

Standard di conteggio function point

Indicazioni Generali

Versione 1.2

Redatto da: S. Coletta, P. De Lazzaro, R. La Piana, C. Manganiello, L. Noci, M. D. Perriello
Approvato da: M. Venzo

Ottobre 2009



TABELLA DELLE VERSIONI

Data	Versione	Descrizione	Cap. /Sez. modificati
Settembre 2001	V1	Nascita del documento	tutti
Settembre 2004	V1.1	adeguamento a IFPUG release 4.1.1	1, 2, 2.1, 2.1.1, 2.2, 2.3, 2.4, 4.1.1, 4.1.3, 4.2.1, 4.2.3, 4.3.1, 4.3.2, 4.3.3, 4.4.1, 4.4.2, 4.4.3, 4.5.1, 4.5.2, 4.5.3, allegato 1, allegato 2
Ottobre 2009	V.1.2	Adeguamento a IFPUG release 4.2	1,2,4,4.1.1,4.1.2,4.1.3, 4.3.4



INDICE

1. INTRODUZIONE	4
1.1. OBIETTIVI	4
2. LA METRICA DEI PUNTI FUNZIONE	5
2.1. L'UTENTE	6
2.1.1. <i>Il punto di vista dell'utente</i>	6
2.2. I CONFINI DELL'APPLICAZIONE	6
2.3. LE FUNZIONI	7
2.4. IL FATTORE DI AGGIUSTAMENTO	8
2.5. CALCOLO FINALE	10
3. INDIVIDUAZIONE DEI CONFINI DELL'APPLICAZIONE	11
3.1. CRITERI PER L'INDIVIDUAZIONE DEI CONFINI DELL'APPLICAZIONE	11
4. INDIVIDUAZIONE E CLASSIFICAZIONE DEGLI ELEMENTI "DATI" E "FUNZIONE"	12
4.1. INTERNAL LOGICAL FILE (ILF)	13
4.1.1. <i>Definizione</i>	13
4.1.2. <i>Criteri per l'individuazione</i>	13
4.1.3. <i>Complessità</i>	14
4.1.4. <i>Esempi</i>	15
4.2. EXTERNAL INTERFACE FILE (EIF)	16
4.2.1. <i>Definizione</i>	16
4.2.2. <i>Criteri per l'individuazione</i>	16
4.2.3. <i>Complessità</i>	17
4.2.4. <i>Esempi</i>	18
4.3. EXTERNAL INPUT (EI)	18
4.3.1. <i>Definizione</i>	18
4.3.2. <i>Criteri per l'individuazione</i>	19
4.3.3. <i>Complessità</i>	20
4.3.4. <i>Esempi</i>	21
4.4. EXTERNAL OUTPUT (EO)	21
4.4.1. <i>Definizione</i>	21
4.4.2. <i>Criteri per l'individuazione</i>	21
4.4.3. <i>Complessità</i>	21
4.4.4. <i>Esempi</i>	21
4.5. EXTERNAL INQUIRY (EQ)	21
4.5.1. <i>Definizione</i>	21
4.5.2. <i>Criteri per l'individuazione</i>	21
4.5.3. <i>Complessità</i>	21
4.5.4. <i>Esempi</i>	21
5. RIEPILOGO PASSI PER IL CONTEGGIO DEI FP	21
6. MISURA DELLA "BASELINE"	21
7. ALLEGATO 1	21
8. ALLEGATO 2	21
9. ALLEGATO 3	21



1. INTRODUZIONE

1.1. OBIETTIVI

Il documento ha l'obiettivo di fornire delle linee guida per l'applicazione della metrica dei Punti Funzione (FP - Function Point) con riferimento alle applicazioni che verranno sviluppate e/o mantenute nell'ambito dei Progetti Informatici gestiti da Consip.

Quanto descritto di seguito è basato sulle norme di conteggio dell'International Function Point User Group (IFPUG), così come esposte nel manuale ufficiale: "Function Point: Manuale sulle Regole del Conteggio", ver. 4.2 (2004) (traduzione italiana a cura del Gruppo Utenti Function Point Italia - GUFPI-ISMA), di cui è richiesta la conoscenza.

Il presente documento costituisce delle linee guida base per l'applicazione della metrica dei Punti Funzione ed è integrato con altri documenti che forniscono indicazioni specifiche per applicazioni di tipo data warehouse, per applicazioni con interfaccia grafica e per siti web.

FP - I.G.		
FP - DWH	FP - GUI	FP - WEB

FP - I.G.	Standard conteggio Function Point - Indicazioni Generali, Vers. 1.2, Consip
FP - DWH	Standard conteggio Function Point - Applicazioni data warehouse, Vers. 2.1, Consip
FP - GUI	Standard conteggio Function Point - Applicazioni con interfaccia grafica GUI, Vers. 1.2, Consip
FP - WEB	Standard conteggio Function Point - Progetti di realizzazione di Siti Web, Vers. 1.1, Consip

Il documento è articolato come segue :

- descrizione generale della metrica dei Punti Funzione;
- criteri di individuazione delle Applicazioni e dei relativi confini;
- criteri di individuazione e classificazione degli Elementi di tipo Dati e Funzione, con la descrizione di alcuni esempi;
- valori del Fattore di Aggiustamento;
- misura della "baseline";
- fasi del Ciclo di Vita del Software (CVS) in cui applicare la metrica e modalità di raccolta delle informazioni;
- allegati.

La decodifica delle sigle e degli acronimi utilizzati nel presente documento è riportata nel glossario generale presente nell'allegato1.

Come supporto nel conteggio dei FP, laddove, si trovi difficoltà nell'applicazione delle regole IFPUG, è possibile utilizzare le Linee Guida Italiane (LGI) fornite dal Comitato per le regole del conteggio (CPC) del Gruppo Utenti Function Point Italia (GUFPI-ISMA).

Le indicazioni contenute nelle LGI non sono in contrasto con le regole standard bensì ne costituiscono un ideale perfezionamento ed approfondimento (per ulteriori informazioni: <http://www.gufpi-isma.org/>).



2. LA METRICA DEI PUNTI FUNZIONE

I Punti Funzione si ripromettono di misurare la “quantità” di contenuto funzionale del software, così come essa è percepita dall’utente. Essi costituiscono perciò una metrica di volume che, oltre a consentire una misura sulla base di elementi visibili e riconosciuti dall’utente ha la caratteristica di poter essere applicata già durante le fasi alte del CVS e di essere indipendente dalla tecnologia utilizzata per lo sviluppo.

La metrica, ideata nel 1979 da Allan J. Albrecht, fu da lui stesso perfezionata nel 1984, in ambito IBM. Nel 1986 si è costituito l’ IFPUG che ha pubblicato delle linee guida e i loro successivi aggiornamenti. Successivamente sono nate varie associazioni, allo scopo di diffondere e assistere gli utenti della metrica, tra cui il GUFPI-ISMA che, tra l’ altro, cura la traduzione in italiano delle linee guida IFPUG.

I Punti Funzione, in quanto metrica “primaria”, possono essere correlati ad altre unità di misura ottenendo delle metriche “derivate”. Così, ad esempio, correlando la dimensione in FP di un’ Applicazione ai giorni/persona necessari per realizzarla, si può ricavare la produttività; il numero di errori per FP, invece, è un indice di difettosità.

La metrica dei Punti Funzione può essere utilizzata :

- per stimare un software di nuovo sviluppo, allo stadio iniziale di un progetto;
- per rivedere ed aggiornare la stima di cui al punto precedente, ad ogni fase del CVS;
- per misurare il volume effettivo al termine della realizzazione, ossia all’ atto della messa in esercizio delle funzioni sviluppate.

E’ possibile, con la stessa tecnica, misurare un software già esistente (calcolo della “baseline” ad una certa data).

Quando un software è soggetto a revisione con conseguente variazione del contenuto funzionale, non è possibile applicare la metrica dei Punti Funzione per un calcolo diretto della differenza, ma occorre calcolare la “baseline” dopo l’ intervento. Tale differenza non sempre è proporzionale all’ impegno di lavoro necessario per produrla.

Visto il successo della metrica dei Punti Funzione, tuttavia, si è pensato di utilizzarla anche per valutare direttamente l’ impegno di un progetto di revisione. In questo caso si utilizza un conteggio diverso da quello usato per i nuovi sviluppi, che fornisce un valore, in Punti Funzione, che non è pari alla differenza tra le due baseline, prima e dopo l’ intervento, in quanto tiene conto, ad esempio, delle sostituzioni di funzionalità con delle altre. A rigore questa non può essere considerata una metrica “primaria” e non va confusa con la precedente, anche se entrambe sono utilizzate, in sede di stima, per proiettare i costi di un progetto.

La metrica dei Punti Funzione misura un’Applicazione operando in due fasi successive:

- la prima, che riflette le funzionalità fornite all’ utente, e fornisce il valore degli Unadjusted Function Point (UFP);
- la seconda che valorizza la complessità delle funzionalità generali fornite all’ utente, attraverso il valore del Fattore di Aggiustamento (VAF - Value Adjustment Factor).



Affinando il conteggio ottenuto dalla prima fase con quello risultante dalla seconda fase, viene determinato il numero finale dei Function Point (FP) di un'Applicazione.

2.1. L'UTENTE

Un *utente* è una qualsiasi persona che specifica dei Requisiti Funzionali Utente e/o qualsiasi persona o cosa che comunica o interagisce con il software in qualsiasi momento.

2.1.1. IL PUNTO DI VISTA DELL'UTENTE

Fondamentale ai fini del conteggio dei FP è il punto di vista dell'utente.

Il *punto di vista dell'utente* è una descrizione formale delle sue necessità applicative espresse nel linguaggio utente. Il gruppo di sviluppo traduce le informazioni utente in un linguaggio tecnico al fine di fornire una soluzione.

Un punto di vista dell'utente:

- è una descrizione delle funzionalità applicative
- è approvato dall'utente
- può essere usato per il conteggio dei FP
- può variare nella sua rappresentazione fisica (ad es. elenco di transazioni, insieme di proposte, documento di requisiti, specifiche funzionali, specifiche di dettaglio, manuale utente)

2.2. I CONFINI DELL'APPLICAZIONE

Un sistema software si considera composto di Applicazioni.

Propedeutica all' utilizzo del metodo è l'individuazione dei confini dell'Applicazione o delle Applicazioni oggetto dell' intervento.

Il confine dell'applicazione indica la linea di separazione fra il software oggetto di misura e l'utente.

Il confine dell'applicazione:

- definisce che cosa è esterno all'applicazione
- è l'interfaccia concettuale tra l'applicazione, che costituisce l'"interno", e il mondo utente, che costituisce l'"esterno"
- agisce come una "membrana" attraverso la quale i dati elaborati dalle transazioni (EI, EO ed EQ) entrano ed escono dall'applicazione
- comprende i dati logici mantenuti dall'applicazione (ILF)
- aiuta a identificare i dati logici referenziati ma non mantenuti dall'applicazione (EIF)
- dipende dal punto di vista applicativo esterno che ha l'utente sull'applicazione. È indipendente da considerazioni tecniche e/o di natura implementativa.

Sulla base di tale linea di separazione vengono individuati i processi proprietari delle informazioni e dei dati che verranno trattati durante il conteggio.

Per una descrizione più dettagliata si rimanda alla sezione 3 del documento.



2.3. LE FUNZIONI

Nel conteggio degli UFP, le funzionalità vengono valorizzate considerando il "cosa" viene consegnato all'utente e non il "come", nel senso che vengono prese in considerazione solo quelle richieste dall'utente ed a questo visibili (non hanno, ad esempio, alcun peso archivi temporanei o aree di passaggio di dati fra moduli software).

Si distinguono Funzioni di tipo Dati e Funzioni di tipo Transazione; nel prosieguo queste verranno rispettivamente denominate Elementi di tipo Dati ed Elementi di tipo Funzione (vedi figura 1).

Gli Elementi di tipo Dati sono:

- Internal Logical File (ILF): aggregato logico di dati o di informazioni di controllo gestiti internamente all'Applicazione;
- External Interface File (EIF): aggregato logico di dati o di informazioni di controllo condivisi con altre Applicazioni.

Gli Elementi di tipo Funzione sono:

- External Input (EI): processo elementare di elaborazione di dati o informazioni di controllo che provengono dall'esterno del confine dell'Applicazione misurata;
- External Output (EO): processo elementare che invia dati o informazioni di controllo all'esterno del confine dell'Applicazione;
- External Inquiry (EQ): processo elementare che invia dati o informazioni di controllo all'esterno del confine dell'Applicazione (la differenza tra l'EQ e l'EO dipende dall'intento primario del processo).

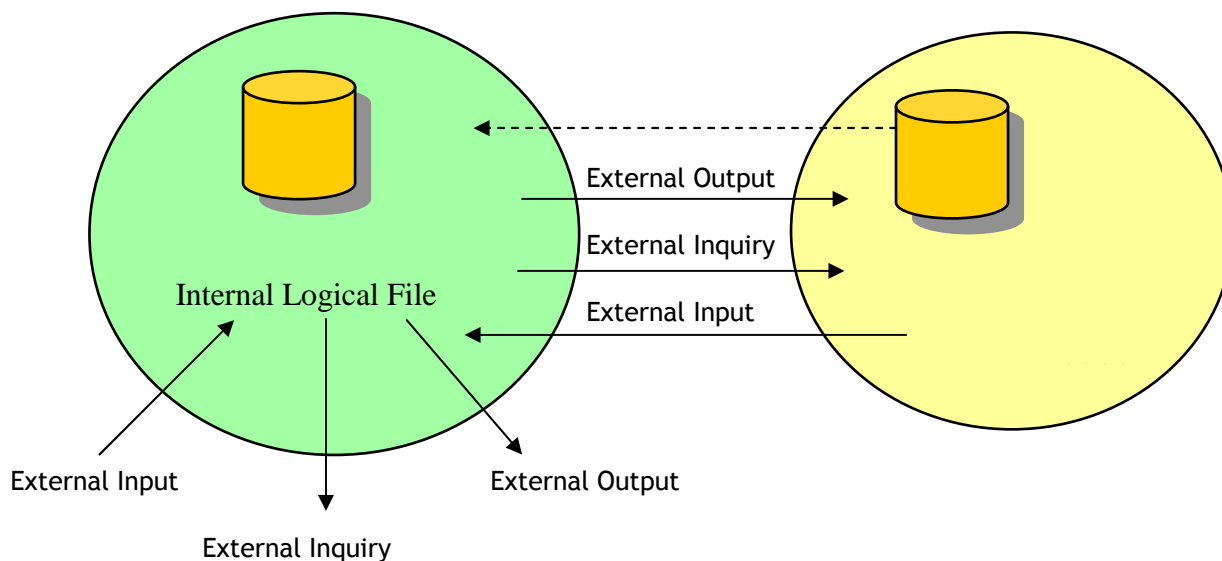
La complessità di ciascun Elemento viene quantificata attraverso l'assegnazione di un peso (Basso, Medio, Alto) e porta alla valorizzazione degli Unadjusted Function Point.

Per una descrizione più dettagliata degli Elementi e della terminologia cui si fa riferimento, si rimanda alla sezione 4 del documento.



APPLICAZIONE

ALTRA APPLICAZIONE



UTENTE ESTERNO

Fig. 1 Elementi del conteggio dei Punti Funzione

2.4. IL FATTORE DI AGGIUSTAMENTO

Alla luce del recente orientamento desunto dalle comunicazioni dell'IFPUG e dai lavori del comitato tecnico ISO preposto alla standardizzazione di un metodo di misurazione funzionale del software, si conviene di assumere un valore del fattore di aggiustamento unico e unitario, ossia di utilizzare come metrica di dimensionamento funzionale dei progetti il numero di UFP (Unadjusted Function Point), ottenuto dalla somma dei contributi non pesati delle funzioni di tipo dati e delle funzioni di tipo transazione (somma calcolata secondo le formule riportate nelle successive sezioni).

In conclusione:

$$\text{VAF} = \text{VAFA} = \text{VAFB} = 1$$

Per completezza si documenta comunque la valutazione del fattore di aggiustamento.

Il Fattore di Aggiustamento (VAF) viene misurato sulla base di 14 sotto-fattori, detti Caratteristiche Generali del Sistema, che vengono utilizzati al fine di valutare la complessità delle funzionalità oggetto di conteggio.

Le 14 caratteristiche sono:



- 1 Comunicazione dati
- 2 Distribuzione dell' elaborazione
- 3 Prestazioni
- 4 Utilizzo intensivo della configurazione
- 5 Frequenza delle transazioni
- 6 Inserimento dati interattivo
- 7 Efficienza per l' utente finale
- 8 Aggiornamento interattivo
- 9 Complessità elaborativa
- 10 Riutilizzabilità
- 11 Facilità di installazione
- 12 Facilità di gestione operativa
- 13 Molteplicità di siti
- 14 Facilità di modifica

Per ognuna di esse viene attribuito un valore, relativamente all' Applicazione in oggetto, utilizzando una scala di sei valori detta anche scala del grado di influenza (DI - Degree of Influence):

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

La somma dei pesi delle caratteristiche (TDI - Total Degree of Influence) determina il valore del Fattore di Aggiustamento (VAF), secondo la seguente formula :

$$VAF = (TDI * 0,01) + 0,65$$

In base a tale formula, i fattori di complessità hanno un impatto sul calcolo iniziale dei Punti Funzione (UFP) che varia fra il -35% ed il +35%.

Per una descrizione più dettagliata della valutazione di tali fattori, si rimanda alla documentazione IFPUG.



2.5. CALCOLO FINALE

Il calcolo varia a seconda che si tratti di un progetto che sviluppi una nuova Applicazione o che effettui una revisione di un' Applicazione preesistente.

Nuovo Sviluppo.

Si applica la seguente formula :

$$AFP = (UFP + CFP) * VAF$$

dove :

UFP è il numero di Unadjusted Function Point;

CFP è il numero di UFP relativi alle funzionalità di conversione (rilasciate all'atto dell'installazione dell' Applicazione); per funzionalità di conversione si intende, ad esempio, quelle funzioni che consentono di migrare una base dati da un data base di un certo tipo ad un altro;

VAF è il valore del Fattore di Aggiustamento

Revisione di un' Applicazione preesistente.

Si applica la seguente formula :

$$AFP = [(ADD + CHGA + CFP) * VAFA] + (DEL * VAFB)$$

dove:

ADD è il numero di Unadjusted Function Point relativi alle funzionalità aggiunte;

CHGA è il numero di UFP relativi alle funzionalità modificate dall'applicazione, valutati al completamente dell'intervento;

CFP

conversione (rilasciate all'atto dell'installazione dell'Applicazione);

DEL è il numero di UFP relativi alle funzionalità cancellate;

VAFA è il valore del Fattore di Aggiustamento dell'Applicazione dopo l'intervento.

VAFB è il valore dei Fattore di Aggiustamento dell'Applicazione prima dell'intervento.



3. INDIVIDUAZIONE DEI CONFINI DELL'APPLICAZIONE

Come già premesso al paragrafo 2.2 per poter effettuare un corretto conteggio in FP di un'Applicazione è determinante averne fissato i confini.

Di seguito si descrivono i criteri e le modalità con cui è possibile determinarli.

3.1. CRITERI PER L'INDIVIDUAZIONE DEI CONFINI DELL'APPLICAZIONE

I criteri che seguono possono essere adottati a partire dalle specifiche che descrivono i dati e le funzionalità del sistema da sviluppare :

- **Processo amministrativo.**
Occorre guardare al sistema dal punto di vista utente, ossia individuare l' iter amministrativo e le funzioni automatiche che lo supportano. E' opportuno che un' Applicazione corrisponda (per quanto possibile) ad un unico processo amministrativo.
- **Struttura organizzativa utente.**
Occorre raggruppare le funzionalità del sistema secondo gli uffici che le utilizzano. E' opportuno che un' Applicazione corrisponda (per quanto possibile) ad un' unica struttura organizzativa utente.
- **Proprietà dei dati.**
E' opportuno analizzare i file logici che sono mantenuti dall' Applicazione e tener presente che quelli in comune a più Applicazioni devono costituire un insieme di piccola entità. E' opportuno che un ILF appartenga (per quanto possibile) ad una sola Applicazione.

Non è necessario che un' Applicazione corrisponda ad un' unica piattaforma tecnologica di riferimento.

Per progetti che effettuino revisioni di Applicazioni preesistenti, il confine iniziale deve essere conforme con quello già fissato per l' Applicazione oggetto di modifica; il confine finale potrà modificarsi in funzione di ciò che viene aggiunto o modificato.



4. INDIVIDUAZIONE E CLASSIFICAZIONE DEGLI ELEMENTI “DATI” E “FUNZIONE”

In questa sezione vengono descritte le modalità con cui identificare e classificare gli Elementi di tipo Dati e quelli di tipo Funzione.

Gli Elementi di tipo Dati vengono classificati in tre tipologie:

1. Dati di Business
2. Dati di Riferimento
3. Dati di Decodifica.

Per una corretta identificazione dei File Logici occorre anche considerare il concetto di “dipendenza - indipendenza” tra due entità e di significatività di queste per l’utente.

In linea di massima, per chiarire, se si considerino due entità A e B che sono legate da una relazione:

- Se B ha un significato applicativo, anche senza la A e viceversa allora B è un’entità indipendente da A e A è un’entità indipendente da B, pertanto A e B devono essere considerate due file logici.
- Se B non ha un significato applicativo senza la A, allora B è un’entità dipendente da A, e A e B devono essere raggruppate in un unico file logico.

Per accertare la proprietà di (in-)dipendenza tra due candidati Logical Files analizzare i requisiti di cancellazione di occorrenze dei candidati, come previsto dal manuale dello standard.

A titolo esemplificativo si riporta la seguente tabella:

Tipo di relazione tra le due entità, A e B	Se vale la condizione:	Allora conta come LF, con RET:
(1) : (N)	(A e B sono indipendenti)	2 LF
1 : N	B è un’entità dipendente da A	1 LF, 2 RET
	B è un’entità indipendente da A	2 LF
1 : (N)	B è un’entità dipendente da A	1 LF, 2 RET
	B è un’entità indipendente da A	2 LF
(1) : N	A è un’entità dipendente da B	1 LF, 2 RET
	A è un’entità indipendente da B	2 LF
(1) : (1)	(A e B sono indipendenti)	2 LF
1 : 1	(A e B sono dipendenti)	1 LF, 1 RET
1 : (1)	B è un’entità dipendente da A	1 LF, 1 or 2 RET
	B è un’entità indipendente da A	2 LF
(N) : (M)	(A e B sono indipendenti)	2 LF
N : M	B è un’entità dipendente da A	1 LF, 2 RET
	B è un’entità indipendente da A	2 LF
N : (M)	B è un’entità dipendente da A	1 LF, 2 RET
	B è un’entità indipendente da A	2 LF

Legenda

FL = File Logici (ILF or EIF)

(..) = Lato opzionale della relazione

RET = Tipo di elemento record



4.1. INTERNAL LOGICAL FILE (ILF)

4.1.1. DEFINIZIONE

Si definisce Internal Logical File (ILF) un gruppo di dati logicamente collegati o di informazioni di controllo, riconoscibili dall'utente, mantenuti nell'ambito dei confini dell'Applicazione. L'intento primario di un ILF è contenere dati mantenuti da uno o più processi elementari dell'applicazione sottoposta al conteggio.

Il termine "mantenuto" indica la capacità di aggiungere, modificare o cancellare dati attraverso l'uso di funzionalità sviluppate nell'ambito dell'Applicazione.

Le "informazioni di controllo" sono dati utilizzati dall' Applicazione per assicurare la conformità con i requisiti funzionali specificati dall' utente. In altri termini si tratta di informazioni di guida per l'esecuzione di un processo elaborativo dell'applicazione. Nel caso degli ILF questi dati, regole o parametri devono essere memorizzati e mantenuti all' interno dell' Applicazione.

Come illustrato in Figura 1 al paragrafo 2.3 si parla di Internal Logical File allorchè i dati sono mantenuti all' interno del confine dell' Applicazione.

4.1.2. CRITERI PER L'INDIVIDUAZIONE

Perché il gruppo di dati o le informazioni di controllo possano essere definiti un ILF devono essere applicabili tutte le seguenti regole :

- è un gruppo di dati logico, ovvero identificabile dall' utente, che soddisfa determinati requisiti utente;
- è mantenuto all' interno del confine dell' Applicazione;
- è modificato o mantenuto attraverso un processo elementare dell' Applicazione;
- non è contato contemporaneamente come un EIF per l' Applicazione stessa.

Delle tre categorie di tipo Dati precedentemente indicate solo le prime due verificano tutte le regole elencate; la terza categoria invece non verifica la prima regola in quanto i requisiti che portano alla identificazione di un archivio di decodifica non sono di natura funzionale ma tecnica o di qualità secondo quanto previsto dallo standard ISO 14143, recepito dall'IFPUG.

Si specifica che, tipicamente, i dati di decodifica sono i Dati delle Liste o Dati di Traduzione e generalmente sono considerati tali se finalizzati al Data entry; spesso tali dati sono identificati dallo sviluppatore in risposta a uno o più requisiti di qualità dell'utente, altre volte sono identificati in fase di progettazione per soddisfare i requisiti tecnici.

Tipici dati di decodifica sono ad esempio
Stato

- Codice dello Stato
- Descrizione dello Stato

Regione

- Codice della regione
- Descrizione della regione



Sono tabelle di decodifica anche le tabelle con liste di valori validi.

Un ILF va contato solamente una volta, nell'ambito dell'Applicazione; in altre parole, un'Applicazione può usare un ILF (in lettura e/o in aggiornamento) più volte, ma l'ILF va considerato solamente una volta ai fini del conteggio in Function Point.

Un ILF può essere mantenuto, e quindi conteggiato, in più Applicazioni.

Inoltre, nel caso di revisione di un' Applicazione preesistente, devono essere contati solo gli ILF che nascono (ADD), oppure sono cancellati (DEL), oppure sono modificati (CHG).

Tenendo, comunque, presente i criteri citati, i passi per l' individuazione sono :

Per una nuova Applicazione

Esaminare le specifiche che descrivono la base dati riconoscibile dall' utente ed individuare gli oggetti della base dati utilizzati dall'Applicazione in esame.

Per la revisione di un'Applicazione preesistente

Esaminare le specifiche che descrivono la base dati riconoscibile dall' utente ed individuare tutti gli oggetti relativi all'Applicazione a prima e a dopo la revisione. Dal confronto tra le due basi dati si evidenziano gli oggetti aggiunti (ADD) o cancellati (DEL).

L'individuazione di oggetti modificati (CHG) è rilevabile al verificarsi di una delle seguenti condizioni :

- inserimento o eliminazione di un campo oppure variazione del suo formato;
- variazione dei sottoinsiemi logici dei dati (vedi paragrafo 4.1.3).

Ogni oggetto aggiunto, modificato o cancellato individua un ILF se è accaduto in aggiornamento o inserimento o cancellazione.

Gli altri oggetti preesistenti (ILF) non vanno considerati ai fini del conteggio, ma vengono presi in considerazione per valutare la complessità delle funzioni (EI, EO, EQ) attraverso gli FTR (vedi elementi di tipo Funzione).

4.1.3. COMPLESSITÀ

Gli elementi che concorrono a determinare la complessità di un Internal Logical File sono: i Record Element Type (RET) e i Data Element Type (DET).

RET: Un *tipo di elemento record* (RET) è un sottogruppo, riconoscibile dall'utente, di elementi dati in un ILF.

Ci sono due tipi di sottogruppi:

- Opzionali
- Obbligatori

I *sottogruppi opzionali* sono quelli per cui l'utente ha la possibilità di usarne uno o nessuno durante un processo elementare che crea o aggiunge un'occorrenza di dati.

I *sottogruppi obbligatori* sono quelli tra cui l'utente deve utilizzarne almeno uno.

DET: campo unico riconoscibile dall'utente, non ripetuto, utilizzato dall'Applicazione.



Contare un RET per ciascun sottogruppo di un ILF. Se non ci sono sottogruppi contare l'ILF come un solo RET. Qualora l'ILF sia padre di una gerarchia ISA (si intende per essa la derivazione di alcune entità da altre per successivi livelli di specificazione; sono utilizzate al fine di evidenziare relazioni valide unicamente con entità gerarchicamente dipendenti e non quindi ai soli fini esplicativi), contare come RET solo il numero di figli terminali della gerarchia.

Qualora l'ILF contenga entità attributive aggiungere un RET per ciascuna di esse.

Contare tanti DET quanti sono gli attributi dell'ILF che rispettano le seguenti regole:

- attributo significativo per l'utente (per esempio, una data fisicamente memorizzata su più campi, va considerata come un unico DET);
- attributo non ricorsivo (per esempio, una descrizione memorizzata su più occorrenze, va considerata come un unico DET).

Contare un DET per ogni campo unico, riconoscibile dall'utente, non ripetuto, mantenuto in o reperito da un ILF attraverso l'esecuzione di un processo elementare.

Quando due applicazioni mantengono e/o referenziano lo stesso ILF, ma ciascuna mantiene/referenzia DET distinti, per dimensionare l'ILF contare solo i DET effettivamente utilizzati da ciascuna applicazione.

Contare come unico DET un'informazione ripetuta su più campi (per esempio: un ILF contiene 12 campi per il valore dell'importo mensile e un campo per il valore del totale annuale, contare due DET, uno per il valore dell'importo mensile, l'altro per il valore dell'importo annuale).

Nel caso di ILF che contengono più oggetti della base dati contare come DET tutti gli attributi (quelli in comune vanno considerati una sola volta).

4.1.4. ESEMPI

Si riportano di seguito alcuni esempi di ILF, tenendo presente che, comunque, devono essere rispettate le regole precedentemente citate.

Sono da considerare come Internal Logical File:

- dati mantenuti nell'ambito dell'Applicazione (per esempio, ogni gruppo logico di dati individuabile all'interno di un archivio fisico e indipendente dai rimanenti dati creato e/o gestito nell'ambito dell'Applicazione);
- dati di sicurezza/verifica mantenuti nell'ambito dell'Applicazione (per esempio, l'archivio delle chiavi di accesso ai dati dell'Applicazione);
- informazioni di controllo mantenute nell'ambito dell'Applicazione (per esempio, un insieme di controlli su come devono essere selezionate e sommate le informazioni su un prospetto);
- dati mantenuti dall'Applicazione in esame ed utilizzati da altre Applicazioni;
- archivi di back-up, solo se richiesti dall'utente (per esempio, archivi di storicizzazione di dati).

Non sono da considerare come Internal Logical File :

- file temporanei, di lavoro o di sort (per esempio, archivi di temporary storage);
- file di cui l'utente non abbia necessità per scopi applicativi e che servono, invece, per implementare requisiti tecnici o di qualità (per esempio dati di decodifica);
- file introdotti dalla tecnologia usata per la realizzazione (per esempio, file che contengono JCL in ambiente MVS o procedure REXX in ambiente dipartimentale);
- file contenenti dati estratti da altri ILF o EIF per essere stampati o visualizzati;
- metodi di accesso alternativi (come indici alternativi);



- file mantenuti da altre Applicazioni e solamente letti od utilizzati come archivi di riferimento dall'Applicazione in esame (vedi External Interface File).

4.2. EXTERNAL INTERFACE FILE (EIF)

4.2.1. DEFINIZIONE

Si definisce External Interface File (EIF) un gruppo di dati logicamente collegati o di informazioni di controllo, riconoscibili dall'utente, referenziati dall'Applicazione, ma mantenuti nell'ambito dei confini di un'altra applicazione. L'intento primario di un EIF è contenere dati referenziati da uno o più processi elementari situati nei confini dell'applicazione sottoposta al conteggio. Ciò significa che un EIF contato per un'applicazione deve essere un ILF, o parte di un ILF, in un'altra applicazione.

Il termine "mantenuto" ed il significato di "informazioni di controllo" sono descritti al paragrafo 4.1.1.; per le informazioni di controllo va considerato che, nel caso degli EIF, questi dati, regole o parametri sono memorizzati e mantenuti all'interno di un'altra Applicazione diversa da quella conteggiata.

Pertanto:

- un EIF per un'Applicazione è sicuramente un ILF per un'altra Applicazione; ciò significa che un EIF contato per una Applicazione deve essere un ILF, o parte di un ILF, in un'altra Applicazione.
- la principale differenza fra un ILF ed un EIF è che il file d'interfaccia esterno **non** è mantenuto dall'applicazione che si sta misurando, mentre un ILF lo è.

Come illustrato in Figura 1 al paragrafo 2.2, si parla di External Interface File quando i dati acceduti sono esterni ai confini dell'Applicazione e l'attività di ricerca dei dati da utilizzare è a carico dell'Applicazione.

4.2.2. CRITERI PER L'INDIVIDUAZIONE

Perché il gruppo di dati o le informazioni di controllo possano essere definiti un EIF devono essere applicabili tutte le seguenti regole :

- è un gruppo di dati logico, ovvero identificabile dall'utente, che soddisfa determinati requisiti utente;
- è referenziato dall'Applicazione che si sta misurando ed è ad essa esterno;
- non è mantenuto dall'Applicazione che si sta misurando;
- è considerato come ILF per almeno un'altra Applicazione;
- non è stato contato come un ILF per l'Applicazione;

Un EIF va contato solamente una volta, nell'ambito dell'Applicazione; in altre parole, un'Applicazione può usare un EIF più volte, ma l'EIF va considerato solamente una volta ai fini del conteggio in Function Point.

Un EIF può essere referenziato, e quindi conteggiato, in più Applicazioni.

Nel caso di revisione di un'Applicazione preesistente, devono essere considerati solo gli EIF aggiunti (ADD), oppure cancellati (DEL), oppure modificati (CHG).

Tenendo, comunque, presente i criteri citati, i passi per l'individuazione sono :



Per una nuova Applicazione

Esaminare le specifiche che descrivono la base dati riconoscibile dall'utente ed individuare gli oggetti della base dati utilizzati dall'Applicazione; individuare un EIF per ogni oggetto acceduto in lettura dall'Applicazione a meno di quelli privi di attributi propri e quelli che risolvono delle entità attributive (vedi anche paragrafo 4.2.3), dove per entità attributiva si intende un oggetto che contiene solo attributi descrittivi di un altro oggetto da cui dipende.

Per la revisione di un'Applicazione preesistente

Esaminare le specifiche che descrivono la base dati riconoscibile dall'utente ed individuare tutti gli oggetti della base dati relativi all'Applicazione a prima e a dopo la revisione, escludendo quelli privi di attributi propri e quelli che risolvono delle entità attributive (vedi anche paragrafo 4.2.3); dal confronto tra le due basi dati si evidenziano gli oggetti aggiunti (ADD) o cancellati (DEL).

L'individuazione di oggetti modificati (CHG) è rilevabile al verificarsi di una delle seguenti condizioni:

- inserimento o eliminazione di un campo oppure variazione del suo formato;
- variazione dei sottoinsiemi logici dei dati (vedi paragrafo 4.2.3).

Ogni oggetto aggiunto, modificato o cancellato individua un EIF se è acceduto in lettura.

Gli altri oggetti preesistenti (EIF) non vanno considerati ai fini del conteggio, ma vengono presi in considerazione per valutare la complessità delle funzioni (EI, EO, EQ) attraverso gli FTR (vedi elementi di tipo Funzione).

4.2.3. COMPLESSITÀ

Gli elementi che concorrono a determinare la complessità di un External Interface File sono: i Record Element Type (RET) e i Data Element Type (DET).

RET: vedi par. 4.1.3

DET: vedi par. 4.1.3

Per il conteggio dei RET valgono le stesse considerazioni esposte per gli Internal Logical File. Le seguenti regole si applicano quando si contano i DET:

- Contare un DET per ogni campo unico, riconoscibile dall'utente, non ripetuto, reperito da un EIF attraverso l'esecuzione di un processo elementare.
- Quando due applicazioni referenziano lo stesso EIF, ma ciascuna referencia DET distinti, per dimensionare l'EIF contare solo i DET effettivamente utilizzati da ciascuna applicazione.
- Contare un DET per ogni singolo dato richiesto dall'utente per stabilire una relazione con un altro EIF.



4.2.4. ESEMPI

Si riportano di seguito alcuni esempi di EIF, tenendo presente che, comunque, devono essere rispettate le regole precedentemente citate.

Sono da considerare come External Interface File:

- dati o informazioni di controllo di riferimento per l'Applicazione, mantenuti nell'ambito di un'altra Applicazione (per esempio, ogni gruppo logico di dati individuabile all'interno di un archivio fisico indipendente dai rimanenti dati mantenuto da un'altra Applicazione esterna ai confini del conteggio; dati utilizzati dall'Applicazione come riferimento per giungere ad una decisione);
- file di help mantenuto all'esterno dell'Applicazione.

Non sono da considerare come External Interface File:

- file temporanei, di lavoro o di sort (per esempio, archivi di temporary storage);
- file di cui l'utente non abbia visibilità o conoscenza;
- file introdotti dalla tecnologia usata per la realizzazione (per esempio, file che contengono JCL in ambiente MVS o procedure REXX in ambiente dipartimentale);
- file contenenti dati estratti da altri ILF o EIF per essere stampati o visualizzati;
- metodi di accesso alternativi (come indici alternativi);
- file già contati come ILF nell'ambito del conteggio dell'Applicazione;
- dati mantenuti dall'Applicazione in esame ed utilizzati da altre Applicazioni (si tratta di ILF);
- dati esterni ai confini dell'Applicazione, ma mantenuti dall'Applicazione in esame (si tratta di ILF);
- dati dell'Applicazione sottoposti a funzioni di elaborazione ed inviati (per esempio attraverso un file) ad un'altra Applicazione (si tratta di output per l'Applicazione ed input per l'altra Applicazione);
- dati di un'altra Applicazione utilizzati per aggiornare un ILF dell'Applicazione (si tratta di EI).

4.3. EXTERNAL INPUT (EI)

4.3.1. DEFINIZIONE

Un External Input (EI) è un processo elementare che elabora dati o informazioni di controllo provenienti dall'esterno del confine dell'Applicazione. L'intento primario di un EI è mantenere uno o più ILF e/o alterare il comportamento del sistema. I dati elaborati mantengono uno o più ILF. Le informazioni di controllo elaborate possono o meno mantenere un ILF.

Un "processo elementare" è la più piccola unità di attività che abbia un significato per l'utente finale nell'ambito dell'Applicazione.

Per esempio, l'evasione di un ordine potrebbe essere scomposta in più sottoprocessi quali la creazione di un importo dovuto, la lettura di un file per verificare la solvibilità del cliente e l'aggiornamento della quantità in magazzino; il processo elementare dell'utente è, però, costituito dall'evasione dell'ordine e non dai sottoprocessi indicati.

Il processo elementare deve essere autonomo (ossia non dipendente da altri processi elementari) e lasciare l'applicazione in uno stato di coerenza funzionale.

Per esempio, l'immissione di un documento può avvenire attraverso tre schermate; tuttavia, se il documento risulta incompleto fino al riempimento di tutti e tre gli schermi, il processo elementare di immissione del documento risulta anch'esso incompleto fino a quando tutte le informazioni non siano state immesse e solo a questo punto il sistema è in uno stato consistente.



Un processo elementare sottende un “trattamento logico”, ossia una logica elaborativa, richiesta dall'utente, che comprenda edit e/o calcoli e/o algoritmi e/o utilizzo di ILF/EIF.

Il termine “mantiene” indica la possibilità di aggiungere, modificare o cancellare dati di un ILF attraverso l'uso di funzionalità sviluppate nell'ambito dell'Applicazione.

Le “informazioni di controllo” sono dati utilizzati dall' Applicazione per assicurare la conformità con i requisiti funzionali specificati dall' utente. In altri termini si tratta di informazioni di guida per l'esecuzione di un processo elaborativo dell'Applicazione. Nel caso degli EI questi dati, regole o parametri possono essere o meno memorizzati, una volta completato il processo.

Come illustrato in Figura 1 al paragrafo 2.3, un External Input può essere generato dall'utente esterno, oppure da funzioni di elaborazione esterne ai confini dell'Applicazione.

4.3.2. CRITERI PER L'INDIVIDUAZIONE

Perché i dati e le informazioni di controllo elaborati possano essere definiti un input esterno (EI) devono essere applicabili tutte le seguenti regole :

- I dati o le informazioni di controllo sono ricevuti dall'esterno del confine dell'applicazione.
- Almeno un ILF deve essere mantenuto se i dati che attraversano il confine non sono informazioni di controllo che alterano il comportamento del sistema.
- Per il processo identificato, deve valere almeno una delle seguenti tre affermazioni:
 - Il trattamento logico è unico rispetto al trattamento logico utilizzato da altri input esterni dell'applicazione.
 - L'insieme degli elementi dati identificato è diverso dagli insiemi identificati per altri input esterni dell'applicazione.
 - Gli ILF o gli EIF referenziati sono diversi dai file referenziati da altri input esterni dell'applicazione.

Tenendo, comunque, presente i criteri citati, i passi per l' individuazione sono :

Per una nuova Applicazione

Esaminare le specifiche che descrivono le funzionalità del sistema e contare un EI per ogni gruppo logico di dati esterni di input a cui corrisponde un aggiornamento della base informativa.

Contare un EI per ogni gruppo logico di dati esterni di input a cui non corrisponde un aggiornamento della base informativa e a cui non risulta associato nessun dato esterno di output (in questo caso siamo in presenza di un EI per informazioni di controllo).

Per la revisione di un'Applicazione preesistente:



Trarre dalle specifiche che descrivono le funzionalità del sistema gli EI aggiunti (ADD), modificati (CHG) o cancellati (DEL).

Un EI si considera modificato (CHG) al verificarsi di una delle seguenti condizioni:

- sono stati modificati i dati utilizzati (aggiunta o eliminazione di campi);
- è stato inserito o eliminato un accesso agli oggetti letti o aggiornati;
- sono stati inseriti o modificati algoritmi di calcolo e/o controlli che variano la logica elaborativa.

4.3.3. COMPLESSITÀ

Gli elementi che concorrono a determinare la complessità di un External Input sono: i File Type Referenced (FTR) ed i Data Element Type (DET).

FTR: EIF letti e ILF letti o mantenuti dall'EI;

DET: campo unico, riconoscibile dall'utente, non ripetuto.

Per una nuova Applicazione:

Contare tanti FTR (ILF e/o EIF) quanti, sono gli ILF/EIF aggiornati o letti durante l'elaborazione dei dati di input (in caso di un ILF sia mantenuto che letto, contare un solo FTR).

Contare un DET per ogni campo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione ed è richiesto per completare l'input esterno.

Non contare campi che sono reperiti o derivati dal sistema e memorizzati su un ILF durante il processo elementare se i campi non attraversano il confine dell'applicazione.

Aggiungere un DET per la capacità di inviare un messaggio di risposta di sistema fuori del confine dell'applicazione per indicare un errore occorso durante l'elaborazione, per confermare che l'elaborazione è stata completata o per verificare che l'elaborazione dovrebbe continuare.

Aggiungere un solo DET per la capacità di specificare un'azione che deve essere eseguita anche se vi sono vari modi per richiamare lo stesso processo logico.

Per la revisione di un'Applicazione preesistente:

Contare per gli EI aggiunti, modificati o cancellati tanti FTR quanti sono (per gli EI cancellati, quanti erano) gli ILF/EIF letti e/o aggiornati dall'EI (in caso di un ILF sia mantenuto che letto, contare un solo FTR).

Per gli EI aggiunti, modificati o cancellati contare tanti DET quanti sono (per gli EI cancellati, quanti erano) i campi riconoscibili dall'utente, non ricorsivi, utilizzati dall'EI, aggiungendo, nel caso di funzioni "on demand", un DET per il comando di attivazione dell'EI e un DET per la presenza di uno o più messaggi di errore/conferma operazione/validazione associati all'EI.



4.3.4. ESEMPI

Si riportano di seguito alcuni esempi di EI, tenendo presente che, comunque, devono essere rispettate le regole precedentemente citate.

Sono da considerare come External Input:

- dati in input all'Applicazione con cui si aggiornano ILF, come :
 - una maschera su terminale,
 - una scheda parametro,
 - un file (su vari supporti quali disco, dischetti, nastri magnetici ecc.) prodotto da una funzione di un'altra Applicazione;
- informazioni di controllo che entrano nell'Applicazione indipendentemente dal fatto che aggiornino o meno un ILF (per esempio, comunicazione utente di attivazione di un batch ad una determinata data e ora);
- funzioni utente (per esempio, in ambiente WINDOWS, le funzioni "crea", "aggiungi", "aggiorna", "cancella", "salva/salva con nome").

Non sono da considerare come External Input:

- input di un inquiry;
- schermi di navigazione di un menù (non mantengono un ILF);
- modalità diverse di attivazione di uno stesso input logico;
- la parte di output in uno schermo di I/O.

In generale :

- uno schermo di immissione dati che attivi una singola funzione di elaborazione (per esempio, uno schermo per attivare una funzione di inserimento o aggiornamento) va contato come un unico EI;
- una sequenza di schermi di immissione dati, ognuno dei quali non richieda un trattamento specifico, che venga elaborata globalmente da una singola funzione, è un unico EI;
- uno schermo di immissione dati che consenta l'attivazione di più funzioni di elaborazione: contare tanti EI quante sono le funzionalità attivabili tramite lo schermo (per esempio, uno schermo che consenta di eseguire le funzioni di inserimento, aggiornamento e cancellazione, va contato come tre EI);
- nel caso di un inquiry seguito da un input che permetta l'aggiornamento di uno o più ILF, siamo in presenza di un EQ (per la funzione di inquiry) ed un EI (per la funzione di aggiornamento);
- nel caso di una procedura batch di aggiornamento, per individuare gli EI, si può operare come segue:
 - verificare che la procedura sia scomponibile o meno in funzionalità logiche significative per l'utente (in altre parole, ogni funzionalità identificata deve realizzare una specifica richiesta utente, cioè deve essere collegata ad un "evento" per l'utente);
 - individuare per ogni funzionalità i relativi dati di input;
 - conteggiare infine un EI per ciascuna funzionalità, solo se dotata di input (a tale proposito, tenere presente che, nel caso in cui si prelevino da un ILF di un'altra Applicazione informazioni, non in forma real time con cui variare i dati dell'Applicazione in esame, tali dati rappresentano un input per la procedura, e non un EIF).

4.4. EXTERNAL OUTPUT (EO)

4.4.1. DEFINIZIONE

Un External Output (EO) è un processo elementare che genera dati o informazioni di controllo all'esterno del confine dell'Applicazione. L'intento primario di un EO è presentare informazioni ad un utente utilizzando un trattamento logico diverso da, o in aggiunta a, il semplice reperimento di dati o



informazioni di controllo. Il trattamento logico deve contenere almeno una formula matematica o un calcolo, creare dati derivati, mantenere uno o più ILF o alterare il comportamento del sistema.

Per il significato di “processo elementare” e “informazioni di controllo” vedi il paragrafo 4.3.1.

Per le informazioni di controllo va considerato che, nel caso degli EO, questi dati, regole o parametri devono essere inviati ad un utente o ad un’ altra Applicazione.

E’ un EO qualsiasi output che contenga dati derivati (vedi anche paragrafo 4.5).

Per “dato derivato” si intende un dato che richiede un trattamento diverso dal semplice reperimento e presentazione delle informazioni provenienti da ILF/EIF.

Come illustrato in Figura 1 al paragrafo 2.3, le informazioni contenute in un External Output sono generate da funzioni di elaborazione interne ai confini dell'Applicazione e sono destinate all'utente esterno, oppure ad un'altra Applicazione.

4.4.2. CRITERI PER L’INDIVIDUAZIONE

Perché il processo elementare possa essere definito un EO devono essere applicabili tutte le seguenti regole :

- La funzione invia dati o informazioni di controllo all’esterno del confine dell’applicazione.
- Per il processo identificato, deve valere almeno una delle seguenti tre affermazioni:
 - Il trattamento logico è unico rispetto al trattamento logico utilizzato da altri output esterni o interrogazioni esterne dell’applicazione.
 - L’insieme dei dati elementari identificato è diverso dagli insiemi identificati per altri output esterni o interrogazioni esterne dell’applicazione.
 - Gli ILF o gli EIF referenziati sono diversi dai file referenziati da altri output esterni o interrogazioni esterne dell’applicazione.
-
- Il trattamento logico del processo elementare contiene almeno una formula matematica o un calcolo.
- Il trattamento logico del processo elementare crea dati derivati.
- Il trattamento logico del processo elementare mantiene almeno un ILF.
- Il trattamento logico del processo elementare altera il comportamento del sistema.

Tenendo, comunque, presente i criteri citati, i passi per l’ individuazione sono :

Per una nuova Applicazione

Esaminare le specifiche che descrivono le funzionalità del sistema e contare un EO (ADD) per ogni gruppo logico di dati che viene inviato all’ esterno dell’ Applicazione.

Per la revisione di un'Applicazione preesistente:

Trarre dalle specifiche che descrivono le funzionalità del sistema gli EO aggiunti (ADD), modificati (CHG) o cancellati (DEL).

Un EO si considera modificato (CHG) al verificarsi di una delle seguenti condizioni:

- sono stati modificati i dati in output (aggiunta o eliminazione di campi);
- è stato inserito o eliminato un accesso agli oggetti letti;
- sono stati inseriti o modificati algoritmi di calcolo e/o controlli che variano la logica elaborativa.



4.4.3. COMPLESSITÀ

Gli elementi che concorrono a determinare la complessità di un External Output sono: i File Type Referenced (FTR) ed i Data Element Type (DET).

FTR: ILF mantenuti e letti ed EIF letti dall'EO.

DET: campo unico, riconoscibile dall'utente, non ripetuto.

Per una nuova Applicazione:

Contare tanti FTR quanti sono gli ILF mantenuti e letti dall'EO e gli EIF letti dall'EO. Contare un solo FTR per ogni ILF che è sia mantenuto che letto.

Contare un DET per ogni campo riconoscibile dall'utente, non ripetuto, che entra nel confine dell'applicazione ed è richiesto per specificare quando, cosa e/o come i dati devono essere reperiti o generati dal processo elementare.

Contare un DET per ogni campo riconoscibile dall'utente, non ripetuto, che esce dal confine dell'applicazione.

Se un DET entra ed esce dal confine dell'applicazione, contarli solo una volta per il processo elementare.

Contare un solo DET per la capacità di inviare un messaggio di risposta di sistema fuori del confine dell'applicazione, per indicare un errore occorso durante l'elaborazione, per confermare che l'elaborazione è stata completata o per verificare che l'elaborazione dovrebbe continuare.

Contare un solo DET per la capacità di specificare un'azione che deve essere eseguita anche se vi sono vari modi per richiamare lo stesso processo logico.

Non contare i campi che sono reperiti o derivati dal sistema e memorizzati su un ILF durante il processo elementare se i campi non attraversano il confine dell'applicazione.

Non contare come DET:

- costanti;
- variabili di paginazione;
- variabili generate da sistema come data e ora;
- comandi di paginazione come: precedente, successivo, freccia di paginazione su di una applicazione GUI.

Per la revisione di un'Applicazione preesistente:

Per EO aggiunti, modificati o cancellati contare tanti FTR quanti sono (per gli EO cancellati, quanti erano) gli ILF/EIF relativi alla funzione.

Per EO aggiunti, modificati o cancellati contare tanti DET quanti sono (per gli EO cancellati, quanti erano) i dati riconoscibili dall'utente, non ricorsivi, che appaiono nell'EO.

Nel conteggio dei DET escludere sempre:

- costanti;
- variabili di paginazione;
- variabili generate da sistema come data e ora.



4.4.4. ESEMPI

Si riportano di seguito alcuni esempi di EO, tenendo presente che, comunque, devono essere rispettate le regole precedentemente citate.

Sono da considerare come External Output:

- un file prodotto dall'Applicazione con modalità "scheduled" e destinato ad un'altra Applicazione;
- stampa con dati derivati (da notare che se, ad esempio, il report contiene più pagine, ma esse non sono richieste e prodotte indipendentemente, si tratta di un unico EO);
- report on-line con dati derivati;
- informazioni di controllo che escono dall' Applicazione (ad esempio un messaggio per avvisare l' utente che sono stati raggiunti certi valori parametrici di data e ora, una notifica a video di un appuntamento).

Non sono da considerare come External Output:

- report identici con valori dei campi differenti;
- help (vedi EQ);
- modalità diverse per ottenere uno stesso output logico;
- messaggi di errore/conferma;
- campi di totale contenuti in un report di dettaglio (da conteggiare nell' ambito dell' EO del report globale);
- ordinamento o predisposizione di dati senza altra elaborazione logica.

In generale:

- un EO veicolato su diversi media device (supporti fisici diversi come, ad esempio, video o stampante) che mantenga gli stessi DET e lo stesso trattamento logico deve essere contato una sola volta : il trattamento logico non può essere considerato "unico" per differenze dipendenti esclusivamente dalle tecnologie usate e che non aggiungono contenuto applicativo ai processi elementari;
- un inquiry con uno o più dati derivati o calcolati va conteggiato come un EO;
- una procedura batch di quadratura va contata come EO solo se riconosciuta e visibile all' utente.

4.5. EXTERNAL INQUIRY (EQ)

4.5.1. DEFINIZIONE

Una Interrogazione Esterna (EQ) è un processo elementare che invia dati o informazioni di controllo all'esterno del confine dell'applicazione. L'intento primario di una Interrogazione Esterna è presentare informazioni ad un utente attraverso il reperimento di dati o informazioni di controllo da un ILF o EIF. Il trattamento logico non contiene formule matematiche o calcoli e non crea dati derivati. Nessun ILF è mantenuto durante l'elaborazione e il comportamento del sistema non risulta modificato.

Per il significato di "processo elementare" e "dato derivato" vedi 4.4.1

Come illustrato in Figura 1 al paragrafo 2.3, un External Inquiry può essere generato dall'utente esterno, oppure da funzioni di elaborazione esterne ai confini dell'Applicazione.



4.5.2. CRITERI PER L'INDIVIDUAZIONE

Perché il processo elementare possa essere definito un EQ devono essere applicabili tutte le seguenti regole :

- La funzione invia dati o informazioni di controllo all'esterno del confine dell'applicazione.
- Per il processo identificato, deve valere almeno una delle seguenti tre affermazioni:
 - Il trattamento logico è unico rispetto al trattamento logico utilizzato da altri output esterni o interrogazioni esterne dell'applicazione.
 - L'insieme dei dati elementari identificato è diverso dagli insiemi identificati per altri output esterni o interrogazioni esterne dell'applicazione.
 - Gli ILF o gli EIF referenziati sono diversi dai file referenziati da altri output esterni o interrogazioni esterne dell'applicazione.
- Il trattamento logico del processo elementare reperisce dati o informazioni di controllo da almeno un ILF o un EIF.
- Il trattamento logico del processo elementare non contiene formule matematiche o calcoli.
- Il trattamento logico del processo elementare non crea dati derivati.
- Il trattamento logico del processo elementare non mantiene alcun ILF.
- Il trattamento logico del processo elementare non altera il comportamento del sistema.

Tenendo, comunque, presente i criteri citati, i passi per l'individuazione sono :

Per una nuova Applicazione

Esaminare le specifiche che descrivono le funzionalità del sistema e contare un EQ per ogni combinazione di dati con le seguenti caratteristiche:

- assenza di aggiornamenti della base informativa;
- assenza di dati derivati.

Per la revisione di un'Applicazione preesistente:

Trarre dalle specifiche che descrivono le funzionalità del sistema gli EQ aggiunti (ADD), modificati (CHG) o cancellati (DEL).

Un EQ si considera modificato (CHG) se si verifica:

- una modifica dei dati reperiti;
- l'aggiunta o l'eliminazione di un accesso agli oggetti letti;
- sono stati inseriti o modificati controlli che variano la logica elaborativa.

4.5.3. COMPLESSITÀ

Gli elementi che concorrono a determinare la complessità di un External Inquiry sono: i File Type Referenced (FTR) ed i Data Element Type (DET).

FTR: ILF ed EIF letti dall'EQ.

DET: campo unico, riconoscibile dall'utente, non ripetuto.



Contare un DET per ogni campo riconoscibile dall'utente, non ripetuto, che entra nel confine dell'applicazione ed è richiesto per specificare quando, cosa e/o come i dati devono essere reperiti o generati dal processo elementare.

Contare un DET per ogni campo riconoscibile dall'utente, non ripetuto, che esce dal confine dell'applicazione.

Se un DET entra ed esce dal confine dell'applicazione, contarli solo una volta per il processo elementare.

Contare un solo DET per la capacità di inviare un messaggio di risposta di sistema fuori del confine dell'applicazione, per indicare un errore occorso durante l'elaborazione, per confermare che l'elaborazione è stata completata o per verificare che l'elaborazione dovrebbe continuare.

Contare un solo DET per la capacità di specificare un'azione che deve essere eseguita anche se vi sono vari modi per richiamare lo stesso processo logico.

Non contare come DET:

- costanti;
- variabili di paginazione;
- variabili generate da sistema come data e ora;
- comandi di paginazione come: precedente, successivo, freccia di paginazione su di una applicazione GUI.

4.5.4. ESEMPI

Si riportano di seguito alcuni esempi di EQ, tenendo presente che, comunque, devono essere rispettate le regole precedentemente citate.

Sono da considerare come External Inquiry:

- richiesta dell'utente di informazioni contenute in un archivio e relativa risposta (per esempio interrogazioni on-line);
- funzioni utente (per esempio, in ambiente WINDOWS, le funzioni "apri un file", "trova", "browse", "print");
- inquiry propedeutici ad una funzione di aggiornamento di dati online;
- richiesta di Help e risposta;
- "bottoni" per interrogare dati su cui effettuare una selezione.

Non sono da considerare come External Inquiry:

- EQ identici con valori dei campi differenti;
- modalità diverse di selezione di uno stesso inquiry logico (per esempio, attivazione di un inquiry digitando "I" oppure tramite un tasto funzionale);
- schermi di navigazione in un menù (infatti, tali schermi non danno luogo ad accessi a dati della base informativa, ma permettono solamente di selezionare delle funzionalità);
- dati derivati ottenuti da un'attività di ricerca dati (vedi External Output);
- predisposizione dello stesso insieme di dati secondo diversi ordinamenti.

In generale :

- nel caso di combinazioni multiple di richieste di prospettazione, non è il calcolo matematico delle possibili combinazioni dei dati di input a fornire il numero di EQ, ma occorre far riferimento ai raggruppamenti logici diversi dal punto di vista utente;
- uno schermo di attivazione che consenta di selezionare fra più funzioni di interrogazione, dà luogo ad altrettanti EQ;



- le funzioni di interrogazione del Tutorial dei diagnostici, help di funzione, help di campo, sono funzionalità distinte, ma comuni all'intera Applicazione: ognuna di esse è un distinto EQ nell'ambito dell'Applicazione;
- un EQ veicolato su diversi media device (supporti fisici diversi come, ad esempio, video o stampante) che mantenga gli stessi DET e lo stesso trattamento logico deve essere contato una sola volta : il trattamento logico non può essere considerato “unico” per differenze dipendenti esclusivamente dalle tecnologie usate e che non aggiungono contenuto applicativo ai processi elementari.



5. RIEPILOGO PASSI PER IL CONTEGGIO DEI FP

I passi da seguire per effettuare il conteggio dei FP sono i seguenti :

1. Individuare il tipo di conteggio secondo quanto riportato nella sezione 2, al paragrafo 2.4;
2. Definire i confini dell' Applicazione secondo quanto riportato nella sezione 3;
3. Individuare e classificare gli Elementi di tipo Dati e Funzione seguendo i criteri indicati nella sezione 4; per determinare la complessità di ciascun Elemento far riferimento alle tabelle riportate in allegato 2;
4. Riportare gli Elementi di tipo Dati e Funzione, raggruppati per tipologia e livello di classificazione (basso, medio, alto) nella tabella in allegato 3 e calcolare il totale degli UFP dell' Applicazione;
5. Applicare il fattore di aggiustamento (VAF) secondo quanto indicato nella sezione 2;
6. Effettuare il calcolo dei Punti Funzione (FP) dell' Applicazione utilizzando le formule riportate nella sezione 2, al paragrafo 2.4.

Nel caso in cui il conteggio delle dimensioni di un progetto impatti su più Applicazioni occorrerà eseguire i passi descritti per ciascuna di esse e sommare i valori ottenuti, ottenendo così il numero totale di FP dell' intervento.



6. MISURA DELLA “BASELINE”

Il metodo dei Punti Funzione può essere applicato anche per il conteggio di un' Applicazione già installata, definito anche calcolo della “baseline” : inizialmente coincide con i FP della nuova Applicazione e poi deve essere aggiornato ad ogni revisione dell' Applicazione preesistente.

Per il conteggio iniziale di un' Applicazione già installata si usa la seguente formula :

$$AFP_{baseline} = ADD * VAF$$

dove :

ADD è il numero di UFP relativo alle funzionalità che sono state installate con il progetto di sviluppo;
VAF è il valore del Fattore di Aggiustamento.

Per il ricalcolo del numero di FP di un' Applicazione, dopo un progetto di revisione che ne ha variato le funzionalità, si applica la seguente formula :

$$AFP_{baseline} = [(UFPB + ADD + CHGA) - (CHGB + DEL)] * VAFA$$

dove:

UFPB è il numero di Unadjusted Function Point dell' Applicazione, prima dell' intervento;
ADD é il numero di UFP relativi alle funzionalità aggiunte;
CHGA è il numero di UFP relativi alle funzionalità modificate dall'applicazione, valutati al completamente dell'intervento;
CHGB è il numero di UFP relativi alle funzionalità modificate dall'applicazione, valutati prima dell'intervento;
DEL é il numero di UFP relativi alle funzionalità cancellate;
VAFA è il valore del Fattore di Aggiustamento dell'Applicazione dopo l'intervento.



7. ALLEGATO 1

GLOSSARIO GENERALE

ADD	Unadjusted Function Point funzionalità aggiunte
AFP	Adjusted Function Point
AFP _{baseline}	Adjusted Function Point di baseline
C.d.C.	Corte dei Conti
CVS	Ciclo di Vita del Software
CFP	Conversion unadjusted Function Point
CHG	Unadjusted Function Point funzionalità modificate
CHGA	Unadjusted Function Point modificati, dopo l' intervento
CHGB	Unadjusted Function Point modificati, prima dell' intervento
DEL	Unadjusted Function Point funzionalità cancellate
DET	Data Element Type
EI	External Input
EIF	External Interface File
EO	External Output
EQ	External Inquiry
FP	Function Point (Punti Funzione)
FPA	Function Point Analysis
FTR	File Type Referenced
GUFPI-ISMA	Gruppo Utenti Function Point Italia - Italian Software Metric Association
IFPUG	International Function Point User Group
I/O	Input/Output
ILF	Internal Logical File
ISO	International Standard Organization
MAC	Manutenzione Migliorativa Adeguativa Correttiva
MEF	Ministero dell'Economia e delle Finanze
PC	Personal Computer
RET	Record Element Type
S.I.	Sistema Informativo
SIRGS	Sistema Informativo della Ragioneria Generale dello Stato
TDI	Total Degree of Influence
TP	Teleprocessing
UFP	Unadjusted Function Point
VAF	Value Adjustment Factor
VAFA	Value Adjustment Factor, dopo l' intervento
VAFB	Value Adjustment Factor, prima dell' intervento



8. ALLEGATO 2

TABELLE DI VALUTAZIONE DELLE FUNZIONI

EXTERNAL INPUT

File Type Referenced	Data Element Type		
	1-4	5-15	> 15
< 2	Basso	Basso	Medio
2	Basso	Medio	Alto
> 2	Medio	Alto	Alto

EXTERNAL OUTPUT o EXTERNAL INQUIRY

File Type Referenced	Data Element Type		
	1-5	6-19	> 19
< 2	Basso	Basso	Medio
2 o 3	Basso	Medio	Alto
> 3	Medio	Alto	Alto

INTERNAL LOGICAL FILE o EXTERNAL INTERFACE FILE

Record Element Type	Data Element Type		
	1-19	20-50	> 50
< 2	Basso	Basso	Medio
2 - 5	Basso	Medio	Alto
> 5	Medio	Alto	Alto



9. ALLEGATO 3

TABELLA DI CONTRIBUTO AL NUMERO DI UFP

Tipo di Elemento	Grado di complessità			Totale
	Basso	Medio	Alto	
Internal Logical File	__ x 7 = __	__ x 10 = __	__ x 15 = __	_____
External Interface File	__ x 5 = __	__ x 7 = __	__ x 10 = __	_____
External Input	__ x 3 = __	__ x 4 = __	__ x 6 = __	_____
External Output	__ x 4 = __	__ x 5 = __	__ x 7 = __	_____
External Inquiry	__ x 3 = __	__ x 4 = __	__ x 6 = __	_____
UNADJUSTED FUNCTION POINT (UFP)				_____