un EI è di mantenere uno o più ILF e/o modificare il comportamento della applicazione attraverso il suo trattamento logico.

Glossario

Trattamento logico = requisiti espressamente richiesti dall'utente per completare un processo elementare.

Regole IFPUG per EI

IFPUG specifica le seguenti regole:

Per il conteggio di elaborazione di DATI attraverso EI:

- I dati devono essere ricevuti dall'esterno del confine applicativo
- I dati in almeno un ILF devono essere mantenuti attraverso un processo elementare dell'applicazione
- Il processo deve essere autonomo e lasciare l'applicazione in

uno stato consistente

- La logica elaborativa deve essere unica o diversa da quella di altri EI; oppure il set degli elementi dei dati deve essere diverso da quello di altri EI; oppure gli ILF/EIF referenziati sono diversi da quelli referenziati da altri EI

Per il conteggio di elaborazione di INFO CONTROLLO attr. EI, come sopra, ma tenere presente che l'informazione di controllo può o non può mantenere un ILF/EIF.

Caratteristiche del Trattamento Logico

Il trattamento logico si può considerare di varie forme. Alcune di esse sono obbligatorie, possibili oppure vietate rispetto alle tre funzioni transazionali.

Tipo di Trattamento Logico	EI	EO	EQ
Eeguire validazioni	p	p	p
Eeguire formule matematiche e calcoli	p	o*	v
Convertire valori equivalenti	p	p	p
Filtrare e selezionare dati utilizzando criteri specifici per confrontare insiemi multipli di dati	p	p	p
Analizzare condizioni per determinare quali siano applicabili	p	p	p
Aggiornare almeno un ILF	o*	o*	v
Referenziare almeno un ILF o EIF	p	p	o
Reperire dati o informazioni di controllo	p	p	o
Creare dati derivati	p	o*	v
Alterare il comportamento del sistema	o*	o*	v
Preparare e presentare informazioni all'esterno del confine	p	o	o
Capacità di accettare dati o informazioni di controllo che entrano nel confine dell'applicazione	o	p	p
Ordinare o riorganizzare un insieme di dati	p	p	p

P = Possibile

O = Obbligatorio

O* = Obbligatorio Almeno uno

V = Vietato

Criteri per il conteggio

Sono da contare come EI

- ?? Dati transazionali usati per mantenere un ILF (una vendita, un appuntamento inserito, un modulo assicurativo compilato)
- ?? Ingressi che portino informazioni di controllo, es. Un sensore che rileva il superamento di una soglia termica
- ?? Messaggi da altre applicazioni che richiedono un'azione di elaborazione
- ?? Ingressi che mantengono un ILF
- ?? Dati file di applicazioni precedenti che entrano attraverso una conversione dentro un ILF di nuovo formato
- ?? Dati fisici che scatenano un processo, ad es. Un valore di temperatura

Non sono EI

- le richieste di input che sono parte di una query o di un output
- Schermi di menu che sono usati per navigare o selezionare, ma non mantengono un ILF
- Schermate di login per l'ingresso
- Metodi diversi per invocare la stessa logica elaborativa

- Meccanismi di puntamento e clic / drag per l'inserimento di dati, il refreshing o la cancellazione di dati esclusivamente a video
- Dati che passano tra una elaborazione in linea e demoni batch: infatti non attraversano il confine applicativo, idem per la componente client e server della stessa applicazione

Inserimenti che richiedono più schermate, sono conteggiati come un inserimento unico.

Valutazione della complessità

La valutazione del livello di complessità di un External Input si fonda su due elementi: i File Type Referenced (FTR) e i Data Element Type (DET)

?? (FTR) numero di archivi logici aggiornati (ILF) o coinvolti nell'elaborazione, anche in lettura (EIF).

?? (DET) numero di dati elementari introdotti, in qualità di campi o attributi (comprese le foreign key), che attraversano il confine dell'applicazione attraverso l'EI.

Nel seguito è riportata la tabella per la valutazione del grado di complessità di un External Input.

	Data Element Types		
FTR	1-4	5-15	>=16
<2	Basso	Basso	Medio
2	Basso	Medio	Alto
>2	Medio	Alto	Alto

Regole per Conteggio DET/RTF di EI

Regole specificate da IFPUG per il conteggio dei DET:

- contare un DET per ogni campo/attributo riconoscibile dall'utente, comprese le foreign key, che attraversano il confine applicativo per completare il processo elementare dell'EI
- Non contare un DET per un campo che non viene inserito dall'utente (non attraversa il confine), ma attraverso l'EI viene recuperato o derivato dall'applicazione e mantenuto su un ILF (es. una data generata da sistema, un numero di scaglione contabile calcolato, ecc.)
- Contare un solo DET per un campo logico formato da più campi ma richiesto all'utente sotto forma di un'unica informazione (ad es. più linee dello stesso indirizzo)
- Contare un DET per la capacità in generale di mandare un messaggio di risposta del sistema al di fuori del confine per confermare l'avvenuta elaborazione, o verificare che il processo deve continuare. Contare un DET anche se i messaggi sono più di uno.
- Contare un DET per la capacità in generale di specificare un'azione che l'EI deve intraprendere. Una sola anche se ci sono più metodi di invocazione. (ad es. Tasti funzione che richiedono una particolare azione da eseguire sull'EI).

Regole specificate da IFPUG per il conteggio degli FTR:

- Contare un FTR per ogni ILF mantenuto dal processo elementare legato all'EI
- Contare un FTR per ogni ILF o EIF letto durante l'elaborazione dell'EI
- Contare un solo FTR per ogni ILF che è sia mantenuto che letto dall'EI.

Esercizio di conteggio EI

Riprendendo l'esempio precedente, si consideri che tutte le transazioni di ingresso ritornino messaggi di errore (contare 1 DET in più per ogni EI) e che siano attivate da almeno un tasto funzione (contare 1 DET in più). Le funzioni di cancellazione lavorino avendo come ingresso solo la chiave primaria dell'ILF che viene mantenuto.

Le funzioni di assegnamento e di trasferimento non lavorano sul campo "Performance Rating", e quella di valutazione non modifica il campo "Salary".

Il conteggio EI risulta (cfr. tracciato logico):

EMPLOYEE-MAINTENANCE

CREATE-EMPLOYEE	10 DET (8 DET ILF + 2) – 2 FTR (employee e location)
EMPLOYEE-INQUIRY	
UPDATE-EMPLOYEE	10 DET (8 DET ILF + 2) – 2 FTR (employee e location)
DELETE-EMPLOYEE	3 DET (1 chiave + 2) – 2 FTR (employee e job_assignment)
EMPLOYEE-REPORT	

JOB-MAINTENANCE

CREATE-JOB	6 DET (4 DET ILF + 2) – 1 FTR (job)
JOB-INQUIRY	
UPDATE-JOB	6 DET (4 DET ILF + 2) – 1 FTR (job)
DELETE-JOB	3 DET (1 chiave + 2) – 2 FTR (job e jobassignment -integrità)
JOB-REPORT	

JOB-ASSIGNMENT-MAINTENANCE

ASSIGN-EMPLOYEE-TO-JOB	6 DET (5 DET – Perf + 2) , 3 FTR (Empl, JB, JB_ASS)
JOB-ASSIGNMENT-INQUIRY	
TRANSFER-EMPLOYEE	6 DET (5 DET – Perf + 2) , 3 FTR (Empl, JB, JB_ASS)
EVALUATE-EMPLOYEE	6 DET (5 DET – Salary + 2) , 3 FTR (Empl, JB, JB_ASS)
DELETE-ASSIGNMENT	4 DET (2 FK + 2), 1 FTR (JB_ASS)
JOB-ASSIGNMENT-REPORT	

LOCATION-REPORTING

LOCATION-INQUIRY	
LOCATION-REPORT	

External Output

E' detto External Output (EO) un processo elementare dell'applicazione che genera dati o informazioni di controllo che escono dal confine applicativo. L'intento primario di un EO è di presentare informazione ad un utente attraverso una logica elaborativa diversa, o aggiuntiva, rispetto al semplice recupero di dati o di informazioni di controllo. La logica elaborativa deve: o contenere una formula od un procedimento di calcolo, e creare dei dati derivati, oppure mantenere uno o più ILF, o alterare il comportamento del sistema, o combinazioni delle tre precedenti.

Glossario

Dati derivati = richiedono un'elaborazione ulteriore rispetto al semplice recupero dati, da uno o più ILF o EIF. Sono creati dalla trasformazione di dati esistenti per creare nuovi dati. Gli EO spesso contengono dati derivati.

Può considerarsi EO:

- a. ogni flusso di informazioni che lascia il confine dell'applicazione e che è il risultato di una elaborazione sui dati gestiti dall'applicazione stessa
- b. ogni richiesta di attivazione di funzione/transazione inviata ad un'altra applicazione
- c. ogni report, tabulato, messaggio diretto all'utente con calcoli

Regole IFPUG per il conteggio di EO

IFPUG specifica le seguenti regole per il conteggio degli EO:

- I dati devono uscire dal confine applicativo
- La logica elaborativa legata all'EO deve: contenere almeno un calcolo; oppure creare dati derivati; oppure mantenere almeno un ILF; oppure modificare il comportamento dell'applicazione.
- Il processo deve essere autonomo e lasciare l'applicazione in uno stato consistente
- La logica elaborativa deve essere unica o diversa da quella di altri EO; oppure il set degli elementi dei dati deve essere diverso da quello di altri EO; oppure gli ILF/EIF referenziati sono diversi da quelli referenziati da altri EO

Criteri per il conteggio

Sono da considerare come External Output:

- ?? Report su terminale o stampante che richiedono l'uso di algoritmi o calcoli
- ?? Trasferimenti di dati e o file ad altre applicazioni, quando i dati sono calcolati o derivati
- ?? Documenti di varia natura con calcoli in funzione del tipo di applicazione (fatture, bolle, assegni, modulistica in genere, ecc.)
- ?? Visualizzazione di grafici, se richiedono calcoli

NON considerare come EO:

- ?? Report identici con valori diversi (es. report uguali per reparti diversi)
- ?? Report che non hanno formule, conti o riepilogativi
- ?? File inviati ad altre applicazioni che non hanno formule, conti, o dati derivati, e che non mantengono ILF
- ?? Più supporti diversi se la logica elaborativa non è diversa
- ?? Più modi di invocare lo stesso processo di output
- ?? Messaggi di conferma che i dati sono stati elaborati
- ?? Report ad hoc creati dall'utente usando SQL

Valutazione della complessità

La valutazione del grado di complessità di un External Output si fonda su due elementi: i File Type Referenced (FTR) e i Data Element Type (DET)

?? (FTR) il numero di archivi logici coinvolti nella produzione dell'output

?? (DET) numero di dati elementari prodotti in uscita

Nel seguito è riportata la tabella per la valutazione del grado di complessità degli External Output.

	Data Element Types		
FTR	1-5	6-19	>=20
<2	Basso	Basso	Medio
2-3	Basso	Medio	Alto
>3	Medio	Alto	Alto

Regole per Conteggio DET/RTF di EO

Le seguenti regole valgono per contare i DET di un EO:

- Contare un DET per ogni campo riconoscibile dall'utente, che entra nel confine applicativo ed è necessario per specificare cosa, quando o come i dati devono essere prelevati o generati dal processo elementare
- Se un DET sia entra, sia esce dal confine, contarlo una sola volta per processo elementare
- Contare un DET per la capacità in generale di mandare ad un sistema un messaggio di risposta/avviso/sincronia fuori dal confine applicativo
- Contare un solo DET per la capacità in generale di specificare un'azione da intraprendere da parte dell'EO anche se ci sono più modi per invocare lo stesso processo logico
- Non contare variabili usate per la paginazione, come numeri di pagina e informazioni di posizionamento, o campi data
- Non contare informazioni alfanumeriche quali titoli di report, ID di schermate, intestazioni di colonne e titoli di campi
- Non contare campi che sono manutenti su ILF durante il processo elementare, se questi campi non attraversano il confine
- Contare un DET per un campo logico memorizzato fisicamente come più campi, ma richiesto all'utente come un'informazione unica (es. linee di indirizzo)

- Contare un DET per ogni tipo di etichetta e per ogni tipo di equivalente numerico in una visualizzazione grafica (ad es. 1 DET per visualizzare il titolo di una categoria in un grafico a torta, ed 1 DET per visualizzare la percentuale legata a quella categoria)
- Contare un DET per informazioni testuali che consistono di una parola, una frase, uno o più paragrafi.

Le seguenti regole IFPUG si applicano per contare gli FTR su EO:

- Contare un FTR per ogni ILF mantenuto dal processo elementare legato all'EO
- Contare un FTR per ogni ILF o EIF letto durante l'elaborazione dell'EO
- Contare un solo FTR per ogni ILF che è sia mantenuto che letto dall'EO.

Esempio di conteggio EO

Per conteggiare gli EO nell'esempio precedente si assuma quanto segue (le informazioni date non sono comunque sufficienti per fare un conteggio preciso):

- i report contengono calcoli e le inquiry sono tutte EQ
- i report hanno tra i 6 e i 19 DET che attraversano il confine
- il job report ha 5 DET

EMPLOYEE-MAINTENANCE EMPLOYEE-REPORT	6-19 DET, 2 FTR (employee, location)
JOB-MAINTENANCE JOB-REPORT	5 DET, 1 FTR (job)
JOB-ASSIGNMENT-MAINTENANCE JOB-ASSIGNMENT-REPORT	6-19 DET, 3 FTR (jobass, employee, job)
LOCATION-REPORTING LOCATION-REPORT	6-19 DET, 2 FTR

External inQuiry

Si dice External Inquiry (EQ) un processo elementare dell'applicazione che dà come risultato il prelievo di dati o informazioni di controllo inviate al di fuori del confine applicativo. L'intento primario è di presentare l'informazione ad un utente attraverso il recupero di dati da un ILF o un EIF. La logica elaborativa non contiene calcoli e non crea dati derivati. Non viene mantenuto nessun ILF durante l'elaborazione, ed il comportamento dell'applicazione non viene modificato.

E' External Inquiry (EQ) ogni combinazione di input/output, in cui l'input dà luogo ad un output immediato.

I dati di input servono a dirigere la ricerca e non sono oggetto di elaborazione (non concorrono alla variazione di un Internal Logical File)

E' essenziale ricordare che:

?? i dati in input vengono introdotti esclusivamente per consentire un accesso immediato alle informazioni prodotte in output, non determinano mai aggiornamenti ai dati gestiti dall'applicazione.

Regole IFPUG per conteggio EQ

IFPUG specifica le seguenti regole per il conteggio EQ:

- I dati devono uscire dai confini applicativi
- I dati devono essere prelevati da uno o più ILF/EIF
- La logica elaborativa Non deve creare dati derivati e non deve contenere un calcolo. Inoltre non deve alterare il comportamento dell'applicazione.
- Il processo non deve mantenere un ILF
- Il processo deve essere autonomo e lasciare l'applicazione in uno stato consistente
- La logica elaborativa deve essere unica o diversa da quella di altri EQ; oppure il set degli elementi dei dati deve essere diverso da quello di altri EQ; oppure gli ILF/EIF referenziati sono diversi da quelli referenziati da altri EQ

Criteri per il conteggio

Sono da considerare come External Inquiry

- ?? Dati transazionali prelevati da uno o più ILF/EIF e visualizzati
- ?? Funzioni utente per la visualizzazione, ricerca, stampa
- ?? Inquiry Implicite (recuperi di dati necessari prima di funzioni di modifica o cancellazione)
- ?? File inviati ad altre applicazioni che non contengono formule o prevedono conteggi

Non sono da considerare come External Inquiry

- ?? Metodi multipli che invocano la stessa logica (contati 1 v.)
- ?? Schermate di menu per navigazione o selezione (non si contano)
- ?? Messaggi di errore o di conferma
- ?? Documentazione di sistema in linea a meno che non sia esplicito oggetto di conteggio e implementazione

Valutazione della complessità

La valutazione del grado di complessità di un External Inquiry si fonda su due elementi: i File Type Referenced (FTR) e i Data Element Type (DET)

?? (FTR) il numero di archivi logici coinvolti nella produzione dell'output

?? (DET) numero di dati elementari prodotti in uscita

Nel seguito è riportata la tabella per la valutazione del grado di complessità degli External Inquiry.

	Data Element Types		
FTR	1-5	6-19	>=20
<2	Basso	Basso	Medio
2-3	Basso	Medio	Alto
>3	Medio	Alto	Alto

Regole conteggio DET/RTF per EQ

Le seguenti regole IFPUG valgono per il conteggio dei DET per gli EQ:

- Contare un DET per ogni campo riconoscibile dall'utente che entra nel confine applicativo ed è necessario per specificare cosa, quando o come recuperare i dati e/o generati dal processo elementare
- Contare un DET per ogni campo riconoscibile dall'utente che esce dal confine applicativo
- Se un DET sia entra che esce, contarli una sola volta
- Contare un DET per la capacità in generale di mandare ad un sistema un messaggio di risposta/avviso/sincronia fuori dal confine applicativo
- Contare un solo DET per la capacità in generale di specificare un'azione da intraprendere da parte dell'EQ anche se ci sono più modi per invocare lo stesso processo logico
- Non contare variabili usate per la paginazione, come numeri di pagina e informazioni di posizionamento, o campi data
- Non contare informazioni alfanumeriche quali titoli di report, ID di schermate, intestazioni di colonne e titoli di campi
- Contare un DET per ogni tipo di etichetta e per ogni tipo di equivalente numerico in una visualizzazione grafica (ad es. 1 DET per visualizzare il titolo di una categoria in un grafico a torta, ed 1 DET per visualizzare la percentuale legata a quella

categoria)

- Contare un DET per informazioni testuali che consistono di una parola, una frase, uno o più paragrafi.

Le seguenti regole IFPUG si applicano per contare gli FTR su EQ:

- Contare un FTR per ogni ILF o EIF letto durante l'elaborazione dell'EQ.

Esempio di conteggio EQ

Per il conteggio degli EQ si consideri quanto segue:

- per ogni EQ sono visualizzati messaggi di errore ed è disponibile un tasto funzione (+ 2 DET)

EMPLOYEE-MAINTENANCE EMPLOYEE-INQUIRY	10 DET (8 DET ILF + 2), 1 FTR
JOB-MAINTENANCE JOB-INQUIRY	6 DET (4 DET ILF + 2), 1 FTR
JOB-ASSIGNMENT-MAINTENANCE JOB-ASSIGNMENT-INQUIRY	7 DET (5 DET ILF + 2), 1 FTR
LOCATION-REPORTING LOCATION-INQUIRY	5 DET (3 DET ILF + 2), 1 FTR

Valutazione delle caratteristiche generali del sistema

Per calcolare la misura in Function Point di un'applicazione è necessario determinare un fattore di aggiustamento ricavato valutando il peso di ognuno dei seguenti elementi. Ogni Caratteristica Generale Sistema (GSC) deve essere valutata in termini di "Grado di Influenza" su una scala da 0 a 5. Per ogni GSC vengono fornite delle linee guida per la valutazione; nel caso nessuna delle descrizioni contenute nelle linee guida si dovesse adattare all'applicazione in esame, si dovrà ricorrere ad una valutazione soggettiva sul "grado di influenza".

Le Caratteristiche Generali sono:

1. Comunicazione Dati
2. Distribuzione Elaborazione
3. Prestazioni
4. Utilizzo intensivo configurazione
5. Frequenza delle transazioni
6. Inserimento dati interattivo
7. Efficienza per l'utente finale
8. Aggiornamento interattivo
9. Complessità Elaborativa
10. Riusabilità
11. Facilità Installazione
12. Facilità Gestione Operativa
13. Molteplicità di Siti
14. Facilità di Modifica

Comunicazione Dati

I dati e le informazioni di controllo, utilizzati nell'applicazione, sono ricevuti o inviati attraverso dispositivi di comunicazione.

Valori:

0 – applicazione che opera esclusivamente in ambiente batch o su PC stand alone

1 – applicazione batch ma con data entry o stampe remote

2 – applicazione batch ma con data entry e stampe remote

3 – raccolta dati on line o interfaccia ad una applicazione batch

4 – più che una semplice interfaccia, e l'applicazione gestisce solo un tipo di protocollo di comunicazione TP

5 – più che una semplice interfaccia, e l'applicazione gestisce più di un tipo di protocollo di comunicazione TP.

Note:

La maggior parte delle applicazioni (anche PC) hanno possibilità di data entry remoto e stampa. Se l'aggiornamento ILF avviene in batch, segnare 3. Se avviene in modo interattivo, segnare 4. Per avere 5, occorre la presenza di più protocolli di telecomunicazione (es. applicazione multicanale).

Distribuzione dell'elaborazione

La misura in cui l'elaborazione distribuita dei dati è una caratteristica dell'applicazione.

Valori:

- 0 – l'applicazione non prevede il trasferimento di dati o di funzioni di elaborazione tra i componenti del sistema
- 1 – l'applicazione prepara i dati per elaborazioni end user su un altro componente del sistema quale Foglio di calcolo su PC, o database su PC
- 2 – l'applicazione prepara i dati, li trasferisce e li processa su un altro componente del sistema (non sotto il controllo dell'utente)
- 3 – i processi di elaborazione e di trasferimento dati distribuiti sono on-line e in una sola direzione
- 4 – i processi di elaborazione e di trasferimento dati distribuiti sono on-line e in entrambe le direzioni
- 5 – le funzioni di elaborazione sono dinamicamente eseguite sul componente del sistema più appropriato.

Note: valutare solo per sistemi distribuiti o real-time.
Maggior parte 0. Cli/ser o web applications hanno da 2 a 4, per avere 5 sono necessari più server in bilanciamento in tempo reale.

Prestazioni

Gli obiettivi di Prestazioni dell'applicazione, richiesti o approvati dall'utente, sia in termini di tempo di risposta che di tempificazione delle elaborazioni, influenzano la progettazione, lo sviluppo, l'installazione e le attività di supporto dell'applicazione.

Valori:

0 – l'utente non ha specificato particolari requisiti di Prestazioni;

1 – l'utente ha specificato i livelli di Prestazioni nei requisiti del sistema, ma tali livelli non hanno richiesto attenzione speciale durante lo sviluppo;

2 – i tempi di risposta sono critici durante le ore di picco. Nessuno studio speciale per il carico della CPU è stato richiesto. La scadenza di elaborazione è giornaliera;

3 – i tempi di risposta dell'on-line sono critici durante l'intero arco della giornata. Nessuno studio speciale per il carico della CPU è stato richiesto. La scadenza di elaborazione dipende dall'interfaccia con altri sistemi;

4 – le Prestazioni richieste dall'utente sono stringenti al punto da richiedere una attività di analisi delle Prestazioni durante il disegno;

5 – l'ottimizzazione delle prestazioni riveste estrema criticità; strumenti per la valutazione delle prestazioni devono essere utilizzati in fase di progettazione, sviluppo ed installazione.

Utilizzo intensivo configurazione

Descrive il grado con cui le restrizioni sulle risorse informatiche a disposizione hanno influenzato lo sviluppo dell'applicazione. Nel disegno e nella realizzazione dell'applicazione si dovrà usare un'attenzione particolare per sopperire ai limiti imposti. Ad es. l'utente desidera eseguire l'applicazione su una particolare piattaforma esistente.

Valori:

0 – l'applicazione opera su sistemi standard di produzione con nessun tipo di vincoli

1 – esistono dei vincoli di tipo operativo, ma non è necessario uno studio speciale per superarle

2 – esistono dei vincoli relativi a problemi di sicurezza o temporizzazione

3 – esistono dei vincoli dovuti all'utilizzo di un tipo particolare di processore in una parte specifica del sistema

4 – la configurazione del sistema hardware pone vincoli all'applicazione

5 – in aggiunta, ci sono vincoli per l'applicazione su componenti distribuiti del sistema.

Nota: solitamente le applicazioni valgono 2. Per andare da 3 a 5, occorre un cli/ser oppure un sistema real time.

Frequenza delle transazioni

L'alto volume di transazioni da elaborare influenza la progettazione, lo sviluppo, l'installazione e le attività di supporto dell'applicazione.

Valori:

0 – non esistono dei tempi con picchi di carico riconosciuti

1 – presenza di picchi di carico di transazioni periodica (mensile, quadrimestrale, annuale, ecc).

2 – presenza di picchi di carico di transazioni settimanali

3 – presenza di picchi di carico di transazioni giornalieri

4 – il volume delle transazioni richiede, in fase di progettazione, particolari accorgimenti

5 – il volume di transazioni richiede l'adozione di tecniche e di strumenti specifici in fase di progettazione, sviluppo e installazione.

Inserimento dati interattivo

Funzioni di controllo e di data entry on-line sono presenti nell'applicazione.

Valori:

0 – applicazione esclusivamente batch

1 – dall' 1% al 7% delle transazioni sono di data entry interattivo

2 – dall' 8% al 15% delle transazioni sono di data entry interattivo

3 – dal 16% al 23% delle transazioni sono di data entry interattivo

4 – dal 24% al 30% delle transazioni sono di data entry interattivo

5 – oltre il 30% delle transazioni sono di data entry interattivo.

Nota: le transazioni si riferiscono agli EI, EO, EQ, ciascuno dei quali è un processo elementare.

Efficienza per l'utente finale

Esprime il grado di considerazione per i fattori umani e la facilità di utilizzo per l'utente dell'applicazione misurata. Le funzioni online sono state progettate evidenziando user-friendliness, ed utilizzando tecniche quali:

- ?? menu
- ?? help e documentazione on-line
- ?? movimento automatico del cursore
- ?? scrolling
- ?? stampe remote
- ?? tasti funzione
- ?? sottomissione di batch da transazioni on-line
- ?? selezione tramite cursore delle informazioni a video
- ?? forte utilizzo di campi ad alta intensità, sottolineati, o altrimenti indicati
- ?? uso del mouse
- ?? pop-up windows
- ?? navigazione facilitata tra schermate
- ?? supporto (help, documentazione, messaggi,...) in varie lingue diverse (conta come sei voci)
- ?? supporto per due lingue (conta come quattro voci)

Valori:

0 – nessuno degli elementi indicati è stato preso in considerazione

1 – uno, due o tre degli elementi indicati sono stati presi in considerazione

2 – quattro o cinque degli elementi indicati sono stati presi in considerazione

3 – sei o più degli elementi indicati sono stati presi in considerazione, ma non esistono richieste specifiche da parte dell'utente relativamente all'efficienza

4 – sei o più degli elementi indicati sono stati presi in considerazione, gli aspetti legati ad un utilizzo efficiente da parte dell'utente richiedono attenzione in fase di progettazione

5 – sei o più degli elementi indicati sono stati presi in considerazione, l'interfaccia utente gioca un ruolo determinante: l'utilizzo di strumenti per il prototyping è indispensabile.

Aggiornamento interattivo

Il grado per cui l'applicazione prevede l'aggiornamento on-line delle informazioni contenute negli Internal Logical File.

Valori:

0 – nessuno

1 – aggiornamento da uno a tre file di controllo. Il volume degli aggiornamenti è contenuto e il recovery è semplice

2 – aggiornamento di quattro o più file di controllo. Il volume degli aggiornamenti è contenuto e il recovery è semplice

3 – aggiornamento on-line degli internal logical file principali

4 – inoltre, devono essere previsti meccanismi di protezione dalla perdita dei dati

5 – volume degli aggiornamenti on-line tale da portare a considerazioni di costo nelle politiche di recovery. Procedure di recovery ad alta automazione con minimo intervento operatore.

Note: la maggior parte delle applicazioni online aggiornano ILF e quindi hanno punteggio 3 o superiore. Le applicazioni batch stanno tra 0 e 2. 4 per le applicazioni con sistemi di protezione dalla perdita dati. Con 5 si indicano sistemi con sistemi avanzati di protezione e recupero dei dati.

Complessità elaborativa

Descrive il grado in cui la complessità della logica elaborativa influenza lo sviluppo. Sono presenti 5 componenti:

- ?? elaborazioni logiche complesse
- ?? elaborazioni matematiche complesse
- ?? meccanismi per garantire la sicurezza dei dati
- ?? meccanismi per la gestione delle eccezioni e la completezza delle transazioni (con rollbacking)
- ?? meccanismi di elaborazione che gestiscono più possibilità di input output

Valori:

- 0 – nessuna delle caratteristiche citate è presente
- 1 – una delle caratteristiche citate è presente
- 2 – due delle caratteristiche citate sono presenti
- 3 – tre delle caratteristiche citate sono presenti
- 4 – quattro delle caratteristiche citate sono presenti
- 5 – tutte le caratteristiche citate sono presenti.

Riusabilità

L'applicazione e il codice dell'applicazione sono stati appositamente progettati e sviluppati per essere utilizzati in altre applicazioni.

Valori:

0 – il codice non è riusabile

1 – viene usato del codice riusabile nell'applicazione

2 – meno del 10% dei moduli prodotti considerano i bisogni di più di un utente

3 – più del 10% dei moduli prodotti considerano i bisogni di più di un utente

4 – l'applicazione è stata realizzata e documentata in modo tale da facilitare il riuso, la personalizzazione verso l'utente è solo a livello di codice sorgente

5 – l'applicazione è progettata e sviluppata per diventare generalizzata e usabile in contesti diversi.

Facilità di Installazione

La facilità di conversione e di installazione è una caratteristica dell'applicazione. Un piano di conversione e di installazione da precedenti piattaforme e/o strumenti di conversione sono stati approntati e collaudati durante la fase di collaudo finale.

Valori:

0 – nessuna richiesta utente di installazioni diverse e non è necessario un particolare set-up per altre installazioni del sistema

1 – nessuna richiesta utente di installazioni diverse ma è necessario un particolare set-up per altre installazioni del sistema

2 – conversione e installazione sono criticità evidenziate dall'utente; devono essere fornite e testate norme per l'installazione e per la conversione. L'impatto di una conversione sul progetto non è considerato importante

3 – conversione e installazione sono criticità evidenziate dall'utente; devono essere fornite e testate norme per l'installazione e per la conversione. L'impatto di una conversione sul progetto è considerato importante

4 – oltre al punto (2), devono essere realizzati e testati strumenti di conversione e installazione automatica

5 – oltre al punto (3), devono essere realizzati e testati strumenti di conversione e installazione automatica.

Facilità di Gestione Operativa

La semplicità operativa è una caratteristica dell'applicazione riguardo alla presenza di efficienti procedure di start-up, di back-up e di recovery, sviluppate e provate durante la fase di collaudo finale. L'applicazione minimizza la necessità di attività manuali quali i mount di nastri, la stampa di tabulati e gli interventi operativi per proseguire nell'elaborazione.

Valori:

0 – la gestione operativa dell'applicazione non è considerata critica da parte dell'utente

1 – 4 – selezionare le considerazioni che risultino valide per l'applicazione in esame, tra i seguenti elementi:

- sono fornite procedure di start-up, back-up e recovery ma è necessario l'intervento dell'operatore (contare un punto)
- sono fornite procedure di start-up, back-up e recovery ma non è necessario l'intervento dell'operatore (contare come due punti)
- l'applicazione minimizza il bisogno di montaggio nastri
- l'applicazione minimizza la gestione delle stampanti

5 – l'applicazione è stata disegnata per operazioni unattended. Unattended significa che il sistema può operare senza l'intervento dell'operatore; è necessario l'intervento umano solo per l'attivazione e la chiusura del sistema.

Molteplicità di Siti

L'applicazione è stata progettata e sviluppata in modo specifico per essere installata in più ubicazioni per differenti organizzazioni.

Valori:

0 – nessuna richiesta utente di necessità di più di una installazione

1 – la necessità di installazioni diverse è stata considerata durante il disegno ma l'applicazione può operare solo nello stesso ambiente hardware e software

2 – la necessità di installazioni diverse è stata considerata durante il disegno ma l'applicazione può operare solo in ambienti hardware e software simili a quello originario

3 – la necessità di installazioni diverse è stata considerata durante il disegno e l'applicazione può operare in ambienti hardware e software diversi da quello originario

4 – documentazione e piani di supporto sono forniti e testati per supportare l'applicazione in luoghi diversi, l'applicazione è così come descritta nei punti (1) e (2)

5 – documentazione e piani di supporto sono forniti e testati per supportare l'applicazione in luoghi diversi, l'applicazione è così come descritta nel punto (3).

Facilità di Modifica

L'applicazione è stata progettata e sviluppata in modo specifico per minimizzare gli impatti di modifiche. Per esempio:

- sono stati sviluppati o forniti strumenti di query generalizzati
- i parametri di controllo utente sono gestiti tramite tabelle aggiornabili direttamente dall'utente

Valori:

0 – la facilità di apportare modifiche all'applicazione è considerata inessenziale dall'utente

1 – 5 – selezionare tra i punti seguenti quelli che si adattano all'applicazione in esame:

- sono fornite facilitazioni query per la gestione richieste di query semplici; per esempio logiche and/or applicabili ad un solo Internal Logical File (contare 1 punto)
- sono fornite facilitazioni per la gestione richieste di query di media complessità; per esempio logiche and/or applicabili a più di un Internal Logical File (contare 2 punti)
- sono fornite facilitazioni per la gestione richieste di query di complessità elevata; per esempio combinazioni di logiche and/or applicabili a più di un Internal Logical File (contare 3 punti)

- i dati di controllo sono inseriti in tabelle che vengono gestite dall'utente tramite processi on-line interattivi, ma le variazioni diventano effettive solo a seguito di controllo
- i dati di controllo sono inseriti in tabelle che vengono gestite dall'utente tramite processi on-line interattivi, le variazioni diventano immediatamente effettive (contare 2 punti).

Calcolo dei function point

Il calcolo dei Function Point avviene seguendo tre passi:

1. valorizzare gli Unadjusted Function Point (UFP), contando le funzioni dati e transazionali secondo la tabella fornita.
2. valorizzare le caratteristiche generali (GCS) ed ottenere il TDI (Total Degree of Influence). Calcolare il VAF (Value Adjustment Factor) utilizzando la formula:

$$a. VAF = (TDI \times 0.01) + 0.65$$

3. valorizzare i Function Point (FP) utilizzando ad es. la formula :

$FP = UFP * VAF$ (per applicazioni conteggio iniziale, per altri tipi di conto vedi oltre)

Tabella del conteggio UFP

Componente	Basso	Medio	Alto
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

Tipi di calcolo finale

Esistono diverse tipologie di conteggio:

1. Conteggio PROGETTO DI SVILUPPO

Comprende il conteggio UFP dell'applicazione; il conteggio delle funzionalità di conversione dei vecchi dati file nei nuovi ILF; l'applicazione del VAF

$$\mathbf{DFP = (UFP + CFP) \times VAF}$$

DFP: conteggio finale del progetto di sviluppo

CFP: numero di function points delle funzioni di conversione

2. Conteggio MANUTENZIONE EVOLUTIVA

Comprende il conteggio UFP dell'applicazione con funzioni dati e transazionali date da:

- Aggiunte per manutenzione evolutiva (funzioni prima non esistenti)
- Cambiamenti per manutenzione (esistevano ma con campi, FTR, elaborazioni diverse)
- Cancellamenti per manutenzione

E date anche da funzionalità di conversione dei vecchi dati nei nuovi ILF; il VAF precedente (before) ed il VAF successivo (after) alla manutenzione (il VAF potrebbe cambiare alla successiva versione prodotta dalla manutenzione).

$$\mathbf{EFP = [(ADD + CHGA + CFP) \times VAFA] + (DEL \times VAFB)}$$

EFP : risultato conteggio evolutivo

ADD: UFP delle funzioni aggiunte nella manutenzione

CHGA : UFP delle funzioni modificate

CFP: FP per le funzioni di conversioni dati

VAFA: il VAF del sistema considerato dopo la manutenzione

DEL: gli UFP delle funzionalità cancellate

VAFB: il VAF del sistema considerato prima della manutenzione

3. Conteggio APPLICAZIONE

Conta gli UFP dell'applicazione, ed il VAF. Esiste in due varianti: INIZIALE, quando l'applicazione viene inizialmente consegnata; DOPO MANUTENZIONE, quando l'attività manutentiva ha aggiunto, cambiato, modificato funzioni dati ed il VAF.

INIZIALE: **AFP = ADD X VAF**

AFP: conteggio applicazione iniziale, ADD: UFP f. aggiunte

DOPO MANUTENZIONE:

$$\mathbf{AFP = [(UFPB + ADD + CHGA) - (CHGB + DEL)] \times VAFA}$$

UFPB: UFP prima manutenzione; ADD: UFP delle aggiunte in manutenzione; CHGA: UFP f. cambiate dopo manutenzione; CHGB: UFP f. cambiate prima manutenzione.

Esercizio di conteggio

Si consideri lo sviluppo iniziale di un'applicazione di Catalogo Commerciale. Esso consente di inserire, modificare, cancellare articoli, memorizzati in una tabella articoli interna all'applicazione. La tabella articoli dispone di una sola chiave e di oltre 20 campi per ciascun articolo. E' anche possibile per l'utente recuperare a video le descrizioni.

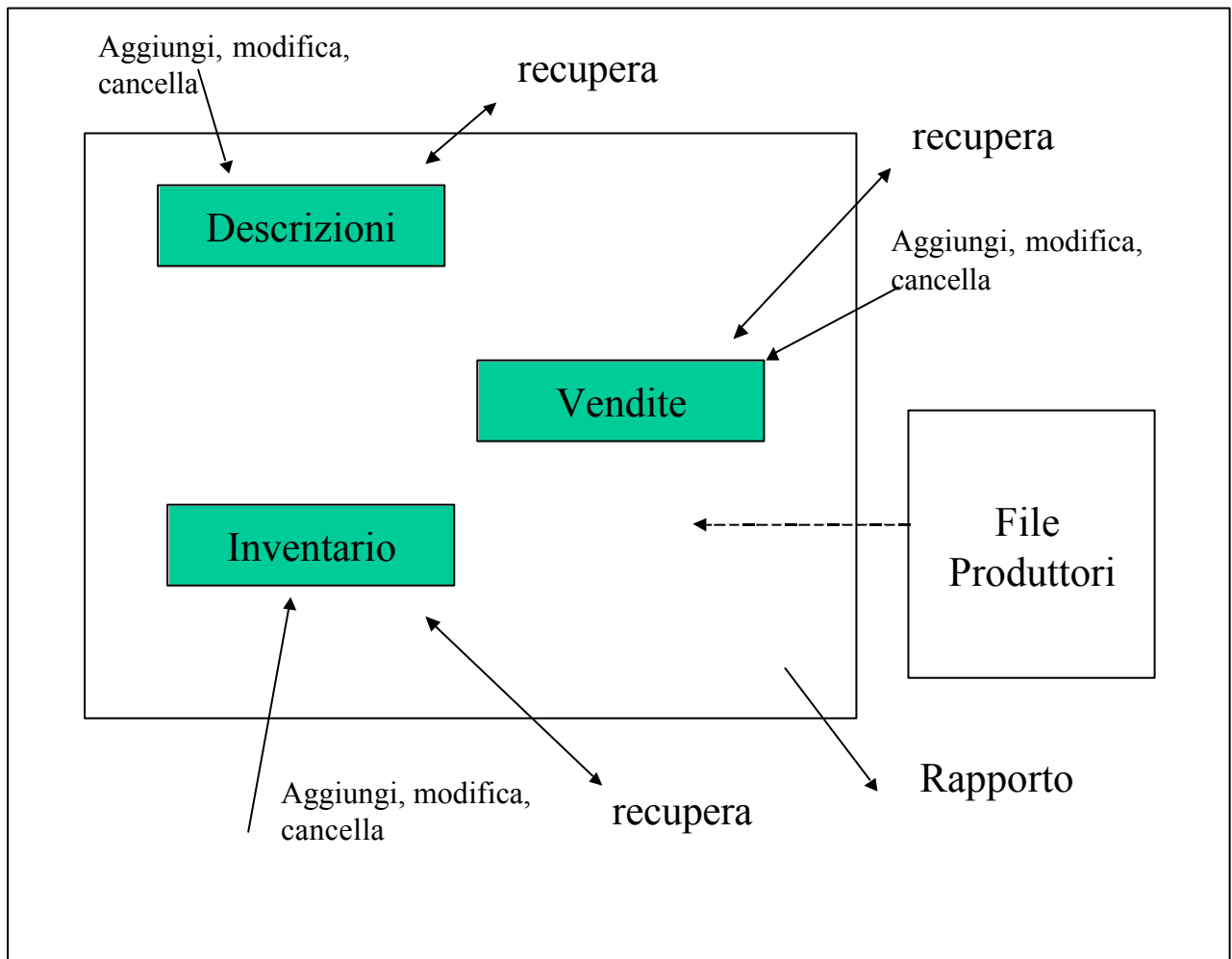
L'applicazione dispone inoltre di una tabella inventario, con una chiave e oltre 20 campi per ciascuna voce, per cui è possibile inserire, modificare, cancellare voci, e recuperare a video le voci.

Un'ulteriore tabella Vendite, con una chiave e 20-50 campi, consente di aggiungere, modificare, cancellare vendite, e recuperare a video le voci.

A fine mese viene generato un rapporto calcolato partendo dai dati delle tre tabelle, ed appoggiandosi ai dati contenuti in un file esterno all'applicazione, che contiene gli indirizzi dei produttori degli articoli commercializzati, e che serve per completezza dei dati di rapporto.

Disegnare lo schema dell'applicazione ed eseguire il conteggio.

Esecuzione conteggio – schema



Esecuzione conteggio

Descrizione	Tipo	DET	RET/FTR	Compless.
Descrizioni	ILF	30	1	B
Descr: agg.	EI	>15	1	M
Descr: mod	EI	>15	1	M
Descr: del	EI	< 5 (confine!)	1	B
Descr: rec	EQ	> 19	1	M
Inventario	EIF	> 15	1	B
Inv: agg.	EI	>15	1	M
Inv: mod	EI	>15	1	M
Inv: del	EI	< 5 (confine!)	1	B
Inv: rec	EQ	> 19	1	M
Vendite	EIF	> 15	1	B
Vend: agg.	EI	>15	1	M
Vend: mod	EI	>15	1	M
Vend: del	EI	< 5 (confine!)	1	B

Vend: rec	EQ	> 19	1	M
Rapporto	EO	> 19	> 1	A
Produttori	EIF			B

Conteggio UFP:

Componente	Basso	Medio	Alto
EI	3 X 3	6 X 4	X 6
EO	X 4	X 5	1 X 7
EQ	X 3	3 X 4	X 6
ILF	3 X 7	X 10	X 15
EIF	1 X 5	X 7	X 10

Totale UFP = **78**

GSC considerate: Data Comm = 4, Distribuzione dell'elaborazione = 0, Prestazioni = 3, Utilizzo intensivo configurazione = 2, Transaction Rate = 3, Online Data Entry = 5, End User Efficiency = 4, Online Update = 3, Complex Processing = 1, Reusability = 0, Installation Ease = 0, Operational Ease = 0, Multiple Sites = 1, Facilitate Change = 2. **TDI = 31**

VAF = (TDI x 0.01) + 0.65 = 0.96

FP APPL INIZ = UFP x VAF = 74.88 = 75

Caso di Studio – Gestione Contatti

L'applicazione di gestione contatti consente di mantenere informazioni su aziende interessate a corsi. I dati sono mantenuti nel gruppo DatiContatto con i seguenti campi:

AZIENDA
NOME CONTATTO
TITOLO
DATA CONTATTO INIZIALE
INDIRIZZO
CITTA
STATO
CAP
TELEFONO
FAX

E' possibile inserire i dati a schermo per creare, cambiare e modificare le informazioni. Tutti i campi indicati sono usati in queste funzioni. La cancellazione richiede solo AZIENDA e NOME CONTATTO.

I seguenti campi sono in DatiContatto ma aggiornati da transazioni separate:

DATA INVIO PACCHETTO
DATA CONTATTO TELEFONICO
NOTE

Ulteriori transazioni: 1) all'invio per posta del pacchetto, in una schermata si inseriscono azienda, nome contatto, data invio, usando un tasto funzione; 2) in due settimane si fa una chiamata telefonica, e si usa una schermata per inserire azienda, nome contatto, data contatto telefonico e note, usando un tasto funzione. La data del contatto viene usata come chiave secondaria per gestire più contatti telefonici.

Il menu dell'applicazione offre le funzioni:

Crea contatto, Preleva Contatto, Aggiorna Contatto, Cancella Contatto, Pacchetto Inviato, Contatto Telefonico Eseguito.

La funzione Preleva Contatto chiede AZIENDA; NOME CONTATTO e usa un tasto funzione, e visualizza tutti i campi di DatiContatto.

Tutte le transazioni fin qui elencate ritornano messaggi di errore definiti in un file mantenuto esternamente, e dotato di 4 campi.

L'applicazione è dotata di Help fornito a livello di schermata e di campi. Sono disponibili da ogni schermata, e il testo dell'help è prelevato da un file di help mantenuto esternamente, e strutturato in meno di 5 campi.

E' inoltre presente un rapporto settimanale di Contatti Scaduti. Il rapporto preleva i dati dal DatiContatto, senza fare cambiamenti su questi dati. Il rapporto è così composto:

Data, Numero pagina; Azienda; Nome Contatto; Data Contatto Iniziale; Pacchetto Inviato; Data Contatto Telefonico Prevista Per (Data calcolata a partire dalla data invio pacchetto); Totali Scaduti (calcolato)

CONTEGGIO

Funzione	Tipo	DET	RET/FTR
DatiContatto	ILF	13	2
Errori	EIF	4	1

Crea Contatto	EI	12	2
Aggiorna Cont	EI	12	2
Cancella Cont	EI	4	2
Pacch Inviato	EI	5	2
Cont Eseguito	EI	6	2
Recupera Cont	EQ	15	2
Help	EIF	< 5	1
Report	EO	6	1
Help Campo	EQ	< 5	1
Help Schermo	EQ	< 5	1

Conteggio GSC:

Data comm 4; Online Data Entry 5; End User Eff 3;
 Online Update 3; Complex Processing 1; Reusability 3;
 Installation Ease 1; Operational Ease 3; Multiple Site
 1; Facilitate Change 2. Total TDI = 26.

Caso di Studio – Project management

Si conti un'applicazione di Project management che consente di definire gli Insiemi Competenze, indicare il Personale che detiene le competenze, inserire le attività da svolgere, assegnare personale alle attività.

Funzioni Transazionali

1. Gli utenti si loggano nel sistema controllando la password in un file mantenuto esternamente
2. Help a livello di campo disponibile per ogni campo di ogni schermata, da un file di Help esterno
3. Per tutte le transazioni interattive a schermo, ci sono messaggi di errore e di conferma; i messaggi sono scritti nel codice e non mantenuti dall'utente
4. Le transazioni a video avvengono usando tasti funzione
5. Un file InsiemiCompetenze viene mantenuto con funzioni di Aggiunta, Modifica, Visualizza. Non ci sono funzioni di cancellazione. Tutti i campi di InsiemiCompetenze sono disponibili sullo schermo per l'aggiunta, la modifica e la visualizzazione
6. I coordinatori di attività inseriscono a schermo le attività. Tutti i campi nel file AttivitàDaEeguire devono essere compilati, ed i campi appropriati devono essere validati facendo riferimento al file Località e al file InsiemiCompetenze. Ogni attività dispone di un ID univoco.

7. Una lista a discesa mostra le priorità delle attività (urgente, importante, media, bassa), a partire da una lista scritta hard-code nel programma.
8. E' disponibile una visualizzazione che contiene tutte le informazioni del file AttivitàDaEeguire
9. Le attività da eseguire possono essere modificate o cancellate se non sono state assegnate. Il file Assegnazioni deve essere referenziato per capire se un'attività è assegnata o meno. Per modificare o cambiare i campi, occorre una validazione sul file Località e InsiemiCompetenze.
10. Le assegnazioni avvengono su personale disponibile con le adeguate competenze, sulla base della priorità delle attività. Il file Assegnazioni viene creato validando sul file Personale (un file mantenuto esternamente) e sul file AttivitàDaEeguire.
11. Gli addetti alle assegnazione possono recuperare e visualizzare le attività non assegnati con tutti i campi del file AttivitàDaEeguire. La visualizzazione può essere ordinata per priorità, ID di competenza, ID di località, e/o data di partenza richiesta. Si può anche stampare una lista con questi campi, che comprende un totale dei task raggruppati per priorità.
12. Gli addetti alle assegnazioni cercano persone con determinati insiemi di competenze, una data località, e una adeguata data di disponibilità (la successiva data di assegnazione disponibile). La schermata di risposta visualizza il nome della persona, gli insiemi di competenze, la località dell'ufficio, ed la successiva data di assegnazione disponibile.
13. E' possibile cancellare un'assegnazione se il lavoro non è cominciato. I dati di assegnazione non possono essere

cancellati o modificati.

14. Gli impiegati hanno accesso al sistema per inserire le date di completamento dell'attività insieme al proprio nome e all'ID dell'attività. Viene validato ed aggiornato solo il file Assegnazioni.
15. Ogni giorno viene generato separatamente un Report che comprende tutte le attività attualmente assegnate (non segnate come complete). Comprende tutti i campi di AttivitàDaEeguire, oltre che il nome persona, la data a cui l'assegnazione deve cominciare, la data attesa di completamento, prelevate dal file Assegnazioni.
16. Viene generato internamente un messaggio di e-mail al supervisore dell'ufficio e ai coordinatori attività avvisando che un dato ID Attività è stato assegnato a un nome persona, con località dell'attività, e la data di inizio assegnazione. Gli indirizzi e-mail sono prelevati dal file Località.
17. Viene inviata una e-mail alla persona a cui è assegnata l'attività, con il nome della persona, email, ID attività, durata dell'attività in giorni, ID località attività, e tutte le informazioni di località, gli ID degli insiemi di competenze necessarie, e le descrizioni degli insiemi di competenze.

Funzioni Dati

ATTIVITÀDAESEGUIRE

ATTIVITAID (PK)
ATTIVITAPRIORITA
ID LOCAZIONE ATTIVITA
ID INSIEMI COMPETENZE NECESSARI (FINO A 2, NON IN SEQUENZA)
DATA INIZIO RICHIESTA
DURATA ATTIVITA IN GIORNI

ASSEGNAZIONI

NOME PERSONA (PK)

ID ATTIVITA (PK)
DATA A CUI ASSEGNAZIONE DEVE COMINCIARE
DATA A CUI ASSEGNAZIONE SI PREVEDE COMPLETATA
(CALCOLATA E MANUTENUTA INTERNAMENTE)
SUCCESSIVA DATA ASSEGNAZIONE DISPONIBILE ATTESA
(CALCOLATA E MANUTENUTA INTERNAMENTE)
DATA COMPLETAMENTO ASSEGNAZIONE (INSERITO DA ADDETTO)

INSIEMI COMPETENZE

ID INSIEME COMPETENZA (PK)
DESCRIZIONE INSIEME COMPETENZA
REQUISITI CERTIFICAZIONI
REQUISITI EDUCATIVI
REQUISITI ADDESTRAMENTO LOCALE
ID INSIEME COMPETENZA AGGIUNTIVI CONSIGLIATI (FINO A 3)

PERSONALE (ESTERNO)

NOME PERSONA (PK)
ID INSIEMI COMPETENZA (FINO A 5 COMPETENZE)
LOCALITA' UFFICIO
INDIRIZZO EMAIL

FILE HELP (ESTERNO)

ID SCHERMATA (PK)
ID CAMPO (PK)
TESTO AIUTO (FINO A 6 RIGHE)

FILE SICUREZZA (ESTERNO)

ID UTENTE LOGIN (PK)
PASSWORD
AUTORIZZAZIONI PROGRAMMA

FILE LOCALITA' (ESTERNO)

ID LOCALITÀ (PK)
INDIRIZZO (3 LINEE)
CITTA
STATO
CAP
TELEFONO UFFICIO
NOME SUPERVISORE UFFICIO
EMAIL SUPERVISORE UFFICIO
NOME COORDINATORE ATTIVITA
EMAIL COORDINATORE ATTIVITA

Conteggio

Funzione	Tipo	DET	RET/FTR
Località	EIF	10 ¹	1 ²
Aiuto	EIF	3 ³	1 ⁴
Sicurezza	EIF	3	1
AttivitàDaEeguire	ILF	6	1
Assegnazioni	ILF	6	1
InsiemiCompetenze	ILF	5	1
Personale	EIF	4	1
1. Login Utente	EQ ⁵	4 ⁶	1
2. Aiuto su Campo	EQ	5 ⁷	1
5. InsiemiCompetenze: Aggiungi	EI	8 ⁸	1

¹ I campi ripetuti si contano come un unico DET

² Si conta come 1 RET il sottogruppo indirizzo, pertanto non si conta come 1 RET Località nel suo intero

³ vedi n. 1

⁴ vedi n. 2

⁵ Viene solo controllato il match utente-password nell'EIF Sicurezza, e ritornata una abilitazione. Nessun ILF viene mantenuto.

⁶ A causa delle specifiche 3. e 4. (messaggi di errore e avviso, tasti per funzioni a schermo), si contano 2 DET (username e password) + 1 DET per messaggi (anche se sono di due tipi si conta 1 solo DET), + 1 DET per tasti funzione (anche se piu' tasti attivano la stessa funzione). Si applica perché il login viene considerata funzione interattiva a schermo.

⁷ Escono dal confine ID schermo (+1 DET) e ID campo (+1 DET) per richiedere il testo, rientrano l'ID schermo (già contato), l'ID campo (già contato) e tre linee di testo, che però si contano come 1 DET, + 2 DET perché è una funzione a schermo con messaggi e tasto funzione.

⁸ 6 DET + 2. Il gruppo di competenze aggiuntive è ripetuto e si conta come unico DET.

5. InsiemiCompetenze: Modifica	EI	8	1
5. InsiemiCompetenze: Visualizza	EQ	8	1
6. Inserisci Attività	EI	8	3 ⁹
7. Lista Discesa Priorità	-	-	_10
8. Visualizza AttivitaDaEeguire	EQ	8	1
9. Modifica AttivitaDaEeguire	EI	8	4 ¹¹
9. Cancella AttivitaDaEeguire	EI	3 ¹²	2 ¹³
10. Crea Assegnazione	EI	8	3 ¹⁴
11. Visualizza Non Assegnate	EQ	8	2 ¹⁵

⁹ Sono coinvolti ATTIVITADAEGUIRE, e anche INSIEMICOMPETENZE e LOCALITA per validare i rispettivi ID.

¹⁰ E' una lista scritta fissa nel codice del programma, le liste hard-coded non si contano

¹¹ Sono coinvolti ATTIVITADAEGUIRE perché direttamente modificato, ASSEGNAZIONI per verificare se un'attività è già assegnata, LOCALITA e INSIEMICOMPETENZE per validare i rispettivi ID presenti in ATTIVITADAEGUIRE

¹² Per la cancellazione basta individuare il record da cancellare, pertanto nell'input è sufficiente far entrare nel confine applicativo la PK. 1 DET + 2 DET di funzione a schermo.

¹³ Impatta su ATTIVITADAEGUIRE e su ASSEGNAZIONI per non cancellare una attività assegnata.

¹⁴ Si appoggia su ASSEGNAZIONI, PERSONALE e ATTIVITADAEGUIRE

¹⁵ ASSEGNAZIONI e ATTIVITADAEGUIRE

11. Stampa lista non assegnate	EO ¹⁶	9 ¹⁷	2 ¹⁸
12. Cerca Persona	EQ ¹⁹	6 ²⁰	2 ²¹
13. Cancella Assegnazione	EI	4 ²²	1
14. Inserim Data Completamento	EI	5 ²³	1
15. Report Attività non Assegnate	EQ ²⁴	9 ²⁵	2 ²⁶
16. Email a supervisor	EQ	8 ²⁷	3 ²⁸
17. Email a impiegato	EQ	16 ²⁹	5 ³⁰

¹⁶ Il calcolo dei totali per raggruppamento costituisce funzione matematica tipica di EO

¹⁷ 6 DET di ATTIVITADAESeguire + 1 TOTALE (campo derivato che attraversa il confine per andare in stampa) + 2

¹⁸ v. n. 15

¹⁹ Non vengono modificati ILF

²⁰ La ricerca avviene facendo entrare nel confine: ID insieme competenze (+1 det), località (+1 det), prima data disponibile (+1 det). La risposta contiene questi tre elementi (già contati) e il nome della persona (+1 det) + 2 det per la funzione a schermo.

²¹ Utilizza ASSEGNAZIONI per verificare la disponibilità e PERSONALE per cercare secondo competenze

²² Per la cancellazione entrano nel confine solo le 2 PK (+ 2 DET), + 2 DET per funzione a schermo

²³ Nome, ID attività e data di completamente (3 det) + 2 det di schermo

²⁴ Non sono presenti campi calcolati o dati derivati, avviene per semplice Join di tabelle

²⁵ tutti i DET di ATTIVITADAESeguire, (6 det) + i 3 det aggiuntivi richiesti. Non si aggiungono i 2 DET perché è un rapporto quotidiano automatico e non dipende da interattività a schermo.

²⁶ ATTIVITADAESeguire e ASSEGNAZIONI

²⁷ Nome, ID Attività, Località, Data, 2 Nomi destinatari email, 2 email = 8 DET. Non è funzione a schermo.

²⁸ LOCALITA, ASSEGNAZIONI, ATTIVITADAESeguire

²⁹ Si contano i 7 DET indicati, più i 10 DET di LOCALITA, -1 DET di LocationID già contato. Non è funzione a schermo.

³⁰ I campi indicati provengono da ATTIVITADAESeguire, INSIEMICOMPETENZE, LOCALITA, ASSEGNAZIONI, PERSONALE (PER EMAIL DESTINATARIO)

assegnato			
-----------	--	--	--

Conteggio a partire da GUI

Consideriamo come esempio di conteggio GUI l'applicazione "Calendario" presente in Windows 3.1.

1. Tipo di Conteggio: Applicazione Esistente
2. Ambito di conteggio (Scope) e Confine Applicativo: si contano solo le funzionalità fornite direttamente dall'applicazione Calendario
3. Conteggio delle funzioni Dati e Transazionali

Calendar consente di mantenere gli appuntamenti giorno per giorno ed ora per ora, eventualmente impostando degli allarmi.

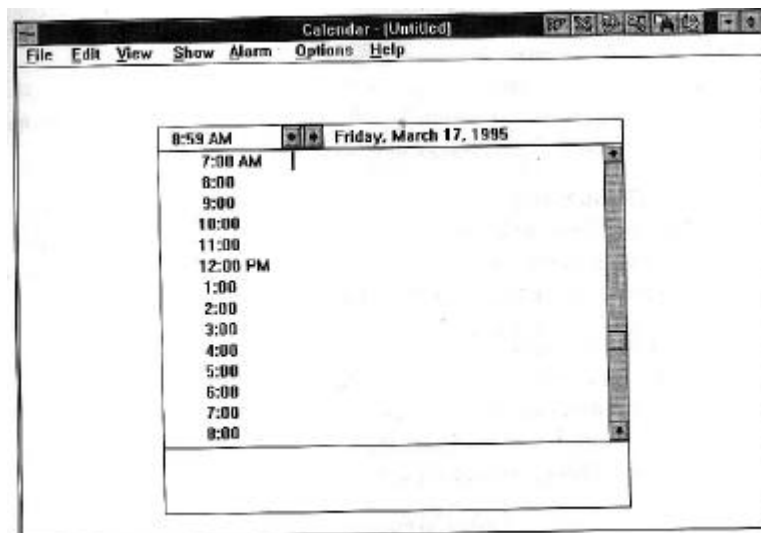


Figure 13-2 Windows Calendar primary window

- Prelievo di ora e data corrente: EQ
Calendar visualizza ora e data corrente, prelevata dallo stato del sistema. Contiene 2 DET (data e ora) e 1 FTR (lo stato di clock del sistema).
- Orologio di Macchina: EIF
L'orologio di macchina viene considerato un file di stato mantenuto esternamente, con due DET sensibili per l'applicazione contata (data e ora) e un 1 RET.

Menu File

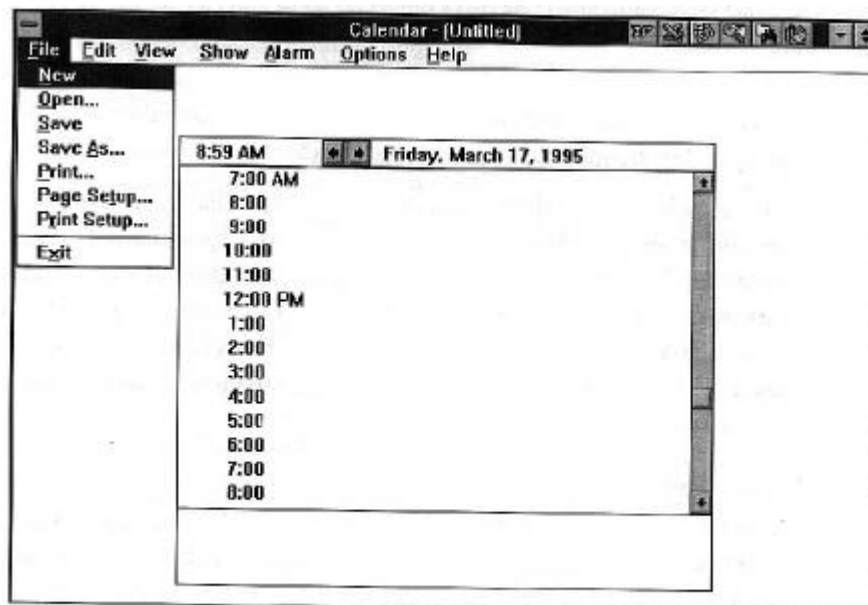


Figure 13-3 Action bar selection: File

- Funzione New: apre un calendario vuoto senza data e con schermo bianco. Non fa contare Function Point
- Funzione Open: apre un calendario per la data richiesta, contenente data richiesta, ora appuntamento, informazione

appuntamento, note ed indicatori allarme, più 1 DET per l'attivazione da comando. EQ con 6 DET e 1 FTR.

- Inserimento Dati Calendario: EI
Si può inserire una nuova data calendario, ora appuntamento, informazioni, note. +1 DET per comando di inserimento. EI con 5 DET e 1 FTR
- Aggiornamento Dati Calendario: EI
Aggiornamento dei dati calendario con data, ora appuntamento, informazioni, note. +1 det per comando. EI con 5 DET e 1 FTR
- Calendario: ILF
Le informazioni sono salvate in un file calendario mantenuto dall'applicazione. Ci sono meno di 20 DET ed 1 RET.

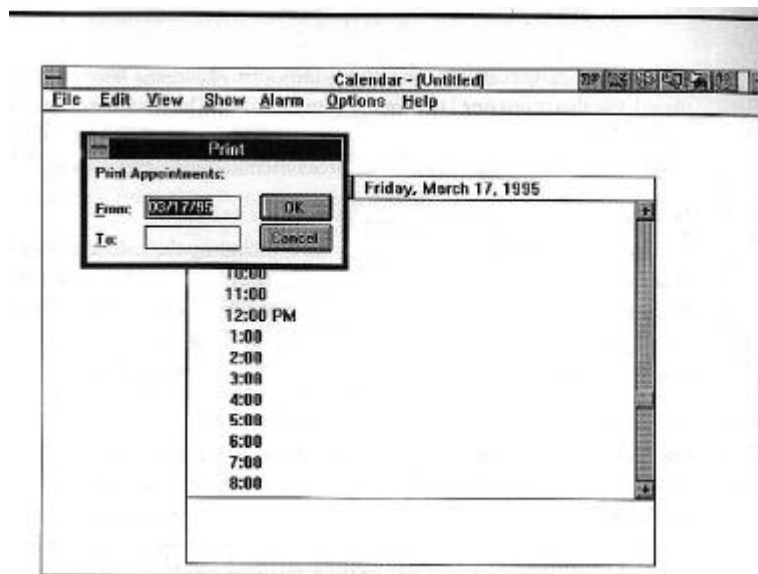


Figure 13-4 Print dialog box

- Funzione Print

La funzione di stampa consente di scegliere da quale data a quale data, e per questo costituisce un EQ a parte rispetto alla visualizzazione, e non tanto perché esce su un supporto diverso. Occorrono i 6 DET della visualizzazione più 2 DET per il da data e alla data. Si possono poi aggiungere i 4 DET del Page Setup e i 4 DET del Print Setup (segue)

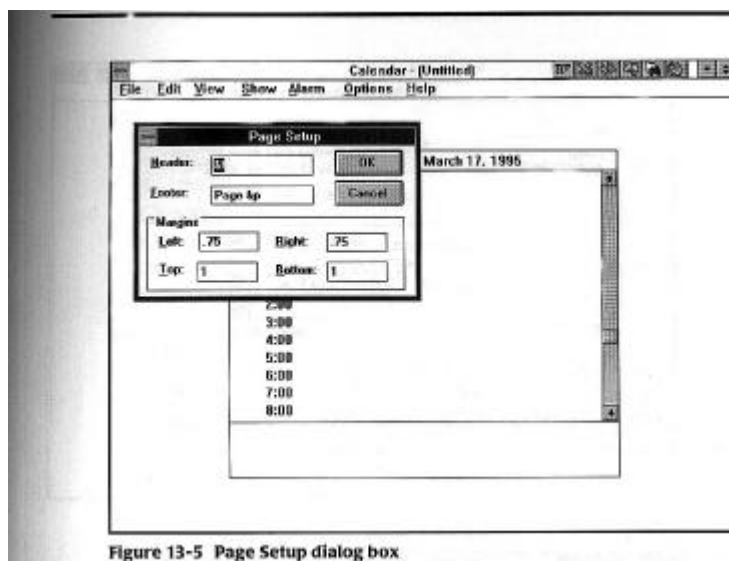


Figure 13-5 Page Setup dialog box

- Funzione Setup Pagina

Contiene impostazione per testata, piede, e 4 ripetizioni della coppia posizione margine e valore (left, right, top e bottom), dunque si contano 4 DET.

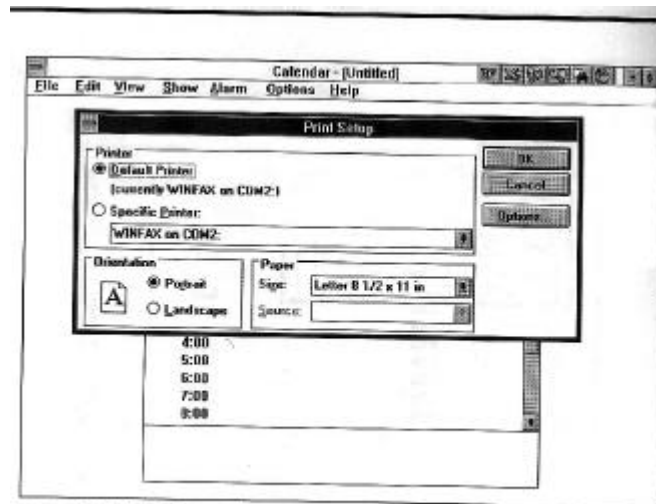


Figure 13-6 Print Setup dialog box

- Funzione di Print Setup

Richiede 4 Det per la scelta stampante specifica/default, formato carta, sorgente carta, orientamento.

- Exit: E' una funzione navigazionale e non si conta.

Menu Edit

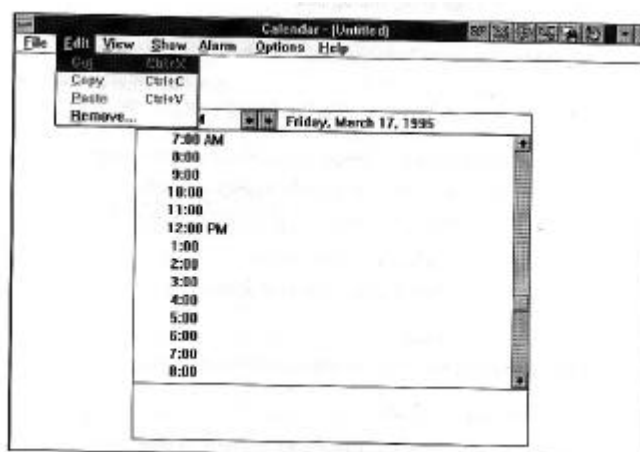
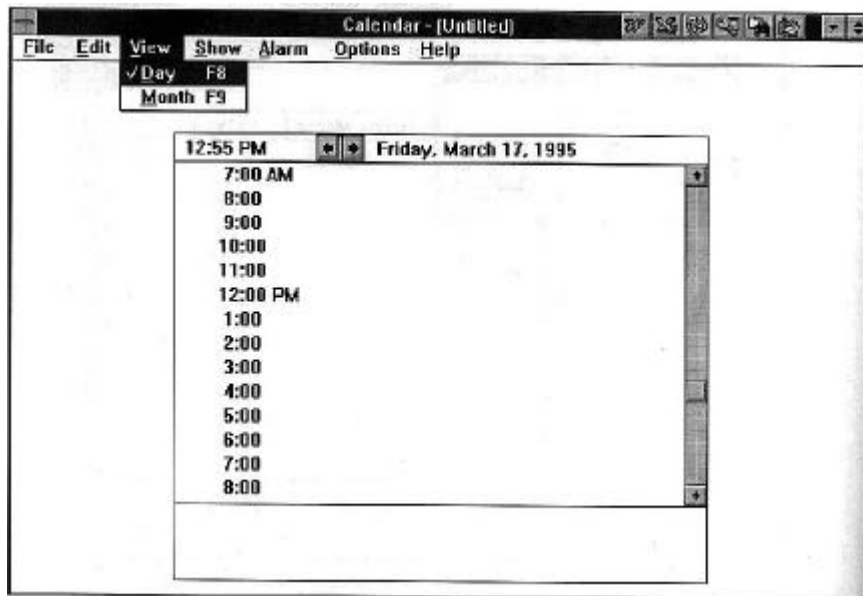


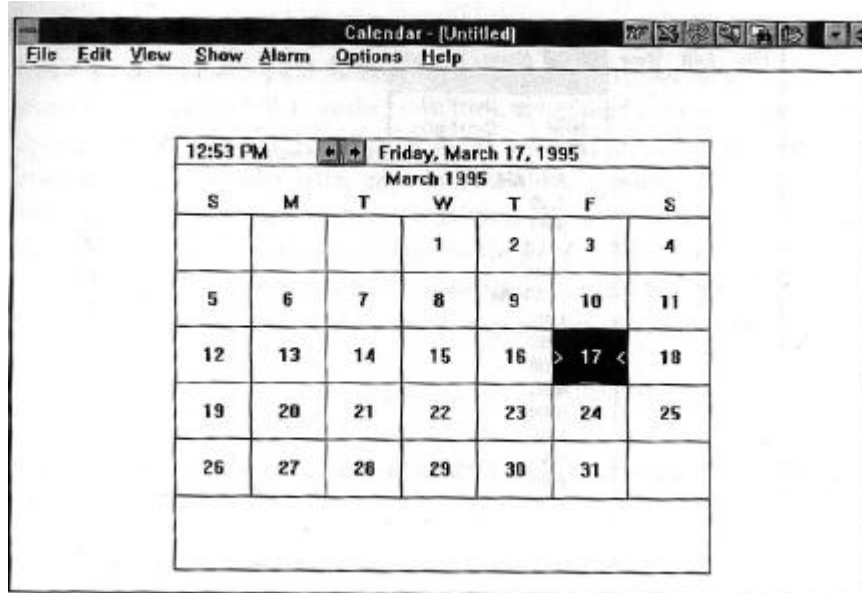
Figure 13-7 Action bar selection: Edit

- Cut Copy e Paste sono funzionalità standard del sistema e non vengono contate.
- Remove permette la cancellazione di un calendario: EI con possibilità di scegliere From date e To Date. Con il tasto attivazione fanno 3 DET e 1 FTR.

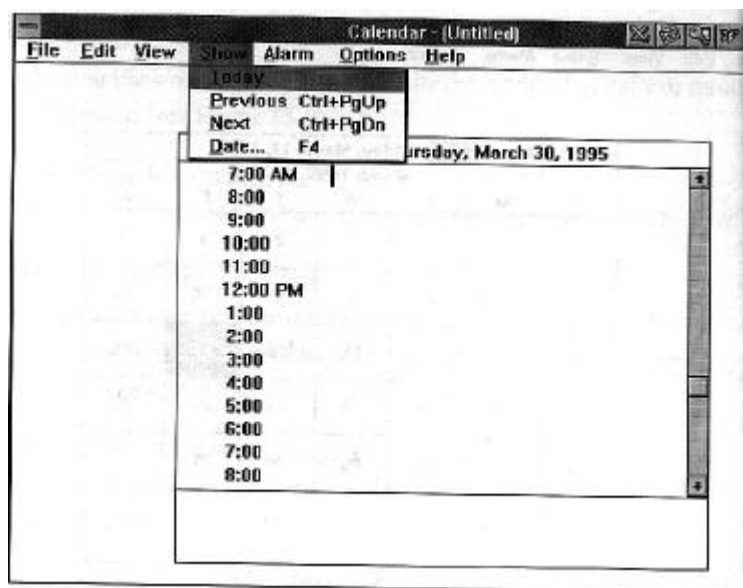


Menu View

- Scelta di visualizzazione giornaliera: già contata nella funzionalità Open
- Scelta di visualizzazione per mese: EQ, visualizza mese ed anno, giorni (ripetuti), e simboli assegnati al giorno. 3 DET + 1 per comando attivazione, 1 FTR.



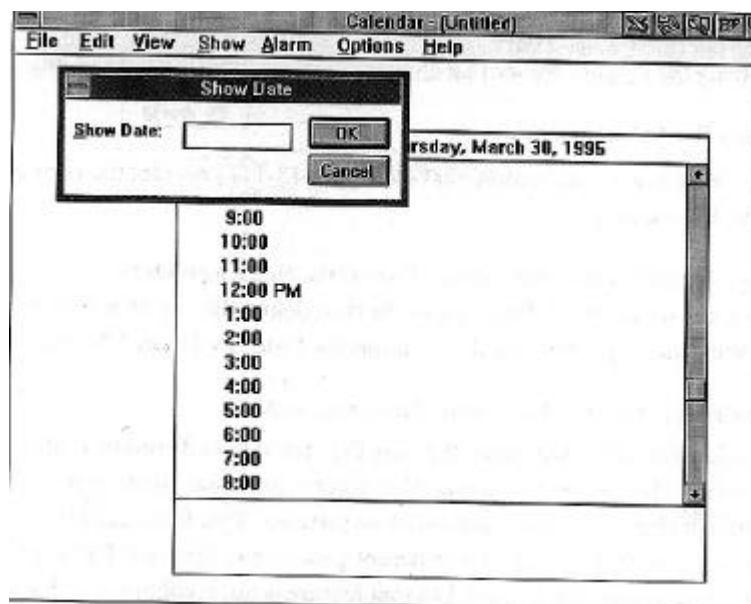
Menu Show



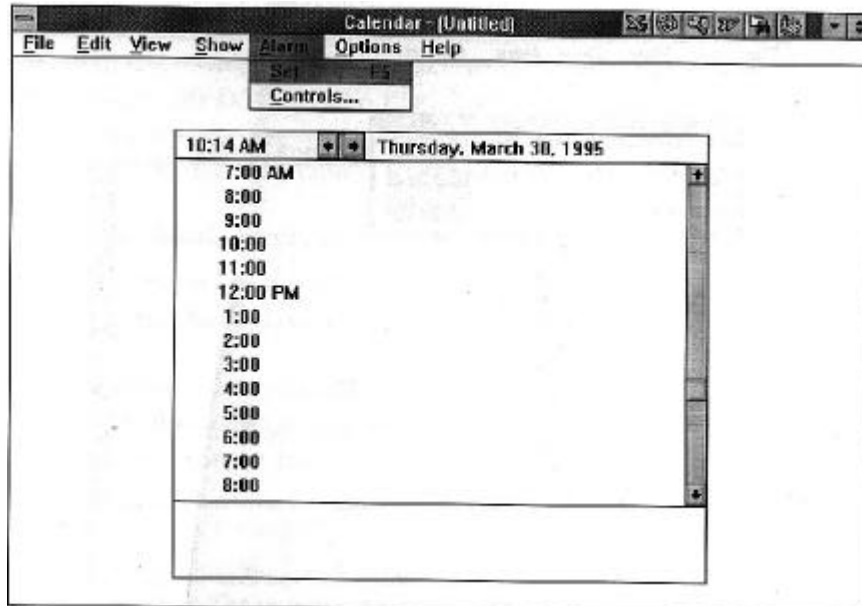
- Show Today, Previous, Next, implementano un EQ già contato per la navigazione lungo il giorno e il

mezzo. Questa funzionalità non ha dunque una logica separata, o nuovi DET o FTR. Fornisce un sistema aggiuntivo di navigazione, ma questo non viene contato nella logica FPA.

- Show Date, è un nuovo EQ perché richiede una data specifica a cui saltare mediante una finestra di dialogo. Risponde con la paginata del giorno, con data e ora, ora appuntamento e descrizione appuntamento (ripetuti), note, e allarme. Contando 1 DET di attivazione, sono 6 DET (la data restituita non si conta perché già entra nell'EQ per la richiesta), con 1 FTR.



Allarme



- E' possibile Impostare (Set) e togliere l'allarme attraverso un segno di spunta (toggle) sulla voce di menu. Si tratta di un EI unico, quindi, e non di due separate funzioni. I Det sono Data, Ora, Toggle (on/off) e 1 di comando di attivazione, per un totale di 4 DET e 1 FTR.

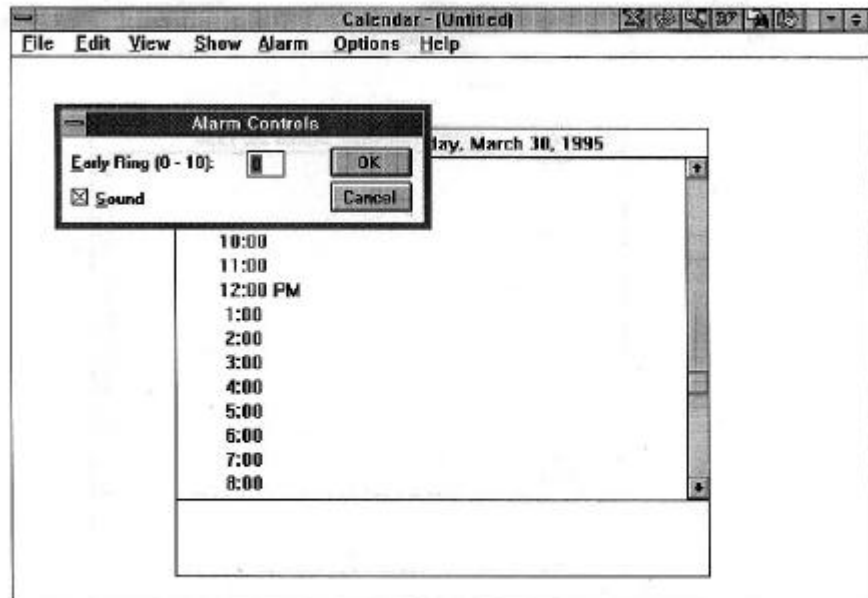


Figure 13-14 Alarm Controls dialog box

- Impostazione di Allarme a livello di applicazione (Alarm Controls): EI. Imposta per tutti gli allarmi l'anticipo di innesco, la modalità silenziosa (toggle), e il tasto di attivazione. Dunque 3 DET e 1 FTR.
- ILF Controlli: le impostazioni di controllo non sono salvate nell'ILF del calendario perché sono indipendenti dalla particolare data, dunque vanno un ILF separato a livello di applicazione.
- Attivazione dell'allarme: EO. Il sistema calcola a partire dall'area corrente e dall'anticipo impostato di sistema quando è ora di far suonare l'allarme ed avvisare l'utente. Utilizza l'ora (1 DET), nessun esplicito tasto di attivazione, e 2 FTR (l'ILF di

calendario per cercare gli allarmi impostati e l'ILF di impostazione Controllo).

- Conferma e Chiusura Allarme: EI. L'utente conferma di aver visto l'allarme e lo disattiva. 1 DET (conferma) e 0 FTR (non viene coinvolto nessun file)

Options

Il menu Opzioni comprende Mark per assegnare un simbolo ad una data, Special Time per assegnare una particolare fascia oraria nel sistema, Day Settings per impostare il formato orario 12/24 e cambiare l'intervallo a cui sono distanziati gli appuntamenti.

- Mark

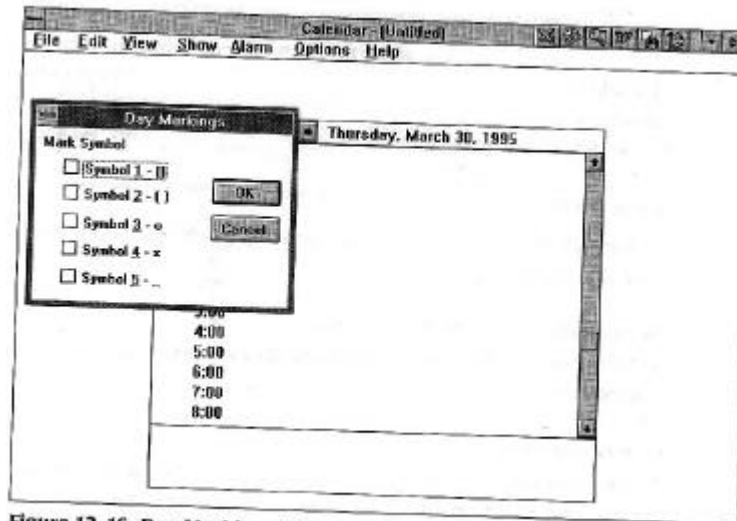


Figure 13-16 Day Markings dialog box

Si può contrassegnare una specifica data con fino a 5 simboli contemporaneamente. Si tratta di un EI con 4 DET (data, numero simbolo, selezione simbolo, tasto comando), 1 FTR.

- Inserimento Special Time: EI

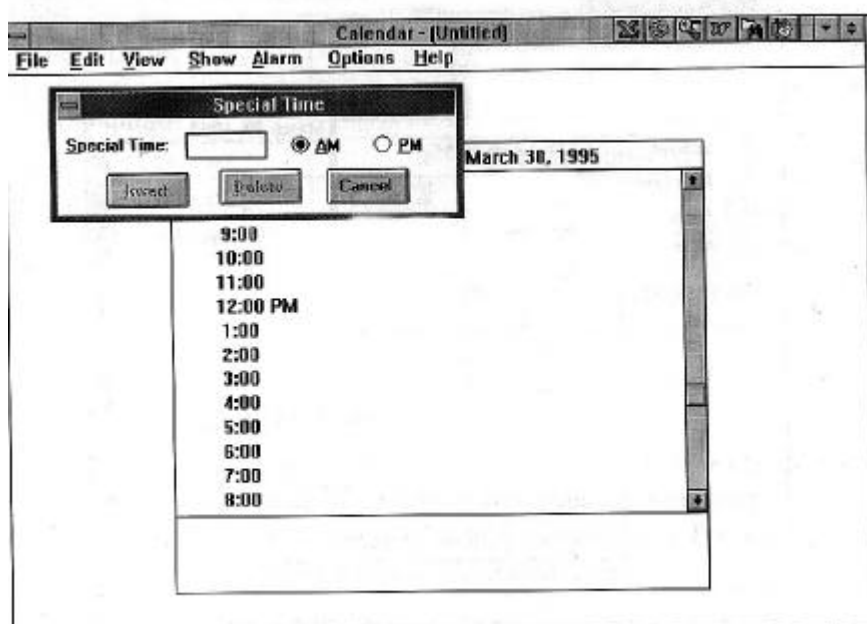


Figure 13-17 Special Time dialog box

Consente di inserire l'ora particolare con AM/PM: 3 DET (calcolando il tasto di Insert o di Delete) ed 1 FTR.

Day Settings

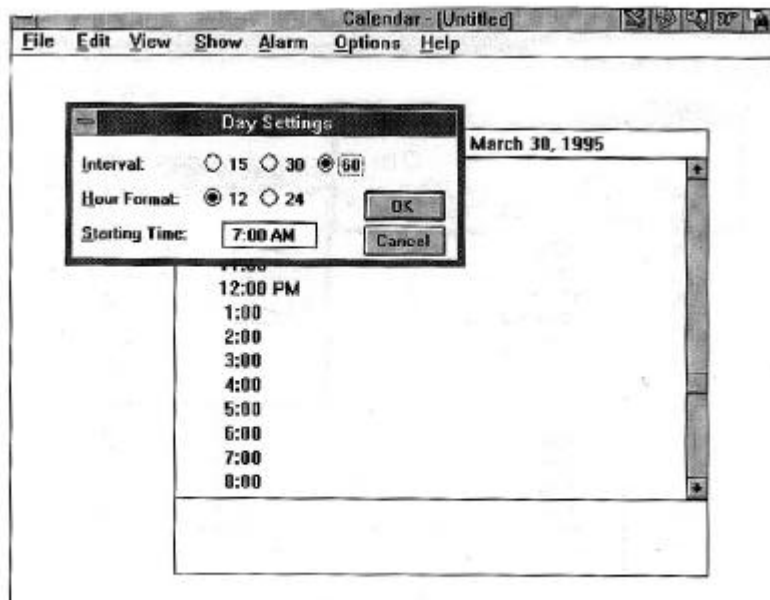


Figure 13-18 Day Settings dialog box

Si tratta di un EI con 4 DET (Intervallo, Formato, Ora di Partenza, Attivazione) ed 1 FTR.

Calcolo delle tabelle

	Function Levels			
	Low	Average	High	Total
External inputs	9*3	*4	*6	27
External outputs	1*4	*5	*7	4
External inquiries	5*3	*4	*6	15
Internal logical files	2*7	*10	*15	14
External interface files	1*5	*7	*10	5
Total unadjusted function points (UFP) = 65				

Calcolo del VAF

Characteristic	Degree of Influence	Characteristic	Degree of Influence
1. Comunicazione Datis	4	8. Online update	3
2. Distribuzione dell'elaborazione	0	9. Complex processing	1
3. Prestazioni	3	10. Reusability	2
4. Utilizzo intensivo configurazione	2	11. Installation ease	1
5. Transaction rate	3	12. Operational ease	3
6. Online data entry	5	13. Multiple sites	2
7. End user efficiency	4	14. Facilitate change	2
Total degree of influence (TDI) = 35			

$$\text{VAF} = (\text{TDI} \times 0.01) + 0.65 = 1.00$$

$$\text{FP} = \text{UFP} \times \text{VAF} = 65 \times 1 = 65 \text{ (Applicazione)}$$

Il conteggio nel paradigma OOP

L'Analisi a FP nasce per cercare di essere il più possibile svincolata dalla tecnica implementativa. Tuttavia la sua impostazione tradisce un retaggio legato ad applicazioni "mainframe", che ragionano in termini di tracciati e di flussi.

Per poter applicare il conteggio FP a paradigmi più complessi quali OOP, occorre una "mappatura" convenzionale tra i concetti FP ed i concetti OOP. Tuttavia, allo stato della specifica (4.1.1) non esiste una mappatura ufficiale. Tutto si riconduce comunque a tentare di prescindere il più possibile dall'implementazione, per mantenere un punto di vista logico a livello utente.

Il punto di vista base per il conteggio, oltre alle specifiche, è il class diagram UML.

FP	OOP
ILF	Classe logica ³¹ interna all'applicazione. Attributi modificabili
EIF	Classe logica esterna all'applicazione. Attributi/Stato non modificabili
DET	Attributi logici ³² della classe
RET	Attributi utilizzati in modo polimorfico per raccogliere valori di diversi sottotipi; Presenza di ereditarietà (la

³¹ La cui presenza cioè non è data da motivi strettamente implementativi, ma dall'analisi del requisiti

³² Attributi cioè la cui presenza non è data da motivi implementativi

	classe contata è sottoclasse); Presenza di specializzazione unicamente dal punto di vista logico (la classe contata è superclasse)
Processo Elementare	Non è detto che coincida con l'invocazione di un singolo metodo: riferirsi a sequence diagram di alto livello e agli use case.
EI	Metodo invocato che cambia lo stato di una classe interna
EQ	Metodo che presenta all'esterno del boundary stati di una o più classi interne
EO	Metodo che presenta all'esterno stati di una o più classi interne modificandole, o per calcolo, o per alterazione dello stato generale del sistema.

Conteggio di Web Application

Anche nel conteggio FP di una web application, il problema è dato dall'impostazione "legacy" dell'analisi. In più, la realizzazione di una web application è un frutto di una integrazione estremamente complessa di numerose tecnologie in stato di continua evoluzione (HTML, DHTML, JavaScript, XML, Flash, XSLT, J2EE, ASP, PHP, ASP.NET ...), e la ripetibilità dei processi produttivi in questo contesto è piuttosto scarsa. Anche in questo caso si può tuttavia tentare un mappaggio tra concetti Web e concetti FP.

FP	Web
ILF/EIF, DET/RET	Legate alle basi dati su cui si appoggia la web application, oppure ai file di sorgente dati da cui attinge per il funzionamento. Distinguere tra dati che vengono effettivamente modificati dalla web application, e tra dati che sono solo usati come lettura (possibilmente da link esterni - EIF). La scrittura di dati in sessione nelle web appl (ASP/JSP/PHP/.NET) è da considerarsi trattamento con ILF.
EI/EO/EQ	Contate sulle maschere HTML o generate dinamicamente con scripting serverside che compongono l'interfaccia applicativa verso l'utente internet. Un salvataggio da una form dentro il db è una EI. Un invio di dati via email all'utente che ne fa richiesta è una EQ se i dati sono solamente "pescati",

	una EO se sono risultato di un calcolo a partire dai dati interni alla web application.
Confine Applicativo	Dato dal dominio del sito, e dalla particolare funzionalità di web appl che si analizza. Le funzionalità del browser sono da scartare. Motori di ricerca non sviluppati nell'applicazioni non sono da includere.
Punto di vista utente	Nel caso di una web application, considerare l'utenza del particolare servizio. Nel caso di un intero sito, considerare gli utilizzatori della home page.
Processo Elementare	Se occorrono più pagine HTML/ASP/JSP/PHP per inserire dati, attraverso più form, ma logicamente si tratta di un inserimento unico, occorre considerarlo come un unico processo elementare.

Dalla Stima della Dimensione alla stima dello Sforzo: COCOMO II

I FP sono una stima della dimensione del software, ma non dello sforzo e della durata (e quindi del costo) che il software potrà avere. Nella determinazione di questi fattori entrano solitamente in gioco altri parametri che FP non considera, quali la percentuale di software riusato e la facilità con cui riusarlo, la coesione del team di sviluppo, la dimestichezza con la tematica di cui tratta il software o con la tecnologia, la maturità del processo di sviluppo aziendale (si usano standard per la progettazione?), il talento degli analisti e dei programmatori, e via dicendo.

COCOMO II (Constructive Cost Model 2° Edizione), redatto nel 2000, propone un modello guida per la stima di sforzo e durata del progetto, a partire da una misura della dimensione del progetto (ad es. in FP). Il modello è strutturato in modo da dover essere calibrato sugli effettivi parametri vitali e di funzionamento dell'azienda. Esso propone anche una calibrazione "base" iniziale eseguita su una media di circa 200 progetti.

Sono disponibili due formule, PM (per calcolare i mesi uomo), e TDEV, per calcolare la durata in giorni. Entrambe si basano su KSLOC (migliaia di righe di codice stimate nel software).

Formule base di COCOMO II

La programmazione nominale in mese uomo è stimata dalla formula

$$PM = A * (KSLOC ^ E) * \text{product}(EM[i])$$

dove KSLOC è la dimensione del codice in migliaia di righe di codice sorgente,e

$$E = B + 0.01 * \text{sum}(SF[i])$$

A e B sono costanti che derivano dalla calibrazione del modello.

EM[i] sono gli Effort Multipliers che per un progetto con caratteristiche nominali sono tutti ad 1, dando una produttoria 1.0

SF[i] sono gli Scale Factors che per un progetto con caratteristiche nominali danno una sommatoria di 19

Il tempo di sviluppo in mesi, TDEV:

$$TDEV = C * PM ^ F$$

$$\text{Dove } F = D + 0.2 * 0.01 * \text{sum}(SF[i]) = D + 0.2 * (E - B)$$

C e D sono altre due costanti derivate dalla calibrazione del modello

Le costanti COCOMO II.2000 ottenute su 160 campioni di modello sono:

$$A = 2.94, B = 0.91, C = 3.67, \text{ e } D = 0.28$$

Risolviendo TDEV in funzione di KSLOC:

$$TDEV = 3.67 * (2.94 * KSLOC ^ 1.1) ^ .318$$

Invertendo per trovare KSLOC in funzione di TDEV:

$$KSLOC = (.34 * (.272 * TDEV) ^ 3.14) ^ .909$$

Il numero di persone necessarie nominalmente nel team di ottiene con:

$$N = PM / TDEV$$

Applicazione della formula base nel caso di un progetto stimato a 100 KSLOC. Consideriamo i moltiplicatori di sforzo superiori al valore nominale di 1.0 per modellare uno sviluppo di un grosso progetto

$$\mathbf{PM = A * (KSLOC ^ E) * product(EM[i]) = 2.94 (100)^{1.15} = 586.51 \text{ Mesi Uomo}}$$

Deriviamo da questo la durata prevista

$$\mathbf{TDEV = C * PM ^ F = 3.67 (586.6)^{(0.28 + 0.2 * (1.15 - 0.91))} = 3.67 (586.6)^{0.328} = 29.7 \text{ Mesi}}$$

Da cui si evince che il numero di persone medio necessario al progetto:

$$\mathbf{PM / TDEV = 19.75 = 20}$$

Da FP a KSLOC

Il mappaggio da FP a KSLOC avviene attraverso una tabella, ottenuta mediante lo studio comparato delle due misurazioni. La tabella può essere calcolata internamente sullo storico produttivo aziendale, oppure estratta dalla letteratura.

Linguaggio	SLOC/UFP
C	128
Basic	64
C++	55
Java	53
Visual Basic	29
Visual C++	34
Cobol	91
Assembly	320

Configurazione del Modello COCOMO II

In COCOMO II è possibile applicare fattori correttivi per adattare la dimensione in KSLOC sui seguenti parametri:

- SU : Software Understanding
- AA : Assessment and Assimilation (possibilità di usare in modo rapido porzioni riusabili)
- UNFM: Unfamiliarity dei programmatori con la tecnologia
- REVL: Requirements Unstability and Volatility

Alcuni degli Effort Multipliers e degli Scale Factors sono invece:

- PREC (Precedentness), FLEX (Development Flexibility), TEAM (Team Cohesion), PMAT (Process Maturity), RELY (Software Reliability), DATA (Database Size), RUSE (Develop for Reusability), CPLX (Product Complexity), DOCU (Documentation), TIME (Execution Time Constraint), ACAP (Analyst Capability), PCAP (Programmers Capability), PCON (Personnel Continuity), APEX (Application Experience), PLEX (Platform Experience), LTEX (Language and Tool Experience), TOOL (Use of software Tools).

Tema d'esame per la certificazione CFPS

Bibliografia e Link Utili

- D. Garmus, D. Herron – Function Point Analysis, Addison Wesley
- IFPUG – Function Point Counting Practices Manual, version 4.1.1, IFPUG
- Boehm, B. et al. – Software Cost Estimation With COCOMO II, Prentice Hall
- Data Processing Organization, www.dpo.it
- IFPUG web site: www.ifpug.org
- GUFPI web site: www.gufpi.org