

# Function Point: Manuale delle Regole di Conteggio

---

Versione 4.3.1





**International Function Point  
Users Group (IFPUG)**

---

**Function Point: Manuale delle Regole di Conteggio**

**Versione 4.3.1**

---

Presidente, Counting Practices Committee  
Adri Timp  
Equens SE, Netherlands  
[cpc@ifpug.org](mailto:cpc@ifpug.org)

---

### **Nota importante sulla traduzione in italiano del CPM**

La traduzione italiana del Manuale delle Regole di Conteggio IFPUG deve considerarsi come un documento a supporto dell'uso della versione originale in inglese e non quale fonte indipendente. I traduttori, i revisori e gli altri soggetti coinvolti nel processo di traduzione, in particolare l'IFPUG, non si assumono alcuna responsabilità riguardo l'esattezza della presente traduzione, in particolare per il suo utilizzo in rapporti commerciali e contrattuali di qualsiasi genere, inclusa quale fonte di informazione per l'esame di certificazione IFPUG CFPS. Riassumendo, nell'utilizzo, l'utente della presente traduzione lo fa a proprio rischio e pericolo, senza alcuna garanzia, implicita o esplicita, circa l'adeguatezza di questa traduzione per qualsiasi finalità specifica. Qualora l'esattezza delle informazioni sia importante, è necessario utilizzare la versione originale del manuale in lingua inglese, IFPUG Counting Practices Manual v. 4.3.1. Questa traduzione è stata completata il 27/07/2011.

---

© 2010 IFPUG. Tutti i diritti riservati. International Function Point Users Group, 2010. I membri dell'IFPUG possono riprodurre parti di questo documento all'interno dei propri manuali sulle regole di conteggio. Qualora si usino parti del presente documento, è necessario riprodurre il seguente testo sulla copertina del documento derivato: "Questo documento contiene materiale estratto dal manuale IFPUG sulle Regole di Conteggio. È stato riprodotto in questo documento con il permesso dell'IFPUG".

ISBN 978-0-9830330-1-1

---

---

## Versione 4.3.1, Gennaio 2010

---

Questa versione sostituisce la versione 4.2.1, che è ora obsoleta.  
Cambiamenti periodici possono essere apportati alle informazioni ivi contenute.

---

### Counting Practices Committee (Comitato per le Regole di Conteggio, CPC IFPUG)

---

Bonnie S. Brown, EDS an HP Company  
Royce Edwards, Software Composition Technologies  
E. Jay Fischer, JRF Consulting, Inc.  
David Garmus, The David Consulting Group  
Janet Russac, Software Measurement Expertise, Inc.  
Adri Timp, Equens SE, Netherlands  
Peter Thomas, Steria

---

### Gruppo per la traduzione in italiano (Gruppo Utenti Function Point Italia – Italian Software Metrics Association, GUFPI-ISMA)

---

<b>Traduttori e Revisori</b>	Nicola Argentieri, ISTAT
	Ivana Beni, Al maviva
	Luigi Buglione, Engineering.IT
	Marisa Cappella, ISTAT
	Ciro Coppola, Al maviva
	Filippo De Carli, Shared Service Center
	Tommaso Iorio
	Gianfranco Lanza, CSI Piemonte
	Monica Lelli
	Marco Motta, Vodafone Italia
	Mauro Pescara
	Roberto Reggiani
	Carmine Russo, Auselda
	Luca Santillo, Agile Metrics

---

Per informazioni su ulteriori copie di questo manuale contattare:

IFPUG  
191 Clarksville Road  
Princeton Junction, NJ 08550  
U.S.A.  
(609) 799-4900  
E-mail: [ifpug@ifpug.org](mailto:ifpug@ifpug.org)  
Web: <http://www.ifpug.org>

---

---

Il metodo FSM IFPUG come riportato in questo manuale si basa sul CPM versione 4.0, emesso nel 1990. Molte persone hanno partecipato al CPC negli anni, affinando la visione e la comprensione attraverso numerose e complesse discussioni sulla metodologia. Grazie al contributo di tutti i membri attuali e passati, il metodo è evoluto fino all'attuale standard ISO FSM. Il CPM rappresenta la 'stratificazione' dei contributi di tutti coloro che hanno partecipato al CPC. Il CPC è riconoscente per il supporto ed il contributo dei nostri Direttori per gli Standard di Conteggio, tra cui Carol Dekkers, Bruce Rogora e Mary Bradley, e per l'impegno profuso da Carol Dekkers nel far avanzare entrambi i documenti ISO nel processo di approvazione. L'IFPUG estende l'apprezzamento a tutti i membri del CPC, Direttori per gli Standard di Conteggio e Rappresentanti ISO, attuali e passati.

---

#### **CPC - Membri Attuali e Passati**

---

Allan Albrecht, Creatore della FPA  
Kim Albee (CPM 4.0)  
Maarten Barth (CPM 4.0)  
Andy Belden (CPM 4.0)  
Angela Benton (CPM 4.1)  
Mary Bradley (CPM 4.0, CPM 4.1, ex Presidente)  
Bonnie Brown (CPM 4.2, CPM 4.3, Vice Presidente)  
Kevin Chinoy (CPM 4.1)  
Jean-Marc Desharnais (CPM 4.0)  
Rob Donnellan (CPM 4.0)  
Ian Drummond (CPM 4.0)  
Martin D'Souza (CPM 4.2, CPM 4.3)  
Boyd Edmiston (CPM 4.0)  
Royce Edwards (CPM 4.3)  
Peter Fagg (CPM 4.1)  
Jay Fischer (CPM 4.2, CPM 4.3)  
Sean Furey (CPM 4.1)  
Steve Galea (CPM 4.1)  
Barbara Gardner (CPM 4.0)  
David Garmus (CPM 4.0, CPM 4.1, CPM 4.2, CPM 4.3)  
Jim Glorie (CPM 4.1, CPM 4.2)  
Paul Goodman (CPM 4.0)  
Phil Hain (CPM 4.0)  
David Herron (CPM 4.0)  
Steve Hone (CPM 4.2)  
Bob Huckaby (Past Chair)  
Valerie Marthaler (CPM 4.1, CPM 4.2, CPM 4.3, ex Presidente)  
Frank Mazzucco (CPM 4.0)  
Pam Morris (CPM 4.0, CPM 4.1, CPM 4.2)  
Jolijn Onvlee (CPM 4.0, CPM 4.1)  
Bruce Paynter (CPM 4.2)  
Dave Phillips (CPM 4.1)  
Ben Porter (CPM 4.0, ex Presidente)  
Robin Ragland (CPM 4.0, CPM 4.1, ex Presidente)  
Roger Roy (CPM 4.0)  
Eberhard Rudolph (CPM 4.0)  
Grant Rule (CPM 4.1)

---

---

**CPC - Membri Passati (*cont.*)**

---

Bill Rumpf (CPM 4.0)  
Janet Russac (CPM 4.3)  
Michael Schooneveldt (CPM 4.0, CPM 4.1)  
Linda Smith (CPM 4.0, ex Presidente)  
Jack Sprouls (CPM 4.0)  
Denis St. Pierre (CPM 4.0, CPM 4.1)  
Peter Thomas (CPM 4.3)  
Koni Thompson (CPM 4.1, CPM 4.2)  
Adri Timp (CPM 4.1, CPM 4.2, CPM 4.3, Presidente)  
Tony Tiongson (CPM 4.0)  
Stephen Treble (CPM 4.1)  
Eddy van Vliet (CPM 4.1, CPM 4.2, CPM 4.3)  
Terry Vogt (CPM 4.2)  
Gary Walker (CPM 4.0, CPM 4.1)  
Ewa Wasylkowski (CPM 4.0, CPM 4.1)

---

---

Questa pagina è lasciata intenzionalmente vuota.

---



---

## Indice

---

Prefazione	iii
Introduzione al Manuale delle Regole di Conteggio	vii

### Parte 1 FSM

Prefazione	i
Introduzione	iii
Capitolo 1	Ambito di applicazione..... 1
Capitolo 2	Riferimenti normativi..... 1
Capitolo 3	Termini e definizioni..... 1
Capitolo 4	Acronimi..... 8
Capitolo 5	Processo di Misurazione..... 8
Allegato A	A-1

### Parte 2 Collegamento – Applicare il metodo FSM IFPUG

Capitolo 1	Introduzione..... 1-1
Capitolo 2	Panoramica sul metodo FSM IFPUG..... 2-1
Capitolo 3	Raccogliere la documentazione disponibile..... 3-1
Capitolo 4	Determinare il tipo di conteggio..... 4-1
Capitolo 5	Determinare l'ambito del conteggio e i confini applicativi..... 5-1
Capitolo 6	Misurare le funzioni di tipo dati..... 6-1
Capitolo 7	Misurare le funzioni di tipo transazionale ..... 7-1
Indice analitico	i-1

## **Parte 3 Prassi di conteggio**

Introduzione		1-1
Capitolo 1	Dati di decodifica.....	1-3
Capitolo 2	File logici.....	2-1
Capitolo 3	Dati condivisi.....	3-1
Capitolo 4	Progetti di manutenzione evolutiva e attività di manutenzione ordinaria.....	4-1
Capitolo 5	Attività di conversione dei dati.....	5-1
Indice analitico		i-1

## **Parte 4 Esempi**

Introduzione		1-1
Capitolo 1	Esempi di conteggio delle funzioni di tipo dati.....	1-3
Capitolo 2	Esempi di conteggio delle funzioni di tipo transazionale.....	2-1
Indice analitico		i-1

## **Parte 5 Appendici e Glossario**

Appendice A	Tabella per il calcolo della dimensione funzionale.....	A-1
Appendice B	Cambiamenti dalla precedente versione.....	B-1
Appendice C	Dimensione funzionale pesata.....	C-1
Glossario		G-1

---

## Prefazione

---

- Introduzione** L'uso dei function point come misura della dimensione funzionale dei sistemi informativi è cresciuto nel decennio trascorso, passando da poche organizzazioni interessate ad una consistente lista di organizzazioni in tutto il mondo.
- IBM CIS & A Guidelines 313** Verso la fine degli anni '70, Allan Albrecht dell'IBM definì i concetti che permettono di misurare l'output dei progetti di sviluppo software. Queste definizioni sono state estese nel documento *IBM CIS & A Guideline 313, AD/M Productivity Measurement and Estimate Validation*, pubblicato il giorno 1 novembre 1984.
- Versione 2.0** Col diffondersi dell'uso dei function point si è avuta una sempre più ampia applicazione di tale misura. Ciò ha permesso di effettuare dei test sulla descrizione originale della misura e ha reso necessario creare delle linee guida per interpretare le regole originali per nuovi ambienti. Ciò si è concretizzato nella Versione 2.0 (Aprile 1988) dell'*International Function Point Users Group (IFPUG) Function Point Counting Practices Manual*.
- Versione 3.0** La Versione 3.0 (Aprile 1990) del *Function Point Counting Practices Manual* dell'IFPUG è stata una delle principali pietre miliari nell'evoluzione delle misure di dimensioni funzionali. Per la prima volta il Comitato dell'IFPUG per le Regole del Conteggio si è impegnato nel passaggio da una serie di varie interpretazioni delle regole a un documento coerente che rappresentasse un punto di vista comune sulle regole di conteggio dei function point. In tal senso è stato fatto il primo passo per stabilire degli standard sulle misure dei function point che ne permettessero l'applicazione in organizzazioni diverse.

**Versione 4.0** La versione 4.0 (Function Point: Manuale sulle Regole di Conteggio, IFPUG, Gennaio 1994) ha costituito la successiva pietra miliare nell'evoluzione delle misure di dimensioni funzionali. Questa versione rispecchiava l'utilizzo dei function point come stima della dimensione di un progetto, sin dalle fasi preliminari dello sviluppo, attraverso l'uso delle tecniche di ingegneria del software. La crescita rapida del numero di applicazioni con finestre ad interfaccia grafica è stata una spinta ad includere in tale versione il conteggio delle GUI (Graphical User Interface). Poiché le attività di conteggio incontravano una maggiore varietà di situazioni, questa versione ha posto enfasi sull'interpretazione e sulla pratica dell'uso delle regole di conteggio. Alla documentazione sono stati inclusi vari esempi e sono stati aggiunti dei casi di studio. Infine, la versione 4.0 proseguiva nel chiarire il conteggio dei function point e nel migliorarne la coerenza.

**Versione 4.1** La versione 4.1 (Gennaio 1999) ha fornito chiarimenti sulle regole esistenti, regole nuove o modificate che affrontano situazioni in precedenza non documentate, ed inoltre fornisce nuovi suggerimenti ed esempi per facilitare la comprensione. Il Comitato per le Regole del Conteggio IFPUG ha esaminato ed elaborato le richieste dei membri, seguendo il Processo di Revisione del Manuale contenuto nel Capitolo 1 di questo Manuale.

Le variazioni comprese nella versione 4.1 chiariscono:

- l'identificazione di un utente, di un processo elementare, e delle informazioni di controllo;
- la differenziazione tra Output Esterni (EO) e Interrogazioni Esterne (EQ);
- l'identificazione dei Tipi di Elementi Dati (DET) e dei Tipi di Elementi Record (RET) per le funzioni di tipo dati;
- l'identificazione dei Tipi di Elementi Dati (DET) per le funzioni di tipo transazionale.

La versione 4.1 continua il processo di chiarimento e di miglioramento della coerenza del conteggio dei function point.

Infine, con l'eccezione delle 14 Caratteristiche Generali del Sistema, la versione 4.1 è stata progettata per essere conforme agli standard ISO se e quando una qualsiasi guida di conformità dovesse divenire uno standard.

**Versione 4.1.1** La versione 4.1.1 (Aprile 2000) ha apportato alcune correzioni editoriali e tipografiche.

**Versione 4.2** La versione 4.2 non modifica alcuna regola precedentemente emessa, ma fornisce chiarimenti e interpretazioni evolute delle regole esistenti, che aumenteranno ulteriormente la coerenza tra i conteggiatori.

La procedura e le regole dell'analisi dei function point (FPA – Function Point Analysis) dell'IFPUG sono concise e facili da usare. Per rispettare tali caratteristiche e per rendere il Manuale delle Regole di Conteggio (CPM – Counting Practices Manual) ancor più pratico come manuale di riferimento, il Comitato per le Regole di Conteggio (CPC – Counting Practices Committee) ha deciso di ristrutturare il CPM 4.2 in quattro parti:

1. Procedura e Regole
2. Prassi di Conteggio
3. Esempi
4. Appendici

Per aiutare gli utenti della FPA in maniera tempestiva, il CPC ha inizialmente pubblicato i risultati delle proprie ricerche in testi distinti, affiancati al CPM:

- *“Counting Logical Files”* (Settembre 2001)
- *“FPA in an Enhancement Environment”* (Aprile 2002)
- *“Counting Code Data”* (Settembre 2003)
- *“Counting Shared Data”* (Settembre 2003)

Questi documenti sono stati ora incorporati come capitoli della Parte 2 del CPM.

Il CPC ritiene che il CPM 4.2, con le sue nuove linee guida e esempi, garantirà risultati più coerenti tra i CFPS (Certified Function Point Specialist).

**Versione 4.2.1** La versione 4.2.1 (Gennaio 2005) ha apportato alcune correzioni tipografiche.

**Versione 4.3** Nella versione 4.3 (Gennaio 2010), la procedura e le regole dell'analisi dei function point (FPA – Function Point Analysis) dell'IFPUG sono concise e facili da applicare. Per rispettare tali caratteristiche e per rendere il Manuale delle Regole di Conteggio (CPM – Counting Practices Manual) ancor più pratico quale manuale di riferimento, il Comitato per le Regole di Conteggio (CPC – Counting Practices Committee) ha deciso di ristrutturare il CPM 4.3 in modo conforme con gli standard di presentazione ISO. Inoltre, la versione 4.3 contiene lievi modifiche e fornisce nuovi esempi, chiarimenti e interpretazioni migliorate delle regole esistenti, che permetteranno di aumentare ulteriormente la coerenza tra conteggiatori diversi.

Il CPC ritiene che il CPM 4.3, con il suo nuovo formato ed esempi garantirà risultati più coerenti tra i Certified Function Point Specialist.

**Versione 4.3.1** La versione 4.3.1 (Gennaio 2010) ha apportato alcune correzioni tipografiche ed ha incorporato revisioni editoriali dell'ISO.

**Versioni  
future**

Questo documento deve essere considerato in evoluzione. Si deve ancora definire come misurare i nuovi ambienti, man mano che essi saranno introdotti. Si deve essere in grado di far ciò mantenendo la validità dei conteggi già effettuati. Ciò non sarà un compito facile, anche se sarà un compito essenziale, se si sarà in grado di misurare il progresso che si sta facendo nel trasmettere valore agli utenti e alle organizzazioni che essi rappresentano.

Il Comitato per le Regole del Conteggio desidera ringraziare tutti coloro che hanno contribuito alla nostra ricerca e alla produzione di questo manuale.

*Adri Timp*

Presidente, Counting Practices Committee

---

# Introduzione al Manuale delle Regole di Conteggio

---

**Introduzione** Questa introduzione definisce gli obiettivi del manuale ed il suo processo di revisione. Sono inoltre descritte altre pubblicazioni correlate al manuale.

**Contenuti** Questo capitolo include le seguenti sezioni:

Argomento	Pagina
Obiettivi del Manuale delle Regole di Conteggio	viii
Documenti utilizzati per la versione 4.3	viii
Lettori a cui è rivolto	ix
Struttura del Manuale delle Regole di Conteggio	x
Processo di revisione del manuale	xi
Frequenza delle modifiche	xi
Processo di modifica	xi
Documentazione IFPUG complementare	xii
Requisiti per la formazione	xv

## Obiettivi del Manuale delle Regole di Conteggio

Gli obiettivi principali del *Manuale delle Regole di Conteggio (CPM, Counting Practices Manual) IFPUG, Versione 4.3*, sono:

- essere conforme alla ISO/IEC 14143-1:2007 *Information technology – Software measurement – Functional size measurement – Definition of concepts*;
- fornire una descrizione chiara e dettagliata del conteggio dei function point;
- assicurare che i conteggi siano conformi alle regole di conteggio dei membri affiliati all'IFPUG ;
- fornire una guida per calcolare i function point dagli output delle più comuni metodologie e tecniche;
- fornire una conoscenza comune che permetta ai produttori di tool di realizzare un supporto automatizzato per il calcolo dei function point.

## Documenti utilizzati per la versione 4.3

Per sviluppare questa versione è stata utilizzata la seguente documentazione.

- Il metodo dell'Analisi dei Function Point (FPA – Function Point Analysis) dell'IFPUG è basato su *IBM CIS & A Guideline 313, AD/M Productivity Measurement and Estimate Validation*, del 1 Novembre 1984. La metodologia di conteggio dei function point descritta nel documento “313” è generalmente referenziata come “Albrecht 1984”.
- La versione attuale del manuale, CPM 4.3, è basata principalmente sul *Function Point: Manuale sulle Regole del Conteggio, IFPUG, Versione 4.2.1*.
- Il CPM 4.3 è progettato per essere conforme con ISO/IEC 14143-1:2007 *Information technology – Software measurement – Functional size measurement – Definition of concepts*
- “*Framework for Functional Sizing*”: questo articolo dell'IFPUG spiega che la dimensione del prodotto comprende tre grandezze: dimensione funzionale, dimensione tecnica e dimensione della qualità. Il metodo della FPA dell'IFPUG fornisce una misura della dimensione funzionale.
- I problemi non trattati a sufficienza dai documenti precedentemente citati sono stati risolti dal Comitato dell'IFPUG per le Regole del Conteggio sulla base di varianti delle regole di conteggio esistenti e convalidati attraverso studi di impatto.

Questo manuale, nella sua versione attuale, dovrebbe essere considerato lo standard IFPUG per il calcolo dei function point. È necessario che ciascun membro dell'IFPUG assuma un ruolo attivo nell'assicurare la consistenza del conteggio. L'aderenza dei membri IFPUG a questo standard contribuirà in modo notevole alla consistenza del conteggio.



## **Lettori a cui è rivolto**

Gli standard contenuti in questo manuale dovrebbero essere applicati da chiunque usi l'analisi dei function point per misurare la dimensione funzionale. Il manuale è stato predisposto per essere usato sia da persone inesperte, sia da persone con una media o buona esperienza nella misurazione della dimensione funzionale.

## Struttura del Manuale delle Regole di Conteggio

Il Manuale delle Regole di Conteggio (CPM – Counting Practices Manual) è suddiviso in cinque parti principali:

- Parte 1: Metodo di misurazione della dimensione funzionale FSM-IFPUG
- Parte 2: Collegamento – Applicare il Metodo di Misurazione della Dimensione Funzionale IFPUG
- Parte 3: Prassi di Conteggio
- Parte 4: Esempi
- Parte 5: Appendici e Glossario

### **Parte 1 – FSM**

La Parte 1 costituisce il metodo IFPUG per la misurazione della dimensione funzionale (FSM – Functional Size Measurement). Parlare un idioma al livello di un madrelingua, apprendere la grammatica e i vocaboli da soli non è sufficiente. Essi forniscono solo un quadro di riferimento. Occorre fare esperienza per comprendere come viene parlata una lingua nell'uso pratico, come si applicano le regole grammaticali, quali sono le espressioni idiomatiche più comuni, e così via. Lo stesso vale per la FPA. La conoscenza del processo e delle regole, come presentati nella Parte 1, è necessaria, ma da sola non è una condizione sufficiente per applicare la FPA in maniera corretta. Questo è il motivo per cui il CPM include le parti di seguito riportate.

### **Parte 2 – Collegamento**

La Parte 2 fornisce una guida per il dimensionamento del software conforme al metodo IFPUG per la misurazione della dimensione funzionale (FSM), come illustrato nella Parte 1 del presente CPM IFPUG.

### **Parte 3 – Prassi di Conteggio e Parte 4 –Esempi**

Nelle Parti 3 e 4 sono forniti esempi dettagliati per spiegare i concetti e le regole di conteggio. Ciascun esempio dovrebbe essere considerato a sé stante. Poiché ogni esempio si propone di illustrare uno specifico scenario, ci possono essere varianti tra i vari esempi. Sebbene gli esempi nel manuale trattino argomenti simili tra loro, essi non rappresentano un unico insieme di requisiti utente.

### **Parte 5 – Appendici e Glossario**

La Parte 5 contiene informazioni aggiuntive di rilievo, come le tabelle per il calcolo pronte per l'uso, la descrizione del passaggio dal CPM 4.2 (e CPM 4.2.1) al CPM 4.3, le Caratteristiche Generali del Sistema (GSC) ed il glossario.

In linea di principio, ciascuna parte è valida a sé stante.

## Processo di revisione del manuale

Questa sezione illustra la frequenza con cui sono riportate le modifiche al Manuale delle Regole di Conteggio (CPM) e definisce il procedimento con cui si effettuano le modifiche.

### Frequenza delle modifiche

Una nuova versione del Manuale delle Regole di Conteggio (CPM) può entrare in vigore all'inizio di ciascun anno, nel mese di Gennaio. Essa includerà ogni definizione, regola o prassi di conteggio nuova o variata che sia stata resa definitiva dal Comitato per le Regole del Conteggio (CPC, Counting Practices Committee) rispetto la precedente versione.

### Processo di modifica

Le seguenti attività delineano il processo per aggiungere o variare informazioni nel Manuale delle Regole di Conteggio (CPM). Le spiegazioni per ciascuna attività sono riportate di seguito alla tabella.

Passo	Azione
1	La questione viene presentata al CPC.
2	La questione viene assegnata per l'analisi.
3	Il CPC esamina e discute la questione.
4	Il CPC presenta una proposta di soluzione ai membri dell'IFPUG.
5	Viene attivato uno studio d'impatto per valutare se la modifica proposta ha qualche impatto sui conteggi esistenti.
6	Viene presa la decisione finale.
7	I membri IFPUG vengono informati della decisione.
8	I cambiamenti diventano effettivi e sono riportati nella successiva versione del Manuale delle Regole di Conteggio (CPM).

**Presentazione questione** Il lettore propone idee, cambiamenti o questioni al Comitato per le Regole del Conteggio inviando una email a [ifpug@ifpug.org](mailto:ifpug@ifpug.org) oppure a [cpc@ifpug.org](mailto:cpc@ifpug.org).

<b>Assegnazione ricerca</b>	Uno dei membri del CPC viene incaricato della responsabilità di identificare tutte le alternative, le spiegazioni e l'impatto potenziale di ciascuna alternativa se realizzata. Si effettua un esame approfondito degli standard di conteggio esistenti e degli articoli 'storici' mentre si predispongono possibili alternative. Inoltre, si tenta di determinare cosa potrebbe essere considerato come <i>pratica comune</i> .
<b>Esame del CPC</b>	Il CPC esamina e discute i motivi ed i potenziali impatti di ogni alternativa. Il riesame e la discussione possono dare luogo ad una proposta di cambiamento oppure il riesame può portare il comitato a rifiutare la richiesta di cambiamento.
<b>Soluzione proposta</b>	<p>Viene proposta una soluzione ai membri dell'IFPUG e vengono richiesti dei commenti scritti.</p> <p>Una copia dei cambi proposti viene inviata ai contatti che l'IFPUG ha con le organizzazioni membro. La proposta può anche essere annunciata e diffusa durante un evento IFPUG. Quest'ultimo caso dipende dalla cadenza delle riunioni del CPC piuttosto che dalla schedulazione dell'evento.</p>
<b>Avvio dello studio d'impatto</b>	<p>Il CPC ha adottato un comportamento cauto nell'avviare uno studio d'impatto. Qualora sia possibile che una <i>prassi comune</i> debba cambiare, o che diverse organizzazioni o tipi di applicazioni siano influenzati dal cambio, viene quindi avviato uno studio d'impatto.</p> <p>Il successo dello studio d'impatto è responsabilità di ciascun membro dell'IFPUG. Qualora il CPC riceva risposte scritte indicanti che l'impatto è minimo o assente, lo studio viene interrotto.</p>
<b>Decisione finale</b>	<p>Il comitato prende una decisione finale sulla base dei risultati della ricerca, dei commenti scritti forniti dai membri e dello studio d'impatto.</p> <p>Il comitato può ripetere iterativamente più di una volta i passi dal 2 al 5 (dall'analisi allo studio d'impatto) prima di prendere una decisione finale. La decisione finale può riflettersi quindi in un cambio oppure il comitato può decidere di non operare alcuna variazione.</p>
<b>Comunicazione della decisione</b>	<p>La decisione finale viene comunicata per iscritto ai membri dell'IFPUG attraverso i contatti che l'IFPUG ha con le diverse organizzazioni.</p> <p>Se taluno dei risultati provenienti da uno studio d'impatto contribuisce a prendere una decisione, tali risultati ed una raccomandazione su come minimizzare l'impatto del cambio saranno altresì comunicati.</p>
<b>Data effettiva della decisione</b>	Il Manuale delle Regole di Conteggio (CPM) viene aggiornato al fine di rispecchiare le decisioni ratificate. La data effettiva di validità di tali decisioni sarà quella della nuova versione del manuale (Gennaio successivo).

## Documentazione IFPUG complementare

Il presente Manuale delle Regole di Conteggio (CPM) rappresenta un modulo della documentazione IFPUG. Tutti i documenti si completano tra di loro.

La seguente tabella descrive le altre pubblicazioni.

Documento	Descrizione
IFPUG Brochure (Disponibile)	<p>Questa pubblicazione è una introduzione all'International Function Point Users Group. Essa include una breve descrizione sulla storia dell'organizzazione, introduce all'analisi dei function point e definisce gli scopi dell'IFPUG. La brochure include anche la richiesta di associazione.</p> <p><u>Destinatari:</u> questa pubblicazione è rivolta a tutti coloro i quali vogliono avere informazioni generali sull'IFPUG oppure hanno necessità del modulo di adesione.</p>
IFPUG: Organizational Structure and Services (Disponibile)	<p>Questa pubblicazione descrive i servizi IFPUG, fornisce l'elenco dei membri del consiglio di amministrazione, dei comitati e dei membri affiliati in tutto il mondo.</p> <p><u>Destinatari:</u> questa pubblicazione è rivolta a tutti coloro i quali vogliono ricevere informazioni di base sull'IFPUG.</p>
Linee Guida per la Misurazione del Software (Data di Pubblicazione: Agosto 2004)	<p>Questo manuale fornisce delle generalità sulle metriche del software per organizzazioni che si adoperano a creare o migliorare programmi di misura del software. Il manuale è rivolto sia a responsabili di sistema sia ai clienti, fornisce importanti motivazioni per effettuare la misura del software ed esamina le componenti di programmi di misurazione efficaci</p> <p><u>Destinatari:</u> questo manuale è rivolto ai membri IFPUG, ai coordinatori di attività sui Function Point, alle persone che preparano prospetti per il management ed a coloro i quali hanno precise conoscenze e lavorano direttamente con i function point.</p>
Quick Reference Counting Guide (Data di Pubblicazione: 2010)	<p>Questa guida rapida è un sommario sulle regole e le procedure per il conteggio dei function point.</p> <p><u>Destinatari:</u> queste informazioni riassuntive sono rivolte a chiunque intenda applicare l'analisi dei function point.</p>
Adjusted Functional Size Quick Reference Guide (Data di Pubblicazione: 2010)	<p>Questa guida è un sommario delle Caratteristiche Generali del Sistema (GSC – General Systems Characteristics).</p> <p><u>Destinatari:</u> queste informazioni riassuntive sono rivolte a chiunque intenda utilizzare le Caratteristiche Generali del Sistema (GSC).</p>

<b>Documento</b>	<b>Descrizione</b>
Function Point Analysis Case Studies  (Date di pubblicazione: Case Study 1, Versione 3.0: Settembre 2005 (CPM 4.2)  Case Study 2 Versione 3.0: Marzo 2006 (CPM 4.2)  Case Study 3 Versione 2.0: Settembre 2001 (CPM 4.1)  Case Study 4 Versione 2.0: Settembre 2005 (CPM 4.2) )	I casi di studio illustrano le più importanti tecniche di conteggio che sono incluse nel Manuale delle Regole di Conteggio dei Function Point. I casi di studio illustrano il conteggio dei function point per una applicazione presa ad esempio. I casi di studio illustrano il conteggio dei function point che viene effettuato sia alla fine della fase di analisi sia alla fine della realizzazione del sistema.  <u>Destinatari:</u> i casi di studio sono particolarmente indicati sia per le persone che si avvicinano per la prima volta all'analisi dei function point sia per chi ne ha una media o buona conoscenza.
Glossario IFPUG  (Disponibile con il <i>CPM</i> e con le <i>Linee Guida per la Misurazione del Software</i> )	Questo documento è un glossario che definisce i termini usati nelle varie pubblicazioni IFPUG.  <u>Destinatari:</u> il glossario è raccomandato a chiunque riceva un qualsiasi documento IFPUG o a chiunque necessiti di definizioni di termini IFPUG.
"A Framework for Functional Sizing", IFPUG, Settembre 2003	Questo articolo spiega che la dimensione del prodotto comprende tre possibili dimensioni: dimensione funzionale, tecnica e qualitativa. Il metodo FPA dell'IFPUG fornisce una misura della dimensione funzionale.
"IT Measurement: Practical Advice from the Experts", Addison-Wesley, Aprile 2002	Questo libro rappresenta un'eccellente raccolta di articoli scritti da esperti nel campo dell'Information Technology. È stato curato dall'IFPUG per includere le più recenti prospettive nell'applicazione pratica delle misure e metriche del software.

## Requisiti per la formazione

Analisi sull'usabilità di questa pubblicazione hanno mostrato che la sola lettura del Manuale delle Regole di Conteggio non rappresenta un addestramento sufficiente per applicare il conteggio dei function point a livelli ottimali. Si raccomanda un percorso formativo soprattutto per coloro i quali siano inesperti nel conteggio dei function point.

**Nota:** Per la formazione relativa ai function point, ci si assicuri che il materiale utilizzato sia certificato IFPUG. Contattando l'IFPUG Executive Office al numero (1) 609-799-4900 è possibile richiedere una lista degli istruttori per i corsi di addestramento con materiale certificato.

Oltre alle specifiche informazioni sui function point, il presente manuale include altresì termini utilizzati nelle tecniche di analisi strutturata e di progettazione, come sistemi gestionali ed entità. Il glossario comprende una definizione di tali termini, ma non anche spiegazioni dettagliate su tali tecniche. Pertanto tutto il materiale risulterà non applicabile o utile qualora non si abbia una conoscenza preliminare di tali tecniche.

Questa pagina è lasciata intenzionalmente vuota.



---

# Parte 1 – FSM

---

---

Questa pagina è lasciata intenzionalmente vuota.

---

## Indice

<b>1</b>	<b>Ambito di applicazione .....</b>	<b>1</b>
1.1	Obiettivo.....	1
1.2	Conformità .....	1
1.3	Applicabilità .....	1
1.4	Platea.....	1
<b>2</b>	<b>Riferimenti normativi .....</b>	<b>1</b>
<b>3</b>	<b>Termini e definizioni .....</b>	<b>1</b>
<b>4</b>	<b>Acronimi.....</b>	<b>8</b>
<b>5</b>	<b>Processo di Misurazione .....</b>	<b>8</b>
5.1	Panoramica .....	8
5.2	Raccogliere la documentazione disponibile .....	9
5.3	Determinare l'ambito del conteggio e il confine e identificare i Requisiti Utente Funzionali .....	9
5.4	Misurare le funzioni di tipo dati .....	10
5.4.1	Panoramica .....	10
5.4.2	Identificare e raggruppare tutti i dati logici in funzioni di tipo dati .....	10
5.4.3	Classificare ciascuna funzione di tipo dati come o un ILF o un EIF .....	11
5.4.4	Contare i DET per ciascuna funzione di tipo dati .....	11
5.4.5	Contare i RET per ciascuna funzione di tipo dati .....	12
5.4.6	Determinare la complessità funzionale di ciascuna funzione di tipo dati .....	13
5.4.7	Determinare la dimensione funzionale di ciascuna funzione di tipo dati .....	13
5.5	Misurare le funzioni di tipo transazionale .....	13
5.5.1	Panoramica .....	13
5.5.2	Identificare i processi elementari .....	14
5.5.3	Classificare ciascun processo elementare come una funzione di tipo transazionale .....	16
5.5.4	Contare gli FTR per ciascuna funzione di tipo transazionale .....	18
5.5.5	Contare i DET per ciascuna funzione di tipo transazionale .....	18
5.5.6	Determinare la complessità funzionale di ciascuna funzione di tipo transazionale.....	19
5.5.7	Determinare la dimensione funzionale di ciascuna funzione di tipo transazionale .....	19
5.6	Misurare le funzionalità di tipo conversione .....	19
5.7	Misurare le funzionalità di tipo manutenzione evolutiva .....	20
5.8	Calcolare la dimensione funzionale .....	20
5.9	Documentare il conteggio dei function point.....	21
5.10	Presentare il risultato del conteggio dei function point .....	22
	<b>Allegato A (informativo) Tabelle unificate di complessità e dimensione funzionale.....</b>	<b>23</b>

## Premessa

ISO (the International Organization for Standardization) e IEC (the International Electrotechnical Commission) costituiscono il sistema specializzato per la normazione a livello mondiale. Gli organismi nazionali che sono membri di ISO o di IEC partecipano allo sviluppo di norme internazionali attraverso comitati tecnici stabiliti dalle rispettive organizzazioni per trattare quei particolari ambiti di attività tecniche. I comitati tecnici ISO e IEC collaborano su temi di interesse comune. Anche altre organizzazioni internazionali, di tipo governativo e non, in collegamento con ISO e IEC, partecipano ai lavori.

Le norme internazionali sono preparate secondo le regole stabilite dalla Parte 2 delle Direttive ISO/IEC.

Nel campo dell'Information Technology, ISO e IEC hanno costituito una commissione tecnica congiunta (Joint Technical Committee), ISO/IEC JTC 1. Le bozze degli Standard internazionali (DIS – Draft International Standard) adottati dal JTC sono inviate agli Organismi Nazionali per una votazione. La pubblicazione quali Standard Internazionali richiede l'approvazione di almeno il 75% degli Organismi Nazionali che partecipano al voto.

Si richiama l'attenzione sulla possibilità che alcuni degli elementi di questa norma internazionale possano essere coperti da brevetti. ISO e IEC non devono essere ritenuti responsabili per l'identificazione di tali brevetti.

La norma internazionale ISO/IEC 20926 è stata predisposta dal comitato tecnico congiunto ISO/IEC JTC 1, *Tecnologia dell'Informazione*, Sotto-Comitato SC 7, *Ingegneria del Software e dei sistemi*.

Questa seconda edizione annulla e sostituisce la prima edizione (20926:2004), rivista dal punto di vista tecnico nella sua interezza.

## Introduzione

L'uso dei function point ('punti funzione') quale misura della dimensione funzionale del software è andato crescendo fin dalla metà degli anni '70, passando da poche organizzazioni inizialmente interessate ad una lista divenuta consistente a livello mondiale. Allan Albrecht è stato il primo a definire un metodo per dimensionare il software dal punto di vista funzionale denominato "analisi dei punti funzione". Con lo svilupparsi dell'uso dei function points, si è allargato il campo di applicazione e l'uso della misura. Sin dalla sua costituzione nel 1986 l'International Function Point Users Group (IFPUG) ha migliorato in modo continuo il metodo originale di Albrecht per il dimensionamento funzionale del software. La presente norma internazionale, con quest'ultimo rilascio, intende proseguire nella direzione del miglioramento continuo del metodo IFPUG che promuove l'interpretazione coerente della misurazione della dimensione funzionale in conformità con ISO/IEC 14143-1:2007. Il metodo di misurazione della dimensione funzionale IFPUG è conosciuto come analisi dei function point (FPA, Function Point Analysis) e le sue unità di misurazione prendono il nome di Function Point (FP).

Le Organizzazioni possono usare il presente standard internazionale per misurare la dimensione di prodotti software al fine di:

- supportare analisi di qualità e produttività;
- stimare costi e risorse necessari per lo sviluppo, miglioramento e manutenzione del software;
- fornire un fattore di normalizzazione per confrontare diversi software;
- determinare la dimensione di un pacchetto applicativo acquistato attraverso la quantificazione di tutte le funzionalità incluse nel pacchetto;
- assistere gli utenti nel determinare il beneficio di un pacchetto applicativo per le loro organizzazioni derivante dal dimensionamento funzionale che soddisfi specificatamente i loro requisiti.

L'analisi dei function point (FPA) misura il software quantificando le attività e i servizi (es: funzionalità) che il software fornisce all'utente basandosi principalmente sul disegno logico. Gli obiettivi della FPA sono misurare:

- le funzionalità implementate nel software, che l'utente richiede e riceve;
- le funzionalità impattate dallo sviluppo, manutenzione evolutiva e correttiva del software indipendentemente dalla tecnologia utilizzata per la realizzazione.

Il processo relativo alla FPA è:

- sufficientemente semplice da minimizzare lo sforzo aggiuntivo del processo di misurazione;
- una misura consistente tra differenti progetti e organizzazioni.

Al fine di applicare correttamente la presente norma internazionale, è possibile ricevere formazione ufficiale all'uso del metodo attraverso materiale di formazione certificato dall'IFPUG.

Il presente documento è parte delle pubblicazioni dell'IFPUG. Si raccomanda di leggere questo documento insieme alle altre pubblicazioni IFPUG. Tali pubblicazioni forniscono una guida all'applicazione delle regole descritte nella presente norma internazionale e ulteriori informazioni che aiutino a comprendere come usare ed applicare la dimensione funzionale risultante. Tra le diverse pubblicazioni di supporto dell'IFPUG si ricorda:

- la versione corrente del Manuale di Conteggio dell'IFPUG (CPM, Counting Practices Manual), che incorpora il presente standard, integrando pratiche di conteggio ed esempi che supportano tali implementazioni;
- “Framework for Functional Sizing” (2003), che discute il contributo sia della dimensione funzionale che di quella non-funzionale alla dimensione complessiva del prodotto software; il metodo FSM IFPUG è un metodo per misurare la dimensione funzionale;
- sito web IFPUG: [www.ifpug.org](http://www.ifpug.org).

## 1 Ambito di applicazione

### 1.1 Obiettivo

La presente norma internazionale specifica l'insieme delle definizioni, regole e passi per applicare il metodo IFPUG di misurazione della dimensione funzionale (FSM).

### 1.2 Conformità

La presente norma internazionale è conforme a tutte le disposizioni obbligatorie contenute nella norma ISO/IEC 14143-1:2007.

### 1.3 Applicabilità

La presente norma internazionale è applicabile a tutti i domini funzionali.

NOTA L'IFPUG continua a produrre nel tempo pubblicazioni relative all'uso del metodo in nuovi ambienti e domini applicativi.

La presente norma internazionale è completamente convertibile alle precedenti edizioni del metodo di dimensionamento dell'IFPUG.

Gli analisti che applicano gli IFPUG function point hanno rilevato differenti tassi dei tempi di rilascio (ore per il rilascio / function point) relativi alla realizzazione di applicazioni in differenti domini funzionali calibrati al variare delle dimensioni dei progetti e delle complessità dei software.

### 1.4 Platea

La presente norma internazionale può essere usata da chiunque abbia necessità di una misura della dimensione funzionale. Anche per chi abbia già esperienza con il metodo, questo documento può rappresentare un utile riferimento.

## 2 Riferimenti normativi

I seguenti documenti referenziati sono indispensabili per l'applicazione di questo documento. Per i riferimenti datati, si considera solo l'edizione citata. Per riferimenti non datati, si considera l'ultima edizione del documento referenziato (incluse tutte le modifiche).

ISO/IEC 14143-1:2007, *Information technology — Software measurement — Functional size measurement — Part 1: Definition of concepts*

## 3 Termini e definizioni

### 3.1

#### **manutenzione adeguativa (adaptive maintenance)**

modifica di un prodotto software, eseguita dopo il rilascio, per mantenere fruibile un programma in un ambiente mutato o in corso di cambiamento

NOTA La manutenzione adeguativa comprende le modifiche necessarie ad adattare i cambiamenti nell'ambiente nel quale un prodotto software deve operare. Tali cambiamenti sono quelli che devono essere effettuati per essere allineati con il nuovo ambiente. Ad esempio il sistema operativo potrebbe essere aggiornato ed alcune modifiche possono essere effettuate per adeguare il nuovo sistema operativo.

[ISO/IEC 14764:2006 Software engineering – Software life cycle processes – Maintenance, definizione 3.1]

### 3.2

#### **applicazione (application)**

collezione integrata di procedure automatizzate e dati che forniscono supporto ad un obiettivo organizzativo; è formata da uno o più componenti, moduli o sottosistemi

ESEMPI          conto fornitori, conto clienti, retribuzioni, approvvigionamento, produzione, controllo della catena di montaggio, radar di avvistamento aereo, tracciamento bersagli, azionamento armi, pianificazione linea di volo e prenotazioni passeggeri.

### 3.3

#### **dimensione funzionale dell'applicazione (application functional size)**

misura della funzionalità che un'applicazione fornisce all'utente, determinata dall'applicazione del conteggio dei function point

### 3.4

#### **conteggio in function point dell'applicazione (application function point count)**

attività di applicazione della presente norma internazionale per misurare la dimensione funzionale di un'applicazione

### 3.5

#### **riorganizzazione (arranging)**

attività di disposizione in sequenza degli attributi in una funzione di tipo transazionale

### 3.6

#### **tipo di entità associativa (associative entity type)**

tipo di entità con attributi che forniscono ulteriori dettagli su una relazione multi-a-molti tra altri due tipi di entità

### 3.7

#### **tipo di entità attributiva (attributive entity type)**

tipo di entità che fornisce dettagli su uno o più attributi di un altro tipo di entità

### 3.8

#### **componente funzionale di base (base functional component)**

##### **BFC**

unità elementare di Requisiti Utente Funzionali (FUR) definita e usata da un metodo FSM per obiettivi di misurazione

ESEMPI          Un possibile FUR relativo alla "Gestione dei Clienti" può essere composto dalle seguenti BFC: "Aggiungere un nuovo cliente", "Effettuare un report degli acquisti dei Clienti" e "Modificare i dati dei Clienti". Un altro esempio può includere una raccolta di dati di business logicamente correlati, mantenuti dal software in esame quali "Dettagli del Cliente". Molti altri esempi possono essere formulati.

[ISO/IEC 14143-1:2007, definizione 3.1]

### 3.9

#### **confine (boundary)**

interfaccia concettuale tra il software in esame e i suoi utenti

[ISO/IEC 14143-1:2007, definizione 3.3]

NOTA          Le precedenti versioni della norma internazionale usavano il termine "Confine dell'applicazione".

### 3.10

#### **stato di coerenza funzionale (consistent state)**

momento nel quale l'elaborazione è stata completamente eseguita; il FUR è stato soddisfatto e nessuna ulteriore azione è richiesta

ESEMPIO 1          Il FUR è quello di stampare disegni e segnare l'appropriato importo come pagamento. Se solo parte del Requisito funzionale dell'Utente si è completato (es., solo stampare l'assegno o solo segnare il pagamento), l'applicazione non è in uno stato di coerenza funzionale. La stampa dell'assegno senza segnare l'importo come pagamento causa un'incoerenza nell'applicazione come viceversa mettere l'importo senza stampare l'assegno.



**ESEMPIO 2** Il Requisito funzionale dell'Utente è di avere un'elaborazione batch che accetta in input un file per aggiornare un archivio, crea un report di controllo della produzione e ritorna un report degli errori rimandato all'applicazione chiamante. L'elaborazione non è in uno stato di coerenza funzionale fino a che tutte le parti non si sono completate.

**ESEMPIO 3** Il Requisito funzionale dell'Utente è trasferire un dipendente ad un nuovo impiego e di validare il/la suo/sua autorizzazione sul livello di sicurezza. Per completare questo, una richiesta in tempo reale è mandata all'applicazione Sicurezza (che mantiene il governo dell'autorizzazione sul livello di sicurezza e non sulla sicurezza) che riceve una risposta prima che il trasferimento possa essere completato. Tutti i passi sono creati per ottenere uno stato di consistenza. L'interazione con l'applicazione sicurezza non è un passo o azione indipendente. Non può avvenire all'interno e da sola, e la transazione per trasferire un dipendente non è in uno stato di coerenza funzionale senza di essa.

### 3.11

#### **informazioni di controllo (control information)**

dati che influenzano un processo elementare e specificano cosa, quando o come i dati devono essere elaborati

### 3.12

#### **funzionalità di conversione (conversion functionality)**

funzioni di tipo dati o transazionali fornite per convertire dati e/o soddisfare altri requisiti di conversione richiesti dall'utente

**NOTA** La funzionalità di conversione esiste esclusivamente durante lo sviluppo o la manutenzione evolutiva di una applicazione.

### 3.13

#### **manutenzione correttiva (corrective maintenance)**

modifica reattiva ad un prodotto software effettuata dopo il rilascio al fine di correggerne le problematiche rilevate

**NOTA** La modifica ripara il prodotto software per poter soddisfare i requisiti.

[ISO/IEC 14764:2006 Software engineering – Software life cycle processes – Maintenance, definizione 3.2]

### 3.14

#### **ambito del conteggio (counting scope)**

insieme di FUR da includere nel conteggio dei function point

### 3.15

#### **Tipo di Elemento Dati (Data Element Type)**

##### **DET**

attributo unico, riconoscibile dall'utente e non ripetuto

### 3.16

#### **funzione di tipo dati (data function)**

funzionalità fornita all'utente per soddisfare requisiti di memorizzazione di dati interni od esterni

**NOTA** Una funzione di tipo dati può essere rappresentata sia con un ILF che con un EIF.

### 3.17

#### **dati derivati (derived data)**

dati risultanti da un processo che comporta passi differenti da, o in aggiunta a, reperimento diretto e validazione di informazioni da funzioni di tipo dati

### 3.18

#### **progetto di sviluppo (development project)**

progetto con l'obiettivo di sviluppare e rilasciare la prima versione di un'applicazione software

### 3.19

#### **dimensione funzionale di un progetto di sviluppo (development project functional size)**

misura delle funzionalità fornite agli utenti con il primo rilascio del software, misurate secondo le regole del conteggio in function point di un progetto di sviluppo (DFP).

**NOTA** La dimensione funzionale di un progetto di sviluppo può includere la dimensione di funzionalità di conversione.

**3.20****conteggio in function point di un progetto di sviluppo (development project function point count)**

applicazione della norma internazionale per la misurazione della dimensione funzionale di un progetto di sviluppo

**3.21****processo elementare (elementary process)**

la più piccola unità di attività significativa per l'utente

**3.22****progetto di manutenzione evolutiva (enhancement project)**

progetto con l'obiettivo di sviluppare e rilasciare manutenzione adeguativa

NOTA Un progetto di manutenzione evolutiva può anche sviluppare e rilasciare manutenzioni di tipo correttivo e perfezionativo, ma queste non forniscono un contributo alla dimensione funzionale del progetto di manutenzione evolutiva.

**3.23****dimensione funzionale di un progetto di manutenzione evolutiva (enhancement project functional size)**

misura delle funzionalità aggiunte, modificate o cancellate al termine di un progetto di manutenzione evolutiva, misurato contando i function Point del progetto di manutenzione evolutiva

NOTA La dimensione funzionale di un progetto di manutenzione evolutiva può includere la dimensione delle funzioni di conversione.

**3.24****conteggio in function point di un progetto di manutenzione evolutiva (enhancement project function point count)**

applicazione della norma internazionale per la misurazione della dimensione funzionale di un progetto di manutenzione evolutiva

**3.25****entità dipendente (entity dependent)**

entità non significativa o non importante per il business in sé e per sé senza la presenza di altre entità, tale che

- un'occorrenza dell'entità X deve essere collegata ad un'occorrenza dell'entità Y, e
- dall'eliminazione di un'occorrenza nell'entità Y consegue l'eliminazione di tutte le occorrenze correlate nell'entità X.

**3.26****entità indipendente (entity independent)**

entità significativa o importante per il business in sé e per sé senza la presenza di altre entità

**3.27****Input Esterno (External Input)****EI**

processo elementare che elabora dati o informazioni di controllo che provengono dall'esterno del confine dell'applicazione

NOTA Un EI rappresenta uno dei possibili componenti funzionali di base (BFC).

**3.28****Interrogazione Esterna (External Inquiry)****EQ**

processo elementare che invia dati o informazioni di controllo all'esterno del confine dell'applicazione

NOTA Un'EQ rappresenta uno dei possibili componenti funzionali di base (BFC).

**3.29****File d'Interfaccia Esterno (External Interface File)****EIF**

gruppo di dati logicamente collegati o di informazioni di controllo riconoscibili dall'utente, referenziati dall'applicazione, ma mantenuti all'interno del confine di un'altra applicazione

NOTA Un EIF rappresenta uno dei possibili componenti funzionali di base (BFC).

**3.30****Output Esterno (External Output)****EO**

processo elementare che invia dati o informazioni di controllo all'esterno del confine dell'applicazione e include una logica elaborativa ulteriore oltre a quella prevista per un External Inquiry (EQ)

NOTA Un EO rappresenta uno dei componenti funzionali di base (BFC).

**3.31****Tipo di File Referenziato (File Type Referenced)****FTR**

funzione di tipo dati letta e/o mantenuta da una funzione di tipo transazionale

**3.32****complessità funzionale (functional complexity)**

valutazione della complessità specifica assegnata ad una funzione che usa le regole definite nella presente norma internazionale

**3.33****dimensione funzionale (functional size)**

dimensione del software derivata dalla quantificazione dei Requisiti Utente Funzionali (FUR)

[ISO/IEC 14143-1:2007, definizione 3.6]

**3.34****Requisiti Utente Funzionale (Functional User Requirements)****FUR**

sotto-insieme dei requisiti utente che descrivono cosa il software dovrà fare, in termini di attività e servizi

NOTA I FUR includono ma non sono limitati a:

- trasferimento di dati (es: inserimento dei dati cliente, invio di segnali di controllo);
- trasformazione di dati (es:: calcolo degli interessi bancario, calcolo della temperatura media);
- memorizzazione di dati (es:: memorizzazione di ordini cliente, registrazione del trend delle temperatura esterne);
- reperimento di dati (es:: produrre la lista degli impiegati, determinazione delle coordinate GPS di un aereo).

Requisiti utente che non sono FUR includono ma non sono limitati a:

- vincoli di qualità (es:: usabilità, affidabilità, efficienza e portabilità);
- vincoli organizzativi (es:: aspetti logistici, requisiti hardware e conformità agli standard);
- vincoli ambientali (es: interoperabilità, misure legate alla sicurezza logica e fisica e/o alla privacy);
- vincoli di implementazione (es:: linguaggi di sviluppo, piani di rilascio).

[ISO/IEC 14143-1:2007, definizione 3.8]

**3.35****Function Point****FP**

unità di misura per la dimensione funzionale così come definita nella presente norma internazionale

**3.36****Analisi dei Function Point (Function Point Analysis)****FPA**

metodo per la misurazione della dimensione funzionale così come definito nella presente norma internazionale

**3.37****conteggio di function point (function point count)**

attività relativa all'applicazione delle regole della presente norma internazionale per la misurazione della dimensione funzionale di un'applicazione o progetto

NOTA Esistono tre tipi di conteggio dei function point: applicazione (AFP), progetto di sviluppo (DFP) e progetto di manutenzione evolutiva (EFP).

**3.38****tipo di funzione (function type)**

cinque tipi di componenti funzionali base (BFC) identificati nella presente norma internazionale

NOTA I cinque tipi di funzione sono: External Input (EI), External Output (EO), External Inquiry (EQ), Internal Logical File (ILF) e External Interface File (EIF).

**3.39****File Logico Interno (Internal Logical File)****ILF**

gruppo di dati logicamente collegati o di informazioni di controllo, riconoscibili dall'utente, mantenuti all'interno del confine dell'applicazione oggetto della misurazione

NOTA Un ILF è un componente funzionale di base (BFC).

**3.40****mantenere (maintain)**

capacità di aggiungere, modificare o cancellare dati attraverso un processo elementare

**3.41****significativo (meaningful)**

riconoscibile dall'utente e che soddisfa un Requisito Utente Funzionale (FUR)

**3.42****manutenzione perfettiva (perfective maintenance)**

modifica di un prodotto software effettuata dopo il rilascio al fine di individuare e correggere errori latenti nel prodotto software prima che degenerino in malfunzionamenti

NOTA 1 La manutenzione perfettiva fornisce migliorie per gli utenti, miglioramenti della documentazione dei programmi, e una modifica del codice per migliorare le prestazioni del software, la sua manutenibilità, o altri possibili attributi.

NOTA 2 Contrapposta a: manutenzione adeguativa; manutenzione correttiva.

[ISO/IEC 14764:2006 Software engineering – Software life cycle processes – Maintenance, definizione 3.7]

**3.43****intento primario (primary intent)**

intento principale per importanza

**3.44****trattamento logico (logica elaborativa, processing logic)**

Ciascun requisito specificamente richiesto dall'utente per completare un processo elementare così come validazioni, algoritmi o calcolo e lettura o mantenere una funzionale di tipo dati

**3.45****scopo del conteggio (purpose of the count)**

motivazione per eseguire un conteggio in function point

NOTA Vedere sezione 5.3 a).

**3.46****Tipo di Elemento Record (Record Element Type)****RET**

sottogruppo, riconoscibile dall'utente, di tipi di elemento dati all'interno di una funzione di tipo dati

**3.47****autonomo (autosufficiente, self-contained)**

nessuna attività elaborativa precedente o successiva è necessaria per iniziare o completare il/i FUR

ESEMPIO Il FUR richiede che si possano sia inserire che aggiornare i dati di un impiegato. Ci possono essere diverse parti che completano le informazioni di un impiegato. Queste possono essere rappresentate in diverse vedute, finestre o schede quali:

- Identificazione dell'impiegato,
- Sede di lavoro dell'impiegato,
- Informazioni sul dipendente,
- Informazioni sullo stipendio, e
- Istruzione.

Per inserire i dati di un impiegato, sarà necessario compilare una o più schede, secondo le regole applicative definite. Il processo di 'inserimento' non è autonomo (autosufficiente) fino a che tutte le informazioni obbligatorie non siano state inserite.

Per aggiornare i dati di un impiegato, potranno essere modificate una o più schede in un dato momento, ma tutti i passi del processo coprono i FUR che sono quelli di aggiornare un impiegato. L'inserimento, modifica o cancellazione di informazioni su una singola scheda non rappresenta un processo elementare separato. Esso rappresenta solo uno passo del processo relativo all'aggiornamento dei dati di un impiegato. Anche se è possibile inserire informazioni aggiuntive nel record dell'impiegato, tutte le informazioni congiuntamente vanno considerate facenti parte di un singolo processo elementare: aggiornare i dati di un impiegato.

'Inserire i dati di un Impiegato' e 'Aggiornare i dati di un Impiegato' rappresentano ciascuno un processo autonomo.

**3.48****ordinamento (sorting)**

attività di disposizione in sequenza di righe o record in una funzione di tipo transazionale

**3.49****funzione di tipo transazionale (transactional function)**

processo elementare che fornisce funzionalità all'utente per processare dati

NOTA Una funzione di tipo transazionale può essere rappresentata tramite un External Input (EI), External Output (EO) o un External Inquiry (EQ).

**3.50****utente (user)**

ogni persona o oggetto che comunica o interagisce con il software in ogni istante

ESEMPI        Esempi di 'oggetto' includono, ma non sono limitati a: applicazioni software, animali, sensori, o altro hardware.

[ISO/IEC 14143-1:2007, definizione 3.11]

### 3.51

#### **riconoscibile dall'utente (user recognizable)**

requisiti per i processi e/o dati concordati, e compresi sia da utente(i) che sviluppatore(i) software

### 3.52

#### **punto di vista dell'utente (user view)**

Requisito Utente Funzionale (FUR) così come percepito dall'utente stesso

NOTA        Gli sviluppatori traducono il punto di vista dell'utente in software al fine di fornire una soluzione.

## 4 Acronimi

**BFC**        Base Functional Component (Componente Funzionale di Base)

**DET**        Data Element Type (Tipo di Elemento Dati)

**EI**         External Input (Input Esterno)

**EIF**        External Interface File (File d'Interfaccia Esterno)

**EO**         External Output (Output Esterno)

**EQ**         External Inquiry (Interrogazione Esterna)

**FP**         Function Point

**FPA**        Function Point Analysis (Analisi dei Function Point)

**FTR**        File Type Referenced (Tipo di File Referenziato)

**ILF**        Internal Logical File (File Logico Interno)

**RET**        Record Element Type (Tipo di Elemento Record)

## 5 Processo di Misurazione

### 5.1 Panoramica

Per svolgere un conteggio di function point, si devono effettuare le seguenti attività per identificare e classificare le componenti funzionali di base (ILF, EIF, EI, EO, EQ):

- a) raccogliere la documentazione disponibile, come indicato in sezione 5.2,
- b) determinare l'ambito del conteggio e il confine e identificare i Requisiti Utente Funzionali, come indicato in sezione 5.3,
- c) misurare le funzioni di tipo dati, come indicato nelle sezioni 5.4, 5.6 e 5.7,

NOTA        Le funzionalità di tipo conversione (se applicabile) si misurano come indicato in sezione 5.6; le funzionalità di tipo manutenzione evolutiva (se applicabile) si misurano, come indicato in sezione 5.7.

- d) misurare le funzioni di tipo transazionale, come indicato nelle sezioni 5.5, 5.6 e 5.7,

NOTA        Le funzionalità di tipo conversione (se applicabile) si misurano come indicato in sezione 5.6; le funzionalità di tipo manutenzione evolutiva (se applicabile) si misurano, come indicato in sezione 5.7.

- e) calcolare la dimensione funzionale, come indicato in sezione 5.8,
- f) documentare il conteggio dei function point, come indicato in sezione 5.9, e
- g) presentare il risultato del conteggio dei function point, come indicato in sezione 5.10.

NOTA La Figura 1 fornisce una panoramica grafica del processo di conteggio dei function point.



**Figura 1 — Panoramica grafica del processo di conteggio dei function point**

## 5.2 Raccogliere la documentazione disponibile

La documentazione a supporto di un conteggio di function point deve descrivere le funzionalità fornite dal software o le funzionalità che sono impattate dal progetto software che è oggetto di misurazione.

Si deve ottenere documentazione adeguata per condurre il conteggio di function point o accesso a esperti in materia, che possono fornire informazioni aggiuntive per integrare qualsiasi lacuna nella documentazione.

NOTA Una documentazione appropriata può includere requisiti, modelli dati/oggetti, diagrammi delle classi, diagrammi di flusso dei dati, casi d'uso, descrizioni procedurali, layout dei report, layout delle schermate, manuali utente e altri manufatti di sviluppo del software.

## 5.3 Determinare l'ambito del conteggio e il confine e identificare i Requisiti Utente Funzionali

Per determinare l'ambito del conteggio e il confine di ogni applicazione e identificare i Requisiti Utente Funzionali, si devono effettuare le seguenti attività:

- a) identificare lo scopo del conteggio,

NOTA 1 Un conteggio di function point è svolto per dare risposta a un problema aziendale, ed è il problema aziendale che determina lo scopo.

NOTA 2 Lo scopo del conteggio determina l'ambito del conteggio.

ESEMPIO 1 Lo scopo del conteggio potrebbe essere determinare la dimensione di una particolare versione di un software.

ESEMPIO 2 Lo scopo del conteggio potrebbe essere determinare la dimensione di un'applicazione come contributo allo sforzo dell'organizzazione per determinare la dimensione del suo portafoglio software.

- b) identificare il tipo di conteggio, in base allo scopo, come uno tra:
  - 1) conteggio di function point di un progetto di sviluppo,
  - 2) conteggio di function point di un'applicazione,

- 3) conteggio di function point di un progetto di manutenzione evolutiva,
- c) determinare l'ambito del conteggio, in base allo scopo e al tipo del conteggio,
- d) determinare il confine di ogni applicazione inclusa nell'ambito del conteggio, in base al punto di vista dell'utente, non a considerazioni tecniche.

ESEMPIO 3 Se lo scopo è stimare il costo di una manutenzione evolutiva delle applicazioni Risorse Umane (RU) e Benefici, allora:

- il tipo di conteggio è conteggio di progetto di manutenzione evolutiva;
  - l'ambito include le funzioni di tipo transazionale e di tipo dati aggiunte, modificate o cancellate per entrambe le applicazioni RRU e Benefici, così come qualsiasi requisito di conversione per ciascuna applicazione;
  - il punto di vista dell'utente è che RU e Benefici sono aree funzionali distinte, e quindi esse sono applicazioni separate;
  - esiste un confine tra le applicazioni RU e Benefici, così come tra ciascuna applicazione e l'utente.
- e) i requisiti utente possono includere una miscela di requisiti sia funzionali che non-funzionali; si identifichi quali requisiti sono funzionali, e si escludano i requisiti non-funzionali.

## 5.4 Misurare le funzioni di tipo dati

### 5.4.1 Panoramica

Le funzionalità di tipo dati soddisfano i Requisiti Utente Funzionali per memorizzare e/o referenziare i dati. Si devono valutare tutte le funzionalità di tipo dati nell'ambito del conteggio per identificare ciascun gruppo logico di dati.

Per misurare le funzioni di tipo dati, si devono effettuare le seguenti attività:

- a) identificare e raggruppare tutti i dati logici in funzioni di tipo dati, come indicato in sezione 5.4.2,
- b) classificare ciascuna funzione di tipo dati come o un ILF o un EIF, come indicato in sezione 5.4.3,
- c) contare i DET per ciascuna funzione di tipo dati, come indicato in sezione 5.4.4,
- d) contare i RET per ciascuna funzione di tipo dati, come indicato in sezione 5.4.5,
- e) determinare la complessità funzionale di ciascuna funzione di tipo dati, come indicato in sezione 5.4.6, and
- f) determinare la dimensione funzionale di ciascuna funzione di tipo dati, come indicato in sezione 5.4.7.

### 5.4.2 Identificare e raggruppare tutti i dati logici in funzioni di tipo dati

NOTA 1 Le funzioni di tipo dati si identificano più facilmente utilizzando un modello logico dei dati; tuttavia, ciò non preclude l'utilizzo del processo di misurazione in contesti in cui siano impiegate tecniche alternative di modellazione dei dati o degli oggetti. Si utilizza la terminologia di modellazione dei dati per documentare le regole relative alle funzioni di tipo dati, ma si può applicare il medesimo approccio con altre tecniche.

Per identificare le funzioni di tipo dati, si devono effettuare le seguenti attività:

- a) identificare tutti i dati, o le informazioni di controllo, logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio,
- b) escludere le entità che non sono mantenute da alcuna applicazione,



c) raggruppare le entità che dipendono da altre entità,

NOTA 2 Entità che sono indipendenti da altre entità sono considerate gruppi logici di dati distinti.

d) escludere le seguenti entità, denotate come dati di decodifica:

- 1) entità dati di sostituzione, che contengono un codice e un nome esplicativo o descrizione;
- 2) entità a occorrenza singola, che contengono uno o più attributi che variano raramente, se non mai;
- 3) entità che contengono dati che sono sostanzialmente statici o che possono variare molto raramente;
- 4) entità con valori di default, che contengono valori per il popolamento di attributi;
- 5) entità di valori validi, che contengono valori disponibili per selezione o validazione;
- 6) entità che contengono intervalli di valori per validazione,

NOTA 3 Le entità sopra denotate come dati di decodifica possono anche contenere altri attributi per scopi di controllo e per definire date di validità. La presenza di tali attributi non modifica la natura di tali entità.

e) escludere entità che non contengono attributi richiesti dall'utente, e

f) rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna nelle entità principali.

NOTA 4 Attributi di chiave esterna sono dati richiesti dall'utente per stabilire una relazione con un'altra funzione di tipo dati.

### 5.4.3 Classificare ciascuna funzione di tipo dati come o un ILF o un EIF

Una funzione di tipo dati deve essere classificata come un:

- a) File Logico Interno (ILF) se è mantenuta dall'applicazione oggetto di misurazione, o
- b) File d'Interfaccia Esterno (EIF) se è
  - referenziata, ma non mantenuta, dall'applicazione oggetto di misurazione, e
  - identificata in un ILF di un'altra o altre applicazioni.

### 5.4.4 Contare i DET per ciascuna funzione di tipo dati

Per contare i Tipi di Elementi Dati (DET) per una funzione di tipo dati, si devono effettuare le seguenti attività:

- a) contare un DET per ciascun attributo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di tutti i processi elementari nell'ambito del conteggio,

NOTA 1 Per esempio, in un ILF o EIF, contare un solo DET per i 12 campi ripetuti Importo Budget Mensile. Contare un DET aggiuntivo per identificare il mese applicabile.

- b) contare solo i DET utilizzati dall'applicazione oggetto di misurazione, laddove due o più applicazioni mantengono e/o referenziano la stessa funzione di tipo dati,

NOTA 2 Gli attributi che non sono referenziati dall'applicazione oggetto di misurazione non sono contati.

- c) contare un DET per ogni attributo richiesto dall'utente per stabilire una relazione con un'altra funzione di tipo dati, e
- d) rivedere gli attributi collegati per determinare se sono raggruppati e contati come un solo DET o se sono contati come DET multipli. Il raggruppamento dipenderà da come i processi elementari utilizzano gli attributi nell'applicazione.

ESEMPIO      Gli attributi (nome, iniziale secondo nome, cognome) sono raggruppati e contati come:

- nominativo (nome, iniziale secondo nome, cognome) se sono sempre usati insieme,
- nome proprio (nome e iniziale secondo nome) e cognome se, in aggiunta, il cognome è talvolta usato a se stante, o
- nome proprio, iniziale secondo nome e cognome, se tutti e tre possono essere usati indipendentemente.

#### 5.4.5 Contare i RET per ciascuna funzione di tipo dati

Per contare i Tipi di Elementi Record (RET) per una funzione di tipo dati, si devono effettuare le seguenti attività:

- a) contare un RET per ciascuna funzione di tipo dati (cioè, per default, ciascuna funzione di tipo dati ha un sottogruppo di DET da contare come un RET),
- b) contare un RET aggiuntivo per ciascuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contenga più di un DET:
  - 1) entità associative con attributi non-chiave;
  - 2) entità sotto-tipo (diverso dal primo sotto-tipo);
  - 3) entità attributiva, in relazione diversa da quella 1-1 obbligatoria.

NOTA 1      Una relazione 1-1 obbligatoria riflette una relazione tra due entità in cui ogni istanza è collegata a una e una sola istanza dell'entità collegata.

NOTA 2      Se non si dispone di un modello dei dati, si ricerchino i gruppi ripetuti di dati per identificare i RET.

ESEMPIO 1      Una fattura ha una sezione di testata con le informazioni del cliente e delle linee con le informazioni degli acquisiti (per es. numero dell'oggetto, descrizione, prezzo, peso). La sezione Testata è contata come un RET. Le linee sono un gruppo ripetuto e sono contate come un RET ulteriore.

ESEMPIO 2      Un singolo attributo ripetuto, come più numeri di conto di un cliente, non è un gruppo ripetuto ed è contato come un singolo DET, non come un RET.

ESEMPIO 3      Un gruppo con occorrenze multiple di DET, come anno, mese e import del budget, è un gruppo ripetuto ma è contato come tre DET, non come un RET.

#### 5.4.6 Determinare la complessità funzionale di ciascuna funzione di tipo dati

La complessità funzionale di ciascuna funzione di tipo dati è determinata utilizzando il numero di RET e DET secondo la Tabella 1.

**Tabella 1 — Complessità delle funzioni di tipo dati.**

		DET		
		1 – 19	20 – 50	> 50
RET	1	Bassa	Bassa	Media
	2 – 5	Bassa	Media	Alta
	> 5	Media	Alta	Alta

#### 5.4.7 Determinare la dimensione funzionale di ciascuna funzione di tipo dati

La dimensione funzionale di ciascuna funzione di tipo dati è determinata utilizzando il tipo e la complessità funzionale secondo la Tabella 2.

**Tabella 2 — Dimensione delle funzioni di tipo dati.**

		Tipo	
		ILF	EIF
Complessità funzionale	Bassa	7	5
	Media	10	7
	Alta	15	10

### 5.5 Misurare le funzioni di tipo transazionale

#### 5.5.1 Panoramica

Le funzionalità di tipo transazionale soddisfano i Requisiti Utente Funzionali sull'elaborazione dei dati. Si devono valutare tutte le funzionalità di tipo transazionale nell'ambito del conteggio per identificare ciascun processo elementare unico.

Per misurare le funzioni di tipo transazionale, si devono effettuare le seguenti attività:

- identificare ciascun processo elementare richiesto dall'utente, come indicato in sezione 5.5.2,
- classificare ciascuna funzione di tipo transazionale come un External Input (EI), un External Output (EO) o un'External Inquiry (EQ), come indicato in sezione 5.5.3,
- contare i Tipi di File Referenziati (FTR) per ciascuna funzione di tipo transazionale, come indicato in sezione 5.5.4,
- contare i Tipi di Elementi Dati (DET) per ciascuna funzione di tipo transazionale, come indicato in sezione 5.5.5,
- determinare la complessità funzionale di ciascuna funzione di tipo transazionale, come indicato in sezione 5.5.6, and

- f) determinare la dimensione funzionale di ciascuna funzione di tipo transazionale, come indicato in sezione 5.5.7.

## 5.5.2 Identificare i processi elementari

### 5.5.2.1

Per identificare ciascun processo elementare, si devono effettuare le seguenti attività:

- a) Comporre e/o scomporre i Requisiti Utente Funzionali nella più piccola unità di attività, che soddisfa tutti i seguenti criteri:

- 1) è significativa per l'utente;
- 2) costituisce una transazione completa;

NOTA Per utilizzatori di precedenti versioni di questo Standard Internazionale, il punto 2) non è una modifica, ma piuttosto un raffinamento per aumentare la specificità per promuovere un'interpretazione coerente.

- 3) è autonoma;
- 4) lascia la totalità dell'applicazione oggetto di conteggio in uno stato di coerenza funzionale.

ESEMPIO 1 Un Requisito Utente Funzionale può richiedere che sia fornita una funzione per Mantenere le informazioni dell'Impiegato. Questo requisito è scomposto in unità di lavoro più piccolo, quali Aggiungi Impiegato, Modifica Impiegato, Elimina Impiegato, Interroga Impiegato.

ESEMPIO 2 Singoli requisiti possono affermare la necessità di aggiungere differenti tipi di informazioni dell'impiegato (per es. l'indirizzo, il salario e le informazioni sui dipendenti), ma la più piccola unità di attività significativa per l'utente è Aggiungi Impiegato.

- b) Identificare un processo elementare per ciascuna unità di attività identificata che soddisfa tutti i criteri della sezione 5.5.2.1 a).

### 5.5.2.2

Per determinare processi elementari unici, si devono effettuare le seguenti attività:

- a) Quando confrontato con un processo elementare già identificato, contare due processi elementari simili come il medesimo processo elementare se essi:

- 1) richiedono lo stesso insieme di DET;
- 2) richiedono lo stesso insieme di FTR;
- 3) richiedono lo stesso insieme di trattamento logico per completare il processo elementare (v. lista che segue la sezione 5.5.2.2 b),

NOTA 1 Le attività per misurare le funzioni di tipo transazionale sono descritte in modo sequenziale; tuttavia, in realtà esse sono iterative. Gli FTR e i DET sono identificati come indicato nelle sezioni 5.5.4 e 5.5.5, ma sono necessari per confrontare due processi elementari simili.

NOTA 2 Un processo elementare può includere variazioni minori nei DET o negli FTR, così come molteplici alternative, variazioni o occorrenze del trattamento logico seguente.

ESEMPIO Quando si è identificato un processo elementare per Aggiungere un Impiegato, non lo si divide in due processi elementari per considerare il fatto che l'impiegato può avere o non avere dei dipendenti. Il processo elementare è sempre Aggiungi Impiegato, e si ha una variazione nel trattamento logico e nei DET da considerare per i dipendenti.

- b) Non scindere un processo elementare con molteplici forme di trattamento logico in molteplici processi elementari.

NOTA 3 Un processo elementare che accetta e valida dati da parte dell'utente, legge e filtra dati provenienti da un ILF e ordina e presenta i risultati di nuovo all'utente non può essere scisso in molteplici processi elementari.

### 5.5.2.3

Varie forme di trattamento logico per completare un processo elementare sono identificate in Tab. 3:

**Tabella 3 — Forme di trattamento logico.**

Forme di trattamento logico	
1. Eseguire validazioni.	
ESEMPIO 1	Quando si aggiunge un nuovo impiegato all'organizzazione, il processo valida il DET "tipo impiegato".
2. Eseguire formule matematiche e calcoli.	
ESEMPIO 2	Quando si ottiene il report di tutti gli impiegati dell'organizzazione, il processo include il calcolo dei numeri totali di impiegati stipendiati, impiegati pagati a ore e tutti gli impiegati.
3. Convertire valori equivalenti.	
ESEMPIO 3	L'età dell'impiegato è convertita in un intervallo di età utilizzando una tabella.
4. Filtrare e selezionare dati utilizzando criteri specifici per confrontare molteplici insiemi di dati.	
ESEMPIO 4	Per generare una lista di impiegati per assegnazione, un processo elementare confronta il numero lavoro e l'assegnazione lavoro per selezionare ed elencare gli impiegati corretti con tale assegnazione.
5. Analizzare condizioni per determinare quali siano applicabili.	
ESEMPIO 5	Il trattamento logico utilizzato dal processo elementare che aggiunge un nuovo impiegato dipenderà dal fatto che l'impiegato sia pagato con uno stipendio o in base alle ore lavorate. L'input di DET (e il trattamento logico risultante) basati sulla diversa scelta (stipendiato o a ore) in questo esempio è parte di un unico processo elementare.
6. Aggiornare uno o più ILF.	
ESEMPIO 6	Quando si aggiunge un nuovo impiegato, il processo elementare aggiorna l'ILF Impiegato per mantenere i dati dell'impiegato.
7. Referenziare uno o più ILF o EIF.	
ESEMPIO 7	Quando si aggiunge un nuovo impiegato, si riferenzia l'EIF Valuta per il corretto tasso di conversione in dollari USA per determinare la paga oraria dell'impiegato.
8. Reperire dati o informazioni di controllo.	
ESEMPIO 8	Per visualizzare una lista di impiegati, si reperiscono le informazioni degli impiegati da una funzione di tipo dati.
9. Creare dati derivati trasformando dati esistenti per creare ulteriori dati.	
ESEMPIO 9	Per determinare (derivare) un numero di registrazione di un paziente (per es. SMIJO01), si concatenano i seguenti dati:
—	le prime tre lettere del cognome del paziente (per es. SMI per Smith);
—	le prime due lettere del nome del paziente (per es. JO per John);
—	un progressivo unico a due cifre (a partire da 01).

Forme di trattamento logico	
10. Alterare il comportamento del sistema.	ESEMPIO 10 Il comportamento del processo elementare di pagamento degli impiegati è alterato quando si apporta una modifica alla data di pagamento, da "ogni quindicesimo e ogni ultimo giorno del mese" a "ogni due venerdì", risultando in 26 periodi di paga all'anno invece di 24.
11. Preparare e presentare informazioni all'esterno del confine.	ESEMPIO 11 Una lista di impiegati è formattata e visualizzata all'utente.
12. Capacità di accettare dati o informazioni di controllo che entrano nel confine dell'applicazione.	ESEMPIO 12 Un utente inserisce varie informazioni per aggiungere un ordine del cliente nell'applicazione.
13. Ordinare o organizzare un insieme di dati. Questa forma di trattamento logico non ha impatto sull'identificazione del tipo, né contribuisce all'unicità di un processo elementare; cioè, la disposizione dei dati non costituisce elemento di unicità.	ESEMPIO 13 La lista degli impiegati è ordinata o in ordine alfabetico o per sede di appartenenza.  ESEMPIO 14 In una schermata di inserimento ordini, le informazioni di testata dell'ordine sono sistemate in cima alla schermata, mentre i dettagli dell'ordine sono situati nella parte sottostante.  NOTA 1 Precedenti versioni di questo Standard Internazionale (ISO/IEC 20926:2003) travisavano i termini "ri-ordinare" e "ri-organizzare"; l'utilizzo dei termini "ordinare" e "organizzare" è una correzione e non una modifica.

### 5.5.3 Classificare ciascun processo elementare come una funzione di tipo transazionale

#### 5.5.3.1

Per ciascun processo elementare:

- a) si identifica l'intento primario come uno dei seguenti:
  - 1) alterare il comportamento dell'applicazione,
  - 2) mantenere uno o più ILF,
  - 3) presentare informazioni all'utente;
- b) si identificano le forme di trattamento logico richieste per completare il processo elementare dalla lista presentata nella sezione 5.5.2.3.

#### 5.5.3.2

Ciascun processo elementare deve essere classificato come:

- a) un EI, se:
  - 1) include il trattamento logico di accettare dati o informazioni di controllo che entrano nel confine dell'applicazione;
  - 2) ha l'intento primario di:
    - i) o mantenere uno o più ILF,
    - ii) o alterare il comportamento dell'applicazione;

- b) un EO, se ha l'intento primario di presentare informazioni all'utente e include almeno una delle seguenti forme di trattamento logico:
- 1) eseguire calcoli matematici;
  - 2) aggiornare uno o più ILF;
  - 3) creare dati derivati;
  - 4) alterare il comportamento dell'applicazione;
- c) un'EQ, se ha l'intento primario di presentare informazioni all'utente e:
- 1) riferisce una funzione di tipo dati per reperire dati o informazioni di controllo;
  - 2) non soddisfa i criteri per essere classificato come un EO.

NOTA 1 La Tabella 4 presenta un sommario della relazione tra intento primario e tipo di funzione transazionale.

**Tabella 4 — Relazione tra intento primario e tipo di funzione transazionale.**

Funzione	Tipo di funzione transazionale		
	EI	EO	EQ
Alterare il comportamento dell'applicazione	IP	E	N
Mantenere uno o più ILF	IP	E	N
Presentare informazioni a un utente	E	IP	IP

legenda

- IP l'intento primario del tipo di funzione transazionale  
 E una funzione del tipo di funzione transazionale che è eventualmente presente, ma che non è l'intento primario  
 N il tipo di funzione transazionale non può svolgere questo tipo di funzione

NOTA 2 La Tabella 5 presenta un sommario della relazione tra forme di trattamento logico e tipo di funzione transazionale.

**Tabella 5 — Relazione tra trattamento logico e tipo di funzione transazionale.**

Forma di trattamento logico	Tipo di funzione transazionale		
	EI	EO	EQ
1. Eseguire validazioni	p	p	p
2. Eseguire calcoli matematici	p	o*	n
3. Convertire valori equivalenti	p	p	p
4. Filtrare e selezionare dati utilizzando criteri specifici per confrontare molteplici insiemi di dati	p	p	p
5. Analizzare condizioni per determinare quali siano applicabili	p	p	p
6. Aggiornare almeno un ILF	o*	o*	n
7. Riferire almeno un ILF o EIF	p	p	o
8. Reperire dati o informazioni di controllo	p	p	o
9. Creare dati derivati	p	o*	n
10. Alterare il comportamento del sistema	o*	o*	n
11. Preparare e quindi presentare informazioni all'esterno del confine	p	o	o
12. Accettare dati o informazioni di controllo che entrano nel confine dell'applicazione	o	p	p
13. Ordinare o riorganizzare un insieme di dati	p	p	p

## legenda

- o è obbligatorio che il tipo di funzione transazionale utilizzi la forma di trattamento logico
- o\* è obbligatorio che il tipo di funzione transazionale utilizzi almeno una di queste (o\*) forme di trattamento logico
- p il tipo di funzione transazionale può utilizzare la forma di trattamento logico, ma non è obbligatorio
- n il tipo di funzione transazionale non può utilizzare la forma di trattamento logico

**5.5.4 Contare gli FTR per ciascuna funzione di tipo transazionale**

Per ciascuna funzione di tipo transazionale, si conta un FTR per ciascuna funzione di tipo dati unica, che è acceduta (letta e/o scritta) dalla funzione di tipo transazionale.

NOTA Le attività per misurare le funzioni di tipo transazionale sono descritte in modo sequenziale; tuttavia, in realtà esse sono iterative. Per confrontare due processi elementari simili, è necessaria l'identificazione degli FTR come riferito nella sezione 5.5.2.2.

**5.5.5 Contare i DET per ciascuna funzione di tipo transazionale**

Per contare i DET per una funzione di tipo transazionale, si devono effettuare le seguenti attività:

- a) rivedere tutto ciò che attraversa (entra e/o esce attraverso) il confine,
- b) contare un DET per ciascun attributo unico, riconoscibile dall'utente, non ripetuto, che attraversa (entra e/o esce attraverso) il confine durante l'elaborazione della funzione di tipo transazionale.

ESEMPIO 1 I DET che attraversano il confine includono:

- attributi che l'utente inserisce tramite una schermata. Così come quelli visualizzati in un report o a video,
  - attributi, che entrano nel confine dell'applicazione, che sono richiesti per specificare quando, quali e/o come i dati debbano essere reperiti o generati dal processo elementare,
  - attributi forniti da, o presentati a, l'utente della funzione di tipo transazionale, e
  - attributi in un file elettronico che entrano o escono dal confine.
- c) contare un solo DET per funzione di tipo transazionale per la capacità di inviare un messaggio di risposta dell'applicazione, anche qualora vi siano molteplici messaggi,

ESEMPIO 2 Se si presentano all'utente molteplici messaggi di errore/conferma, si conta un solo DET.

- d) contare un solo DET per funzione di tipo transazionale per la capacità di avviare un'azione/i, anche qualora vi siano molteplici modi di farlo, e

ESEMPIO 3 Se l'utente può avviare la generazione di un report cliccando sul pulsante OK o premendo un tasto funzionale, si conta un solo DET.

- e) non contare i seguenti casi come DET:

- costanti come titoli di report, identificativi di schermata o di riquadro, intestazioni di colonna e titoli dei campi;
- notazioni generate dall'applicazione, come attributi di data e ora;
- variabili di impaginazione, numeri di pagina e informazioni di posizione; per es. 'Righe da 37 a 54 di 211';
- aiuti navigazionali, come la capacità di spostarsi in una lista utilizzando "precedente", "successivo", "primo", "ultimo" e i loro equivalenti grafici;



- attributi generati all'interbo del confine da una funzione di tipo transazionale e salvati in un ILF senza uscire dal confine;
- attributi reperiti o referenziati da un ILF o EIF per contribuire all'elaborazione senza uscire dal confine.

NOTA Le attività per misurare le funzioni di tipo transazionale sono descritte in modo sequenziale; tuttavia, in realtà esse sono iterative. Per confrontare due processi elementari simili, è necessaria l'identificazione dei DET come riferito nella sezione 5.5.2.2.

### 5.5.6 Determinare la complessità funzionale di ciascuna funzione di tipo transazionale

La complessità funzionale di ciascuna funzione di tipo transazionale è determinata utilizzando il numero di FTR e DET secondo la Tabella 6 o la Tabella 7.

**Tabella 6 — Complessità funzionale dell'EI.**

		DET		
		1 – 4	5 – 15	> 15
FTR	0 – 1	Bassa	Bassa	Media
	2	Bassa	Media	Alta
	> 2	Media	Alta	Alta

**Tabella 7 — Complessità funzionale dell'EO e dell'EQ.**

		DET		
		1 – 5	6 – 19	> 19
FTR	0 – 1	Bassa	Bassa	Media
	2 – 3	Bassa	Media	Alta
	> 3	Media	Alta	Alta
NOTA		Un'EQ ha minimo 1 FTR.		

### 5.5.7 Determinare la dimensione funzionale di ciascuna funzione di tipo transazionale

La dimensione funzionale di ciascuna funzione di tipo transazionale è determinata utilizzando il tipo e la complessità funzionale secondo la Tabella 8.

**Tabella 8 — Transactional function size.**

		Tipo		
		EI	EO	EQ
Complessità funzionale	Bassa	3	4	3
	Media	4	5	4
	Alta	6	7	6

## 5.6 Misurare le funzionalità di tipo conversione

L'ambito del conteggio di un progetto di sviluppo o di manutenzione evolutiva può includere anche la dimensione funzionale delle funzionalità di conversione richieste per il progetto. Le funzioni di tipo dati e di tipo transazionale di conversione (che non siano già state contate) vanno contate come indicato nelle sezioni 5.4 e 5.5.

NOTA L'ambito del conteggio stabilisce se le funzionalità di conversione vadano contate.

## 5.7 Misurare le funzionalità di tipo manutenzione evolutiva

I progetti di manutenzione evolutiva possono comportare aggiunte, modifiche o cancellazioni di funzionalità esistenti. Le funzionalità di manutenzione evolutiva vanno misurate secondo quanto segue:

- a) Non modificare il confine già stabilito per la/le applicazioni oggetto di modifica,
- b) Contare le funzioni di tipo dati che sono aggiunte, modificate o cancellate, come indicato in sezione 5.4,
- c) Contare le funzioni di tipo transazionale che sono aggiunte, modificate o cancellate, come indicato in sezione 5.5, and
- d) La dimensione funzionale dell'applicazione può essere aggiornata per riflettere:
  - 1) le funzionalità aggiunte, che aumentano la dimensione funzionale dell'applicazione;
  - 2) le funzionalità modificate, che possono aumentare, diminuire o lasciare inalterata la dimensione funzionale dell'applicazione;
  - 3) le funzionalità cancellate, che diminuiscono la dimensione funzionale dell'applicazione.

NOTA 1 Una modifica di una funzione di tipo dati può comportare aggiungere, modificare o cancellare DET e/o RET.

NOTA 2 Una modifica di una funzione di tipo transazionale può comportare aggiungere, modificare o cancellare DET e/o FTR e/o modificare il trattamento logico.

## 5.8 Calcolare la dimensione funzionale

Lo scopo e l'ambito del conteggio vanno considerati per selezionare e utilizzare la formula appropriate per calcolare la dimensione funzionale.

La dimensione funzionale di un progetto di sviluppo va calcolata utilizzando la Formula (1):

$$DFP = ADD + CFP \quad (1)$$

dove:

DFP è il conteggio di function point del progetto di sviluppo (Development);

ADD è la dimensione delle funzioni che devono essere fornite all'utente dal progetto di sviluppo;

CFP è la dimensione delle funzioni di conversione.

La dimensione funzionale di un'applicazione dalla misurazione dopo il progetto di sviluppo o in qualsiasi momento durante il ciclo di vita dell'applicazione va calcolata utilizzando la Formula (2):

$$AFP = ADD \quad (2)$$

dove:

AFP è il conteggio di function point dell'applicazione;

ADD è la dimensione delle funzioni che devono essere fornite all'utente dal progetto di sviluppo (escludendo la dimensione di qualsiasi funzionalità di conversione) o delle funzionalità che esistono in qualsiasi momento l'applicazione sia conteggiata.

La dimensione funzionale di un progetto di manutenzione evolutiva va calcolata utilizzando la Formula (3):

$$EFP = ADD + CHGA + CFP + DEL \quad (3)$$

dove:

EFP è il conteggio di function point del progetto di manutenzione evolutiva (Enhancement);

ADD è la dimensione delle funzioni aggiunte dal progetto di manutenzione evolutiva;

CHGA è la dimensione delle funzioni modificate dal progetto di manutenzione evolutiva – come sono/saranno dopo (After) l'implementazione;

CFP è la dimensione delle funzioni di conversione;

DEL è la dimensione delle funzioni cancellate dal progetto di manutenzione evolutiva.

La dimensione funzionale di un'applicazione dopo un progetto di manutenzione evolutiva va calcolata utilizzando la Formula (4):

$$AFPA = (AFPB + ADD + CHGA) - (CHGB + DEL) \quad (4)$$

dove:

AFPA è il conteggio di function point dell'applicazione dopo il progetto di manutenzione evolutiva;

AFPB è il conteggio di function point dell'applicazione prima del progetto di manutenzione evolutiva;

ADD è la dimensione delle funzioni aggiunte dal progetto di manutenzione evolutiva;

CHGA è la dimensione delle funzioni modificate dal progetto di manutenzione evolutiva – come sono/saranno dopo (After) l'implementazione;

CHGB è la dimensione delle funzioni modificate dal progetto di manutenzione evolutiva – come sono/erano prima (Before) che il progetto iniziasse;

DEL è la dimensione delle funzioni cancellate dal progetto di manutenzione evolutiva.

## 5.9 Documentare il conteggio dei function point

Il conteggio dei function point va documentato come segue:

- scopo e tipo del conteggio;
- ambito del conteggio e confine dell'applicazione;
- data del conteggio;
- elenco di tutte le funzioni di tipo dati e di tipo transazionale, inclusi per ciascuna il tipo, la complessità e il numero di function point assegnato;
- risultato del conteggio (v. sezione 5.10);
- qualsiasi ipotesi assunta e criticità risolta.

La documentazione del conteggio di function point può anche includere quanto segue:

- identificativo della documentazione di origine su cui si è basato il conteggio;

- identificativo dei partecipanti, i loro ruoli e qualifiche;
- per ciascuna funzione di tipo dati, i numeri di DET e RET;
- per ciascuna funzione di tipo transazionale, i numeri di DET e FTR;
- un riferimento incrociato tra tutte le funzioni di tipo dati e le funzioni di tipo transazionale;
- un riferimento incrociato tra tutte le funzioni di tipo dati e le astrazioni corrispondenti nella documentazione di origine;
- un riferimento incrociato tra tutte le funzioni di tipo transazionale e le astrazioni corrispondenti nella documentazione di origine.

NOTA 1 Si concordi il livello della documentazione con il cliente e lo si informi sui relative costi e benefici.

NOTA 2 Un conteggio di function point documentato in modo completo faciliterà la tracciabilità, l'usabilità e la manutenibilità; tuttavia, un cliente potrebbe essere interessato al solo risultato finale.

## 5.10 Presentare il risultato del conteggio dei function point

NOTA Presentare i risultati dei conteggi di function point in modo omogeneo consente ai lettori di identificare lo standard a cui essi si attengono.

### 5.10.1

I risultanti che si attengono a questo Standard Internazionale vanno riportati come segue:

S FP (IFPUG–IS)

dove:

S è il risultato del conteggio dei function point;

FP è l'unità di dimensione del metodo FSM IFPUG;

IS è questo Standard Internazionale (ISO/IEC 20926:2009).

ESEMPIO 250 FP (IFPUG-ISO/IEC 20926:2009)

### 5.10.2

I risultanti che si attengono a una personalizzazione locale di questo Standard Internazionale vanno riportati come:

S FP (IFPUG–IS–c)

dove:

c rappresenta uno o più caratteri che indicano che il risultato non si attiene in modo completo a questo Standard Internazionale.

ESEMPIO 250 FP (IFPUG–ISO/IEC 20926:2009–a)

## Allegato A (informativo)

### Tabelle unificate di complessità e dimensione funzionale

Per comodità, si riportano le tabelle di complessità e dimensione funzionale in Tabelle da A.1 a A.5.

**Tabella A.1 — Complessità delle funzioni di tipo dati.**

		DET		
		1 – 19	20 – 50	> 50
RET	1	Bassa	Bassa	Media
	2 – 5	Bassa	Media	Alta
	> 5	Media	Alta	Alta

**Tabella A.2 — Dimensione delle funzioni di tipo dati.**

		Tipo	
		ILF	EIF
Complessità funzionale	Bassa	7	5
	Media	10	7
	Alta	15	10

**Tabella A.3 — Complessità funzionale dell'EI.**

		DET		
		1 – 4	5 – 15	> 15
FTR	0 – 1	Bassa	Bassa	Media
	2	Bassa	Media	Alta
	> 2	Media	Alta	Alta

**Tabella A.4 — Complessità funzionale dell'EO e dell'EQ.**

		DET		
		1 – 5	6 – 19	> 19
FTR	0 – 1	Bassa	Bassa	Media
	2 – 3	Bassa	Media	Alta
	> 3	Media	Alta	Alta
NOTA		Un'EQ ha minimo 1 FTR.		

**Tabella A.5 — Dimensione delle funzioni di tipo transazionale.**

		Tipo		
		EI	EO	EQ
Complessità funzionale	Bassa	3	4	3
	Media	4	5	4
	Alta	6	7	6

Questa pagina è lasciata intenzionalmente vuota.

---

## **Parte 2 – Collegamento – Applicare il metodo FSM IFPUG**

---

---

Questa pagina è lasciata intenzionalmente vuota.

---



---

## Parte 2 Collegamento – Applicare il metodo FSM IFPUG

---

**Introduzione** La parte 1 descrive il processo di analisi dei function point per misurare funzionalmente il software seguendo il metodo IFPUG, inoltre descrive regole dettagliate per identificare e conteggiare le funzioni di tipo dati e le funzioni di tipo transazionale.

La parte 2 fornisce una panoramica del metodo IFPUG, insieme ad una guida per l'applicazione delle regole per determinare il tipo di conteggio, stabilire i confini applicativi e misurare le funzioni di tipo dati e le funzioni di tipo transazionale.

**Contenuti** La parte 2 comprende i seguenti capitoli:

<b>Argomento</b>	<b>Pagina</b>
Relazione tra IFPUG e ISO	1-2
Panoramica sul metodo FSM IFPUG	2-1
Raccogliere la documentazione disponibile	3-1
Determinare il tipo di conteggio	4-1
Determinare l'ambito del conteggio e i confini applicativi	5-1
Misurare le funzioni di tipo dati	6-1
Misurare le funzioni di tipo transazionale	7-1
Indice	i-1

## Relazione tra IFPUG e ISO

### Direzione strategica IFPUG

Il metodo dell'analisi dei function point IFPUG è uno standard ISO e deve essere conforme alla norma ISO/IEC 14143-1:2007. Il metodo può misurare soltanto la "dimensione funzionale" e non la "dimensione non-funzionale". Questo non significa che la dimensione non-funzionale non possa, o non debba, essere misurata, ma solo che deve essere chiaramente dichiarata come una misura separata. ("A Framework for Functional Sizing" [IFPUG, 2003]).

### "A Framework for Functional Sizing"

L'articolo "*A Framework for Functional Sizing*" fornisce le basi per comprendere il dimensionamento funzionale in relazione ai requisiti. Il documento esamina differenti tipi di requisiti; i concetti presenti forniscono le basi per dimensionare il software in conformità con la norma ISO/IEC 14143-1 ed il CPM IFPUG.

Il fondamento del documento "*A Framework for Functional Sizing*" è il fatto che ci possono essere molteplici metodi di misura per scopi differenti. La dimensione funzionale potrebbe essere misurata usando il metodo dell'analisi dei function point IFPUG, basato sui requisiti utente funzionali. Altre misure possono essere usate per dimensionare, per esempio, i requisiti non-funzionali.

Entrambi i risultati, in diverse unità di misura, rappresentano differenti misure della dimensione del software: function point IFPUG per la dimensione funzionale e qualsiasi altra misura per la dimensione non-funzionale. Queste dimensioni non possono essere sommate insieme perché rappresentano differenti misure (come il volume e la temperatura di una stanza), ma possono essere usate entrambe nella stima dell'impegno necessario per lo sviluppo di un'applicazione o di un sistema.

L'articolo "*A Framework for Functional Sizing*" fornisce una guida per distinguere fra la dimensione funzionale e quella non-funzionale.

## ISO/IEC 14143-1 – Definizione dei requisiti utente

Nel 1998, fu pubblicato il primo standard ISO/IEC sulla Misurazione della Dimensione Funzionale (ISO/IEC 14143-1:1998). Questo standard definisce la Dimensione Funzionale come 'una dimensione del software derivante dalla quantificazione dei Requisiti Utenti Funzionali'. Esso è stato aggiornato nel 2007 e pubblicato come ISO/IEC 14143-1:2007.

La norma ISO/IEC 14143-1 definisce i concetti fondamentali della Misura della Dimensione Funzionale (FSM – Functional Size Measurement) e descrive i principi generali per applicare un Metodo di FSM. Essa *non* fornisce regole dettagliate su come:

- Selezionare un particolare metodo

- Misurare la Dimensione Funzionale del software usando un particolare metodo
- Usare i risultati ottenuti da un particolare metodo.

La definizione di metodo FSM nella norma ISO/IEC 14143-1 è applicabile quando si voglia stabilire se un metodo per dimensionare il software è un metodo FSM. Essa non preclude lo sviluppo di vari metodi, ma fornisce piuttosto una base per accertare se un particolare metodo è conforme alla definizione di metodo FSM.

La norma ISO/IEC 14143-1 distingue fra due sottoinsiemi di requisiti utente:

- requisiti utente funzionali (FUR – Functional User Requirements),
- requisiti utente non-funzionali (NFR – Non-Functional User Requirements).

Le definizioni della norma ISO/IEC 14143-1 sono le seguenti.

## ISO/IEC 14143-1 – Definizioni

<b>Dimensione funzionale</b>	Una misura del software derivata dalla quantificazione dei Requisiti Utente Funzionali.
<b>Requisiti Utente Funzionali</b>	<p>Un sottoinsieme dei Requisiti Utente. Requisiti che descrivono che cosa il software deve fare in termini di attività e servizi.</p> <p>Nota: I requisiti utente funzionali (FUR, Functional User Requirements) includono ma non sono limitati a:</p> <ul style="list-style-type: none"> <li>• Trasferimento di dati (per esempio: acquisizione di dati dell'utente, invio di segnali di controlli).</li> <li>• Trasformazione di dati (per esempio: calcolo di interessi bancari, calcolo della temperatura media).</li> <li>• Memorizzazione di dati (per esempio: memorizzazione di ordini del cliente, registrare la temperatura ambientale nel tempo) .</li> <li>• Reperimento di dati (per esempio: elenco degli impiegati in carica, recupero della posizione di un aereo).</li> </ul>

**Requisiti  
Utente  
Non-  
Funzionali**

La norma ISO non fornisce una definizione per i Requisiti Utente Non-Funzionali, ma dà alcuni esempi in una nota.

Esempi di Requisiti Utente che sono Requisiti Utente Non-Funzionali includono, ma non sono limitati a:

- Vincoli di qualità (per esempio usabilità, affidabilità, efficienza e portabilità).
- Vincoli organizzativi (per esempio sedi delle attività, hardware su cui installare, conformità agli standard).
- Vincoli ambientali (per esempio interoperabilità, sicurezza, privacy e misure di sicurezza).
- Vincoli di realizzazione (per esempio linguaggio di sviluppo, tempi di consegna).

---

### Panoramica sul metodo FSM IFPUG

---

**Introduzione** Questo capitolo presenta una panoramica del metodo di misurazione della dimensione funzionale (FSM) IFPUG. Esso presenta una sintesi ed un esempio delle procedure di conteggio dei function point.

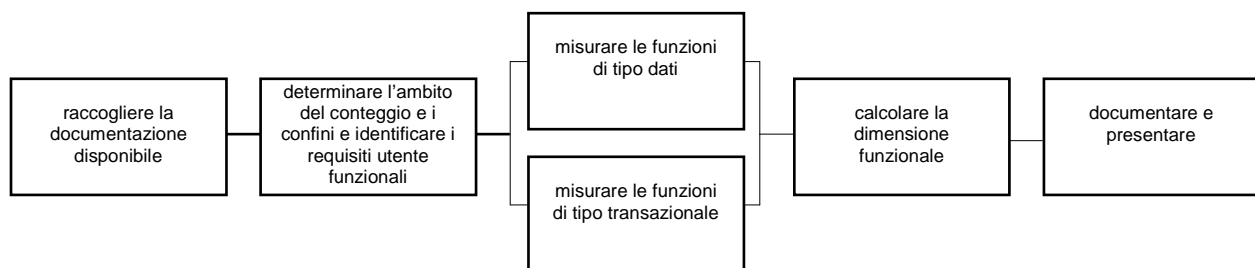
**Contenuti** Questo capitolo comprende le seguenti sezioni:

<b>Argomento</b>	<b>Pagina</b>
<b>Procedura del metodo di Misurazione della Dimensione Funzionale</b>	<b>2-2</b>
Procedure descritte in ogni capitolo	2-2
Esempio sintetico di conteggio	2-3
Diagramma riassuntivo	2-3
Raccogliere la documentazione disponibile	2-4
Determinare l'ambito del conteggio ed il confine e identificare i requisiti utente funzionali	2-4
Misurare le funzioni di tipo dati	2-5
Misurare le funzioni di tipo transazionale	2-6
Calcolare la dimensione funzionale	2-7

# Procedura del metodo di Misurazione della Dimensione Funzionale

Questa sezione descrive ad alto livello la procedura per il metodo di Misurazione della Dimensione Funzionale IFPUG.

## Diagramma della procedura



## Procedure descritte in ogni capitolo

La seguente tabella mostra le procedure per il conteggio dei function point così come sono illustrate nei successivi capitoli della Parte 2.

**Nota:** Un esempio sintetico delle procedure per il conteggio è mostrato nelle seguenti pagine di questo capitolo.

Procedura	Capitolo
Raccogliere la documentazione disponibile	3 Raccogliere la documentazione disponibile
Determinare l'ambito del conteggio e il confine e identificare i requisiti utente funzionali	4 Determinare il tipo di conteggio
Determinare l'ambito del conteggio e il confine e identificare i requisiti utente funzionali	5 Determinare l'ambito del conteggio ed il confine ed identificare i requisiti utente funzionali
Misurare le funzioni di tipo dati	6 Misurare le funzioni di tipo dati
Misurare le funzioni di tipo transazionale	7 Misurare le funzioni di tipo transazionale

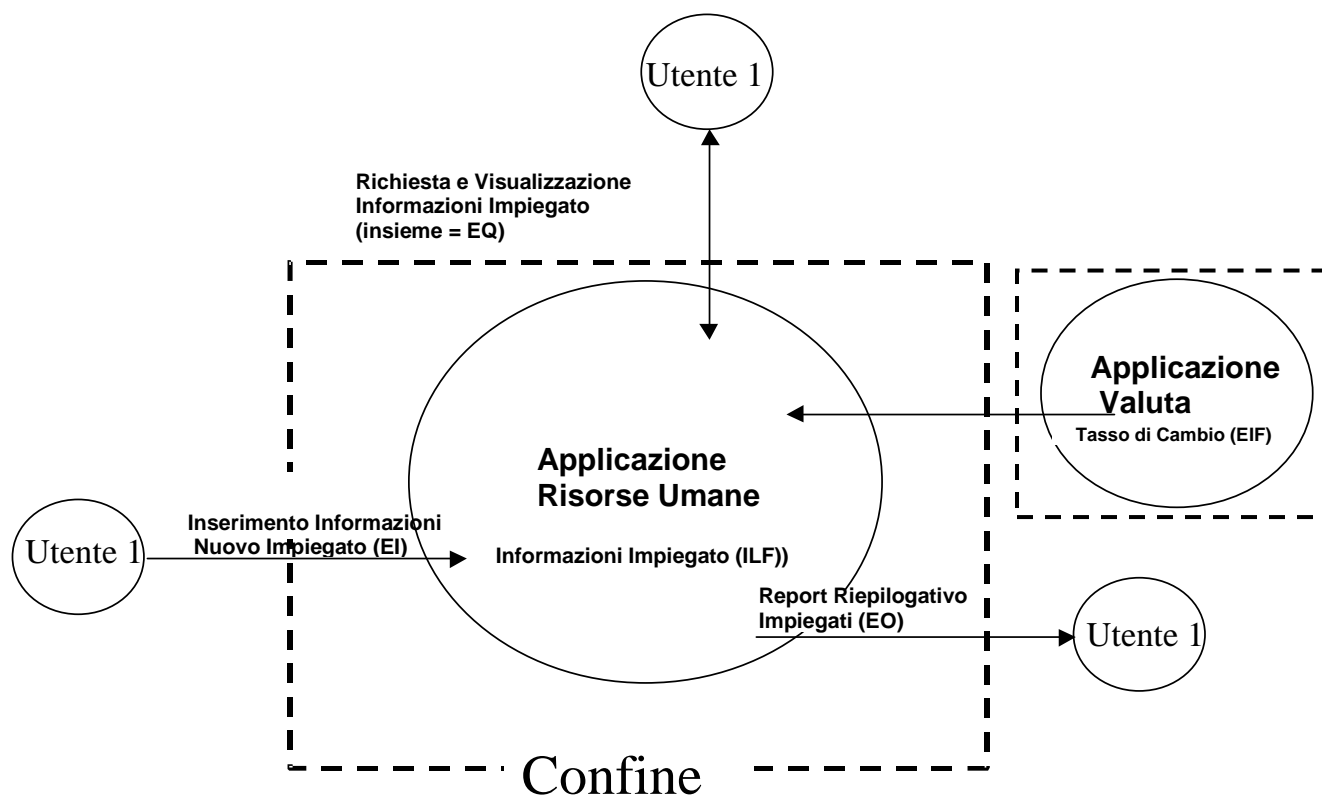
**Nota:** Non ci sono capitoli distinti per “Calcolare la dimensione funzionale” e per “Documentare e presentare” poiché questi passi della procedura non richiedono ulteriore spiegazione oltre quella fornita nella Parte 1.

## Esempio sintetico di conteggio

Questa sezione riporta un esempio sintetico della procedura per il conteggio dei function point e delle componenti che sono incluse nella dimensione funzionale.

### Diagramma riassuntivo

Il seguente diagramma mostra le componenti del conteggio dei function point nel caso dell'esempio relativo all'applicazione Risorse Umane. Il diagramma costituisce il riferimento per i rimanenti paragrafi di questo capitolo.



## Raccogliere la documentazione disponibile

Il primo passo della procedura di conteggio dei function point è raccogliere la documentazione disponibile, secondo quanto riportato nel paragrafo 5.2 della Parte 1, per supportare la misura della dimensione funzionale. Essa dovrebbe descrivere le funzionalità rilasciate dal software e le funzionalità impattate dal progetto software che deve essere misurato.

Una documentazione idonea può comprendere requisiti, modelli dati/oggetti, diagrammi delle classi, diagrammi di flusso, casi d'uso, descrizioni di procedure, layout di report, layout di maschere, manuali utente ed altri manufatti dello sviluppo software. Se la documentazione disponibile non è sufficiente, si otterrà l'accesso alle conoscenze degli esperti in materia che sono in grado di fornire informazioni aggiuntive per colmare eventuali lacune nella documentazione.

Il Capitolo 3 esamina la documentazione disponibile durante il ciclo di vita di un'applicazione.

## Determinare l'ambito del conteggio ed il confine e identificare i Requisiti Utente Funzionali

La misura della dimensione funzionale è diretta a fornire una risposta a un problema di business. Secondo quanto riportato nel paragrafo 5.3 della Parte 1, è il problema di business che determina lo scopo del conteggio.

I conteggi in function point possono essere identificati, in base al loro scopo, come:

- Conteggio in function point di un progetto di sviluppo
- Conteggio in function point di un progetto di manutenzione evolutiva
- Conteggio in function point di un'applicazione

Il Capitolo 4 fornisce una guida per determinare il tipo di conteggio in function point.

L'ambito del conteggio definisce l'insieme di Requisiti Utente Funzionali da includere nel conteggio in function point.

Il confine dell'applicazione rappresenta un'interfaccia concettuale tra il software oggetto di studio e l'utente.

Il precedente diagramma riassuntivo mostra il confine fra l'applicazione Risorse Umane, oggetto di misura, e l'applicazione esterna Valuta. Esso mostra anche il confine fra l'applicazione Risorse Umane e l'utente.

Il Capitolo 5 fornisce ulteriori chiarimenti sull'ambito del conteggio ed il confine dell'applicazione.



## Misurare le funzioni di tipo dati

Una funzione di tipo dati rappresenta la funzionalità fornita all'utente per soddisfare i requisiti relativi alla memorizzazione di dati interni ed esterni. Una funzione di tipo dati è un file logico interno o un file d'interfaccia esterno.

- Un file logico interno (ILF – Internal Logical File) è un gruppo di dati, o di informazioni di controllo, logicamente collegati, riconoscibile dall'utente, mantenuto all'interno del confine dell'applicazione oggetto di conteggio. L'intento primario di un ILF è conservare dati mantenuti tramite uno o più processi elementari dell'applicazione oggetto di conteggio.

Il precedente diagramma riassuntivo mostra un gruppo di dati relativi all'impiegato che sono mantenuti nell'applicazione Risorse Umane come esempio di ILF.

- Un file d'interfaccia esterno (EIF – External Interface File) è un gruppo di dati, o di informazioni di controllo, logicamente collegati, riconoscibile dall'utente, referenziato dall'applicazione oggetto di conteggio, ma mantenuto all'interno del confine di un'altra applicazione. L'intento primario di un EIF è conservare dati referenziati da uno o più processi elementari situati nei confini dell'applicazione oggetto di conteggio. Ciò significa che un EIF contato per un'applicazione deve essere un ILF, o parte di un ILF, in un'altra applicazione.

Il precedente diagramma riassuntivo mostra le informazioni relative al tasso di cambio mantenute dall'applicazione Valuta e referenziate dall'applicazione Risorse Umane come esempio di EIF.

Il Capitolo 6 della Parte 2 tratta l'applicazione delle regole per misurare le funzioni di tipo dati.

La Parte 3 contiene ulteriori indicazioni per misurare le funzioni di tipo dati.

La Parte 4 contiene esempi che illustrano l'applicazione delle regole per misurare le funzioni di tipo dati.

## Misurare le funzioni di tipo transazionale

Una funzione di tipo transazionale è un processo elementare che fornisce funzionalità all'utente per elaborare i dati. Una funzione di tipo transazionale è un input esterno, o un output esterno, o un'interrogazione esterna.

- Un input esterno (EI, External Input) è un processo elementare che elabora dati o informazioni di controllo che provengono 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.

Il precedente diagramma riassuntivo mostra il processo di inserimento delle informazioni relative all'impiegato nell'Applicazione Risorse Umane come esempio di un EI.

- Un output esterno (EO, External Output) è un processo elementare che invia dati o informazioni di controllo all'esterno del confine dell'applicazione e comprende un'elaborazione ulteriore, oltre quella di un'interrogazione esterna. L'intento primario di un output esterno è presentare informazioni a un utente attraverso un trattamento logico diverso dal, o in aggiunta al, 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 e/o alterare il comportamento del sistema.

Il precedente diagramma riassuntivo mostra il processo per produrre un tabulato contenente la lista di tutti gli impiegati memorizzati dell'Applicazione Risorse Umane come esempio di un EO.

- Un'interrogazione esterna (EQ, External Inquiry) è un processo elementare che invia dati o informazioni di controllo all'esterno del confine dell'applicazione. L'intento primario di un'interrogazione esterna è presentare informazioni ad un utente attraverso il reperimento di dati o informazioni di controllo. 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 viene alterato.

Il precedente diagramma riassuntivo mostra il processo per effettuare interrogazioni sulle informazioni relative ad un impiegato come esempio di EQ.

Il Capitolo 7 della Parte 2 tratta l'applicazione delle regole per misurare le funzioni di tipo transazionale.

La Parte 3 contiene indicazioni aggizionali per misurare le funzioni di tipo transazionale.

La Parte 4 contiene esempi che illustrano l'applicazione delle regole per misurare le funzioni di tipo transazionale.

## Calcolare la dimensione funzionale

La dimensione funzionale rappresenta una dimensione del software derivata dalla quantificazione dei requisiti utente funzionali (FUR).

Le funzionalità utente specifiche dell'applicazione sono valutate in termini di *che cosa* è rilasciato dall'applicazione, non *come* è rilasciato. Debbono essere contate *esclusivamente* le componenti richieste e definite dall'utente.

La dimensione funzionale è ottenuta misurando funzioni di tipo dati e funzioni di tipo transazionale. Tali funzioni sono ulteriormente definite nei tipi di funzione descritti nei paragrafi seguenti.

Alcune persone possono voler applicare un fattore di aggiustamento del valore (VAF, Value Adjustment Factor), che considera 14 caratteristiche generali del sistema (GSC, General System Characteristic). Per una guida all'uso del VAF e delle GSC, si rimanda all'Appendice C.

Questa pagina è lasciata intenzionalmente vuota.

---

### Raccogliere la documentazione disponibile

---

**Introduzione** Questo capitolo illustra la disponibilità della documentazione durante il ciclo di vita di un'applicazione e il concetto del ruolo dell'utente.

**Contenuti** Questo capitolo include le seguenti sezioni:

<b>Argomento</b>	<b>Pagina</b>
<b>Punto di vista dell'utente</b>	<b>3-2</b>
<b>Documentazione disponibile durante il ciclo di vita dell'applicazione</b>	<b>3-3</b>
Fase: requisiti utente iniziali	3-4
Fase: requisiti tecnici	3-5
Fase: requisiti utente funzionali finali	3-6
<b>Confronto per fasi del ciclo di vita</b>	<b>3-8</b>
<b>Documentazione utile di progetto/applicazione</b>	<b>3-9</b>
<b>La dimensione funzionale</b>	<b>3-10</b>

## Punto di vista dell'utente

Un *utente* è qualsiasi persona o cosa che comunica o interagisce con il software in qualsiasi momento.

Un punto di vista dell'utente è l'insieme dei Requisiti Utente Funzionali *così come sono percepiti* dall'utente.

I Requisiti Utente Funzionali sono un sotto-insieme dei requisiti utente che specificano cosa deve fare il software in termini di attività e servizi.

Un punto di vista dell'utente rappresenta una descrizione formale delle esigenze applicative dell'utente, espressa nel linguaggio dell'utente. Gli sviluppatori traducono le informazioni dell'utente nel linguaggio dell'Information Technology per fornire una soluzione.

Un punto di vista dell'utente:

- è una descrizione delle funzionalità applicative,
- può essere un insieme di affermazioni verbali fatte dagli utenti su quale sia la propria visione,
- è approvato dall'utente,
- può essere usato per misurare la dimensione funzionale,
- può variare nella sua forma fisica (per es. Catalogo di transazioni, proposte, documento di requisiti, specifiche esterne, specifiche di dettaglio, manuale utente).

Una misurazione di dimensione funzionale è svolta facendo uso delle informazioni in un linguaggio che sia comune sia all'utente/i, sia agli sviluppatori.

## Documentazione disponibile durante il ciclo di vita dell'applicazione

I requisiti utente evolvono rapidamente nelle fasi iniziali di un progetto. Occorre concordare delle decisioni tra utenti e sviluppatori, su quali funzioni saranno incluse in un'applicazione. Queste decisioni relative alle funzioni del progetto possono essere influenzate da:

- le esigenze dell'organizzazione,
- il rischio (economico e tecnico) associato al progetto,
- le risorse disponibili (per es. budget, personale) per il progetto nell'organizzazione,
- la tecnologia disponibile nell'organizzazione,
- l'influenza esercitata da utenti o sviluppatori attraverso commenti e consigli.

All'inizio di un progetto, si produce lo studio di fattibilità. Questo costituisce il livello più alto delle specifiche ed è di solito molto breve; per esempio:

- l'organizzazione necessita di un'applicazione per attenersi ad una nuova legge fiscale;
- l'organizzazione necessita di un'applicazione per gestire l'inventario in modo più efficiente;
- l'organizzazione necessita di un'applicazione per gestire il personale in modo più efficiente.

Dopo lo studio di fattibilità, l'utente sviluppa i requisiti che diventano via via più precisi. Ad un certo punto, l'utente si consulterà con gli sviluppatori per creare i requisiti dettagliati. Gli sviluppatori possono accelerare l'avvio del progetto con propri requisiti di sviluppo e realizzazione, basandosi sullo studio di fattibilità. Il dialogo tra utenti e sviluppatori conduce a requisiti migliorati. Il processo di sviluppo varia da organizzazione a organizzazione. Questo manuale considera, per semplicità, un modello con tre categorie di documenti dei requisiti:

- requisiti utente iniziali
- requisiti tecnici iniziali
- requisiti utente funzionali finali

Come in altre metodologie di sviluppo, la fase dei requisiti utente funzionali finali è quella più accurata per misurare la dimensione funzionale.

## Fase: requisiti utente iniziali

Questa fase rappresenta i requisiti utente antecedenti gli incontri tra gli utenti e gli sviluppatori. Può avere una o più delle seguenti caratteristiche:

- È incompleta.  
Per esempio, nei requisiti utente iniziali possono mancare le specifiche necessarie per l'integrità referenziale.
- Non comprende le funzioni di "utilità".  
Per esempio, possono mancare prospetti e interrogazioni di validazione importanti.
- È impossibile da implementare o molto difficile da utilizzare.  
Per esempio, un utente può richiedere un'interrogazione online che richiede un'ora di elaborazione della CPU.
- È troppo generica.  
Per esempio, i requisiti possono non includere l'elenco specifico dei campi riconoscibili dall'utente.
- Non affronta le esigenze di tutti gli utenti dell'applicazione.  
Per esempio, i requisiti di uno specifico progetto possono cambiare da un utente all'altro, se essi non hanno le stesse necessità funzionali.
- I requisiti sono formulati senza tenere conto dei confini applicativi.  
Per esempio, i confini applicativi attuali e/o futuri non sono stati considerati.
- I requisiti sono espressi in un contesto differente o con una terminologia incompatibile con l'analisi dei function point.  
Per esempio, i Requisiti Utente Iniziali possono fare riferimento ad aspetti fisici o manuali del sistema.

### Esempio

Nel Dipartimento Risorse Umane di un'azienda, un utente esprime i propri requisiti come:

“Quando lavoro con un impiegato, voglio poter vedere le sue informazioni semplicemente inserendo il suo nome.”

Questo requisito implica lo sviluppo di un'interrogazione e di un insieme di dati sugli impiegati.

Funzioni di questo esempio di requisiti utente iniziali:

- EQ     interrogazione su uno specifico impiegato
- ILF    gruppo di dati dell'impiegato



## Fase: requisiti tecnici

Questa seconda fase rappresenta il punto di vista degli sviluppatori sui requisiti creati in base allo studio di fattibilità. Uno dei compiti degli sviluppatori, tra gli altri, è distribuire i requisiti sulle applicazioni esistenti, se ve ne sono. I requisiti tecnici possono includere elementi che sono necessari per l'implementazione, ma che non sono considerati nella misurazione della dimensione funzionale (per es. file temporanei, indici, ecc.). Questa fase può avere una o più delle seguenti caratteristiche:

- Dipendenza dalla tecnologia.

Per esempio, i file fisici variano in funzione dell'ambiente di database.

- Terminologia non familiare all'utente.

Per esempio, gli sviluppatori possono far riferimento a file fisici piuttosto che a gruppo di dati logici.

- Individuazione delle funzionalità ponendo enfasi eccessiva sui vincoli tecnici.

Per esempio, alcuni sviluppatori tendono a circoscrivere l'ambito dei requisiti concentrandosi sulle capacità elaborative attualmente disponibili nell'organizzazione.

- Individuazione dei confini in funzione dell'architettura tecnica piuttosto che dei processi aziendali.

Per esempio, ci possono essere requisiti tecnici distinti per il client e per il server, ma essi sarebbero compresi nel medesimo confine applicativo quando si misura la dimensione funzionale.

### Esempio

Sviluppatore: “Individuo la necessità di un'interrogazione sugli impiegati. È necessario un indice per velocizzare il reperimento dei dati di specifici impiegati.”.

Le funzioni dei requisiti tecnici iniziali potrebbero essere identificate come:

EQ     interrogazione su uno specifico impiegato

ILF    gruppo di dati dell'impiegato

ILF\*   indice del file degli impiegati

\* I file di indici non sono inclusi nella misurazione della dimensione funzionale. In questo esempio, il file di indice è erroneamente identificato come un ILF proprio per illustrare un potenziale errore di conteggio da parte degli sviluppatori.

## Fase: requisiti utente funzionali finali

Questa terza fase dei requisiti è il risultato degli incontri congiunti tra utente/i e sviluppatore/i. Tali incontri congiunti sono necessari per ottenere dei requisiti utente funzionali omogenei e completi per l'applicazione. Ciò fornisce la versione finale dei requisiti utente funzionali prima che abbia inizio la fase di sviluppo e ha le seguenti caratteristiche:

- La terminologia può essere compresa sia dagli utenti, sia dagli sviluppatori.
- Fornisce descrizioni integrali di tutti i requisiti utente, includendo i requisiti di tutti i gruppi di utenti.
- Tutti i processi aziendali sono completamente definiti, incluse tutte le azioni utente, i campi che entrano o escono dal confine applicativo, le fonti dei dati e le validazioni richieste come parte di ciascun processo aziendale.
- Qualsiasi processo e gruppo di dati è concordato da utente e sviluppatore.
- La fattibilità e l'usabilità sono approvate dagli sviluppatori.

### Esempio

Utente: “Quando lavoro con un impiegato, voglio poter vedere le sue informazioni semplicemente inserendo il suo nome.”

Sviluppatore: “Individuo la necessità di un'interrogazione sugli impiegati, ma molti impiegati possono avere lo stesso nome. Non è possibile specificare un singolo impiegato digitando il suo nome; perciò, suggerisco una lista online degli impiegati (nome, sede e codice fiscale) da cui selezionare un impiegato. Sarà necessario un indice per velocizzare il reperimento di uno specifico impiegato.”

Utente: “Concordo sul fatto che sia necessaria in questo caso una lista di selezione degli impiegati, ed essa può essere usata anche per altri fini diversi dalla selezione del singolo impiegato.”

Risultato della discussione tra l'utente e lo sviluppatore:

- aggiungere la lista online degli impiegati ai requisiti utente funzionali e alla dimensione funzionale;
- escludere l'indice degli impiegati dalla dimensione funzionale poiché è una soluzione tecnica.

Funzioni di questo esempio di requisiti utente iniziali funzionali finali:

- EQ    interrogazione su uno specifico impiegato
- EQ    lista online degli impiegati
- ILF   gruppo di dati dell'impiegato

Il documento dei requisiti utente funzionali finali è la versione finale dei requisiti prima che abbia inizio la fase di sviluppo. A questo punto, ci dovrebbe essere accordo sul fatto che i requisiti documentati siano completi, formali ed approvati. La misurazione della dimensione funzionale, assumendo che non vi siano ulteriori variazioni di ambito, dovrebbe essere coerente con la misurazione al completamento dello sviluppo.

## Confronto per fasi del ciclo di vita

Prima di iniziare una misurazione della dimensione funzionale, si determini la fase del ciclo di vita e se si sta approssimando o misurando la dimensione. Si documenti qualsiasi assunto.

L'approssimazione permette di fare delle ipotesi su funzioni e/o loro complessità non note, per determinare una dimensione funzionale approssimata.

La misurazione include l'identificazione di tutte le funzioni e delle loro complessità, per svolgere un'analisi dei function point.

In uno stadio iniziale, i Requisiti Utente Iniziali potrebbero essere il solo documento disponibile per l'analisi dei function point. Nonostante gli svantaggi, questa dimensione può essere molto utile per produrre una stima iniziale. Gli utilizzi dell'analisi dei function point per approssimare la dimensione nelle varie fasi del ciclo di vita sono riportati di seguito:

Fase del Ciclo di Vita	La dimensione può essere approssimata	La dimensione può essere misurata
<b>Proposta:</b> gli utenti esprimono esigenze ed intenzioni.	sì	no
<b>Requisiti:</b> gli sviluppatori e gli utenti rivedono e concordano l'espressione di esigenze e intenzioni degli utenti.	sì	sì
<b>Progettazione:</b> gli sviluppatori possono inserire elementi implementativi che non sono usati per l'analisi dei function point.	sì	sì
<b>Realizzazione</b>	sì	sì
<b>Rilascio</b>	sì	sì
<b>Manutenzione</b>	sì	sì
<p><b>Nota:</b> Non è suggerito alcun ciclo di vita specifico per lo sviluppo. Se si utilizza un approccio iterativo, ci si può attendere di approssimare la dimensione per un certo tempo nel ciclo di vita dell'applicazione.</p> <p>Si ricordi di misurare unicamente i requisiti nuovi o revisionati in base alle esigenze ed alle intenzioni dell'utente.</p>		

## Documentazione utile di progetto/applicazione

In generale, le seguenti voci sono utili quando si svolge una qualsiasi misurazione di dimensione funzionale:

- documento/i dei requisiti,
- diagrammi entità/relazioni,
- modelli a oggetti,
- modelli dei dati,
- schemi di file e di database (con individuati gli attributi logici richiesti dall'utente),
- protocolli di interfacciamento con descrizioni delle procedure di fornitura / file transazione e interfacce da/verso altre applicazioni,
- esempi di report, schermate online e altre interfacce utente,
- demo di operazioni applicative,
- uno o più esperti dell'applicazione (per l'applicazione oggetto di misurazione),
- uno o più clienti / utenti dell'applicazione (a disposizione durante le sessioni di conteggio),
- guida utente, manuali di formazione e help dell'applicazione,
- documentazione di progettazione del sistema,
- specifiche funzionali,
- casi d'uso.

**Nota:** la precedente lista non è esaustiva.

## La dimensione funzionale

Il Manuale delle Regole di Conteggio (CPM, Counting Practices Manual) IFPUG è stato trasformato in uno standard ISO per la misurazione della dimensione funzionale con l'esclusione delle Caratteristiche Generali del Sistema, che misurano requisiti non-funzionali (tecnici e di qualità). Entro certi limiti, questa trasformazione ha permesso al Comitato per le Regole del Conteggio (CPC, Counting Practices Committee) di affrontare alcune questioni, come le tabelle di decodifica, in maniera coerente. La considerazione più importante, in rapporto a tali questioni, riguarda in che modo i requisiti non-funzionali abbiano impatto sulla dimensione. Poiché essi non sono requisiti utenti funzionali, non contribuiscono alla dimensione funzionale. Tuttavia, essi fanno sempre parte dei requisiti utente complessivi (funzionali e non-funzionali) sul software e contribuiscono perciò alla dimensione complessiva dei requisiti.

Per una discussione dettagliata della dimensione funzionale e di come essa guiderà e vincolerà l'evoluzione dell'analisi dei function point, si veda il documento "Framework for Functional Sizing" dell'IFPUG<sup>1</sup>.

---

<sup>1</sup> Framework for Functional Sizing, IFPUG CPC, Versione 1.0, Settembre 2003.

---

### Determinare il tipo di conteggio

---

**Introduzione** Questo capitolo comprende una spiegazione dettagliata dei tipi di conteggio: progetto di sviluppo, progetto di manutenzione evolutiva e applicazione.

**Contenuti** Questo capitolo comprende le seguenti sezioni:

Argomento	Pagina
<b>Definizioni: tipi di conteggio dei function point</b>	<b>4-2</b>
Progetto di sviluppo	4-2
Progetto di manutenzione evolutiva	4-2
Applicazione	4-3
<b>Diagramma dei tipi di conteggio</b>	<b>4-4</b>
Misura della dimensione funzionale stimata e finale	4-4

## Definizioni: tipi di conteggio dei function point

La dimensione funzionale può essere misurata sia per progetti che per applicazioni. Il tipo di conteggio dei function point è determinato, basandosi sullo scopo del conteggio, come uno dei seguenti:

- Conteggio in function point per un progetto di sviluppo
- Conteggio in function point per un progetto di manutenzione evolutiva
- Conteggio in function point di un'applicazione.

I paragrafi seguenti definiscono ciascun tipo di conteggio in function point.

**Nota:** Per le persone che applicano il fattore di aggiustamento del valore (VAF, Value Adjustment Factor), referirsi all'Appendice C per le formule per calcolare il VAF e la dimensione funzionale aggiustata.

### Progetto di sviluppo

Un progetto di sviluppo è un progetto per sviluppare e rilasciare la prima versione di un'applicazione software.

La dimensione funzionale di un progetto di sviluppo è una misura delle funzionalità fornite all'utente con la prima versione del software, come misurate dal conteggio in function point di un progetto di sviluppo attraverso l'applicazione del Metodo di Misura della Dimensione Funzionale (FSM) IFPUG.

### Progetto di manutenzione evolutiva

Un progetto di manutenzione evolutiva è un progetto che sviluppa e rilascia una manutenzione adeguativa.

La dimensione funzionale di un progetto di manutenzione evolutiva è una misura delle funzionalità aggiunte, modificate o cancellate al completamento del progetto di manutenzione evolutiva, come misurate dal conteggio in function point di un progetto di manutenzione evolutiva attraverso l'applicazione del Metodo di Misura della Dimensione Funzionale (FSM) IFPUG.

Linee guida aggiuntive sono riportate nella Parte 3.



## Applicazione

Un'applicazione è una collezione coesa di procedure e dati automatizzati che supporta un obiettivo di business; essa consiste di uno o più componenti, moduli o sottosistemi.

La dimensione funzionale di un'applicazione è una misura delle funzionalità che l'applicazione fornisce all'utente, determinata attraverso l'applicazione del Metodo di Misura della Dimensione Funzionale (FSM) IFPUG.

Essa è anche chiamata *baseline* o dimensione funzionale *installata*. Questa dimensione fornisce una misura dei function point correnti che l'applicazione fornisce all'utente. Questo numero è inizializzato quando il conteggio in function point del progetto di sviluppo è completato. È aggiornato ogni volta che un progetto di manutenzione evolutiva completato altera le funzioni dell'applicazione.

## Diagramma dei tipi di conteggio

Il seguente diagramma illustra i tipi di conteggio dei function point e le loro relazioni (il Progetto A è stato completato per primo, seguito successivamente dal Progetto B).



### Misure della dimensione funzionale iniziale e finale

È importante tenere presente che le misure della dimensione funzionale iniziali sono stime delle funzionalità che verranno rilasciate al termine dei progetti. Inoltre, non appena risultano chiari gli obiettivi e le funzioni sono state sviluppate, è piuttosto normale identificare delle ulteriori funzionalità che non erano state specificate nei requisiti iniziali. Questo fenomeno è talvolta indicato con la locuzione *‘variazione dell’ambito’* (*scope creep*).

E' essenziale aggiornare la dimensione funzionale dell'applicazione al termine del progetto. Se le funzionalità cambiano durante lo sviluppo, la dimensione funzionale alla fine del ciclo di vita dovrebbe riflettere accuratamente tutte le funzionalità rilasciate all'utente.

# Determinare l'ambito del conteggio e il confine e identificare i requisiti utente funzionali

---

**Introduzione** Questo capitolo fornisce una guida per l'applicazione delle regole, procedure e suggerimenti per determinare il confine dell'applicazione e per stabilire l'ambito del conteggio.

**Contenuti** Questo capitolo include le seguenti sezioni:

<b>Argomento</b>	<b>Pagina</b>
<b>Ambito del conteggio e confine</b>	<b>5-2</b>
Scopo del conteggio	5-2
Ambito del conteggio	5-3
Confine	5-4
<b>Determinare l'ambito del Conteggio e il confine – Regole e procedure</b>	<b>5-5</b>
Regole relative al confine	5-5
Procedure relative all'ambito del conteggio ed al confine	5-5
<b>Suggerimenti di ausilio per identificare l'ambito del conteggio e il confine</b>	<b>5-6</b>

## Ambito del conteggio e confine

Questa sezione tratta l'ambito del conteggio ed il confine dell'applicazione e spiega come questi siano influenzati dallo scopo del conteggio.

### Scopo del conteggio

Una misura di dimensione funzionale è finalizzata a fornire una risposta a un problema di business, ed è il problema di business che determina lo scopo.

Lo scopo:

- determina il tipo e l'ambito del conteggio richiesto al fine di ottenere la risposta al problema applicativo in esame;
- influenza la collocazione del confine tra il software in esame e quello circostante; per esempio, se il Modulo Personale del Sistema Risorse Umane venisse sostituito da un pacchetto, l'utente potrebbe decidere di ridefinire il confine e considerare il Modulo Personale come un'applicazione separata.

Esempi di scopo sono:

- determinare la dimensione funzionale di un progetto per usarla come input ad un processo di stima per l'individuazione dell'impegno necessario allo sviluppo della prima versione di un'applicazione;
- determinare la dimensione funzionale di applicazioni installate, per determinare i costi di assistenza per function point;
- determinare la dimensione funzionale di due pacchetti di due potenziali fornitori per essere in grado di confrontare le funzionalità rilasciate da ciascuno.

## Ambito del conteggio

L'ambito del conteggio definisce l'insieme di Requisiti Utente Funzionali che saranno inclusi nel conteggio in function point. L'ambito:

- definisce un (sotto) insieme del software oggetto di misura,
- è determinato dallo scopo stabilito per il conteggio dei function point,
- identifica quali funzioni dovranno essere incluse nella misura della dimensione funzionale così da fornire risposte rilevanti allo scopo del conteggio,
- potrebbe includere più di un'applicazione.

L'*ambito* di:

- un conteggio di function point per un progetto di sviluppo include tutte le funzionalità coinvolte (realizzate o personalizzate) dalle attività del progetto. Esso include anche le funzionalità di conversione sviluppate come parte del progetto di sviluppo.
- un conteggio di function point per un progetto di manutenzione evolutiva include tutte le funzionalità che sono state aggiunte, variate e cancellate. Esso include anche le funzionalità di conversione sviluppate come parte del progetto di manutenzione evolutiva. Il confine dell'applicazione/i interessata/e rimane immutato. Il funzionamento dell'applicazione/i rispecchia l'impatto delle funzionalità che sono state aggiunte, cambiate o cancellate.
- un conteggio di function point per un'applicazione può includere, in base allo scopo (per esempio, procurarsi un pacchetto per una soluzione software):
  - solo le funzionalità che sono utilizzate dagli utenti;
  - tutte le funzionalità disponibili.

L'ambito dei due conteggi sopra è diverso risultando misurata una diversa dimensione funzionale per la stessa applicazione. Tuttavia la determinazione del confine della/e applicazione rimane la stessa e non è influenzata dalla decisione di cambiare l'ambito. La determinazione del confine è indipendente dall'ambito.

## Confine

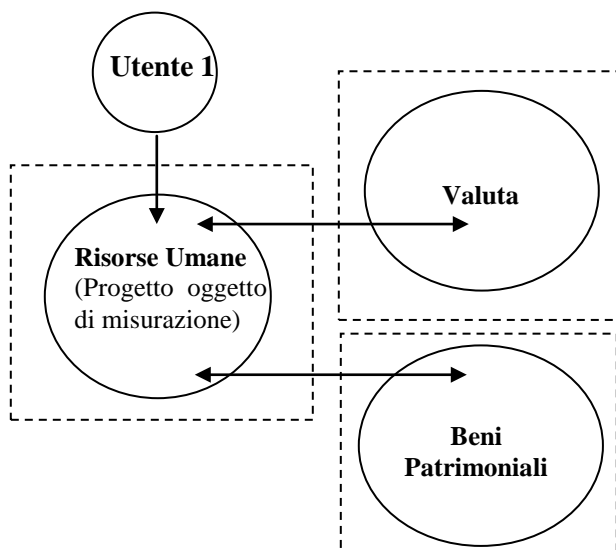
Il confine è un'interfaccia concettuale tra il software sotto esame ed i suoi utenti.

Il confine (anche indicato come confine dell'applicazione):

- Definisce che cosa è esterno all'applicazione
- Indica la linea di separazione tra il software oggetto di misurazione e l'utente.
- 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 l'utente ha dell'applicazione. È indipendente da considerazioni tecniche e/o di natura implementativa.

La determinazione del confine tra il software in esame e le altre applicazioni software può essere soggettivo. Spesso è difficile delineare dove un'applicazione termina ed un'altra inizia. Cerca di collocare il confine da un punto di vista di business, piuttosto che in base a considerazioni tecniche o fisiche. È importante che il confine applicativo sia determinato con attenzione, visto che tutti i dati che attraversano il confine possono potenzialmente essere inclusi nell'ambito del conteggio.

Per esempio, il seguente diagramma mostra i confini fra l'applicazione Risorse Umane e le due applicazioni esterne Valuta e Beni Patrimoniali. L'esempio mostra anche il confine tra l'utente umano (Utente 1) e l'applicazione Risorse Umane.



## Determinare l'ambito del conteggio e il confine – Regole e procedure

Questa sezione definisce le regole e le procedure da applicare per determinare l'ambito del conteggio e il confine della/e applicazioni.

La determinazione del confine è importante perché impatta il risultato della misura della dimensione funzionale. Il confine aiuta nell'identificare i dati che entrano nell'applicazione e che saranno inclusi nell'ambito del conteggio.

### Regole relative al confine

Per individuare i confini devono essere applicate le seguenti regole:

- ❑ Il confine è individuato basandosi sul punto di vista dell'utente.  
L'attenzione è su ciò che l'utente può capire e descrivere.
- ❑ Il confine fra applicazioni collegate è basato su aree funzionali distinte dal punto di vista dell'utente e non in funzione degli aspetti tecnici.
- ❑ Il confine iniziale già stabilito per l'applicazione o le applicazioni da modificare non è influenzato dall'ambito del conteggio.

**Nota:** L'ambito del conteggio può includere più di una applicazione. In tal caso dovrebbero essere identificati tanti confini quante sono le varie applicazioni. Quando il confine di un'applicazione non è ben definito (ad esempio nelle prime fasi di analisi), allora lo si dovrebbe localizzare nel modo più accurato possibile.

### Procedure relative all'ambito del conteggio ed al confine

Quando si effettua una misura di dimensione funzionale, dovrebbero essere eseguiti i seguenti passi:

Passo	Azione
1	Stabilire lo scopo del conteggio.
2	Identificare l'ambito del conteggio.
3	Identificare i confini della(e) applicazione(i).
4	Documentare i seguenti elementi: <ul style="list-style-type: none"> <li>• lo scopo del conteggio;</li> <li>• l'ambito del conteggio;</li> <li>• il confine della(e) applicazione(i);</li> <li>• qualsiasi assunzione correlata con quanto sopra .</li> </ul>

## Suggerimenti di ausilio per identificare l'ambito del conteggio e il confine

### Ambito del conteggio

I seguenti suggerimenti possono essere di aiuto per identificare l'ambito del conteggio:

- Esamina lo scopo del conteggio dei function point come ausilio per determinare l'ambito del conteggio.
- Quando si identifica l'ambito del conteggio della dimensione funzionale installata o di baseline (per esempio, le funzionalità in carico al gruppo di manutenzione), considera tutte le funzioni utilizzate dall'utente che sono attualmente in produzione.

### Confine

I seguenti suggerimenti possono essere di aiuto per identificare il confine dell'applicazione:

- Usa le specifiche di sistema esterne, o ottieni un diagramma di flusso del sistema, e disegna una linea intorno ad esso per evidenziare quali parti sono interne e quali sono esterne all'applicazione.
- Considera come sono mantenuti i gruppi di dati.
- Identifica le aree funzionali assegnando l'appartenenza di certi tipi di oggetti di analisi (come le entità o i processi elementari) a specifiche aree funzionali.
- Considera gli altri dati di misura associati ai function point, come l'impegno, il costo e i difetti. I confini individuati per i function point e per gli altri dati di misura dovrebbero essere gli stessi.
- Intervista gli esperti in materia per assistenza nell'identificazione del confine.



---

### Misurare le funzioni di tipo dati

---

**Introduzione** Questo capitolo fornisce indicazioni per l'applicazione delle norme e delle procedure per misurare le funzioni di tipo dati.

Una funzione di tipo dati rappresenta la funzionalità fornita all'utente per soddisfare i requisiti relativi alla memorizzazione dati interni ed esterni. Le funzioni di tipo dati sono definite come file interni logici (ILF – Internal Logical File) e file esterni di interfaccia (EIF – External Interface File).

Il termine *file* qui utilizzato non deve essere associato ai file intesi nel modo tradizionale dell'elaborazione dei dati. In questo caso, file si riferisce ad un gruppo di dati logicamente collegati e non all'implementazione fisica di tali gruppi di dati.

**Contenuti** Questo capitolo include quanto segue:

Argomento	Pagina
<b>Definizione e intento primario: ILF e EIF</b>	<b>6-2</b>
File Logico Interno	6-2
File d'Interfaccia Esterno	6-2
Differenza tra ILF ed EIF	6-2
Definizione dei termini utilizzati	6-2
<b>Procedura di conteggio delle funzioni di tipo dati</b>	<b>6-4</b>
Regole di identificazione delle funzioni di tipo dati	6-4
Regole di classificazione delle funzioni di tipo dati	6-5
Definizioni e regole per la complessità e il contributo	6-5
DET: Definizione	6-5
DET: Regole	6-5
RET: Definizione	6-7
RET: Regole	6-7
Determinazione della complessità e il contributo	6-8
<b>Suggerimenti di aiuto al conteggio</b>	<b>6-10</b>

## Definizione e intento primario: ILF ed EIF

Questa sezione è dedicata alle definizioni e intento primario dei File Interni Logici (ILF) e dei File Esterni di Interfaccia (EIF). All'interno di questa sezione sono definiti i termini utilizzati per le definizioni e sono riportati degli esempi.

### File Logico Interno (ILF, Internal Logical File)

Un file interno logico (ILF) è un gruppo di dati logicamente collegati o di informazioni di controllo, riconoscibili dall'utente, mantenuti all'interno del confine dell'applicazione sottoposta al conteggio. L'intento primario di un ILF è di contenere dati mantenuti da uno o più processi elementari dell'applicazione sottoposta al conteggio.

### File Esterno d'Interfaccia (EIF, External Interface File)

Un file esterno di interfaccia (EIF) è un gruppo di dati logicamente collegati o di informazioni di controllo, riconoscibili dall'utente, referenziati dall'applicazione ma mantenuti all'interno del confine di un'altra. L'intento primario di un EIF è quello di contenere dati referenziati da uno o più processi elementari situati nei confini dell'applicazione sottoposta al conteggio. Un EIF contato per un'applicazione deve essere un ILF in un'altra applicazione.

### Differenze fra ILF ed EIF

La principale differenza fra un ILF ed un EIF è che l'EIF **non** è mantenuto dall'applicazione che si sta misurando, mentre un ILF lo è.

### Definizioni dei termini utilizzati

I paragrafi seguenti descrivono ulteriormente gli ILF e EIF attraverso la definizione dei termini contenuti nella loro definizione.

#### Informazioni di controllo

Le *informazioni di controllo* sono dati che influenzano un processo elementare dell'applicazione che si sta misurando. Essi specificano cosa, quando o come un dato deve essere elaborato.

Per esempio, qualcuno nel dipartimento del personale stabilisce cicli di pagamento per pianificare quando devono essere pagati gli impiegati di ciascuna sede. Il ciclo di pagamento, o schedulazione, contiene informazioni temporali che condizionano il momento in cui avverrà il processo elementare di pagamento degli impiegati.

<b>Riconoscibile dall'utente</b>	<p>Il termine <i>riconoscibile dall'utente</i> si riferisce ai requisiti definiti per i processi e/o i gruppi di dati concordati, e compresi, sia dal/dagli utenti che dal/dagli sviluppatori del software.</p> <p><u>Per esempio</u>, gli utenti e gli sviluppatori software concordano che l'Applicazione Risorse Umane manterrà e memorizzerà le informazioni sugli impiegati.</p>
<b>Mantenere</b>	<p>Il termine <i>mantenere</i> si riferisce alla capacità di aggiungere, cambiare o cancellare i dati attraverso un processo elementare.</p> <p>Gli <u>esempi</u> comprendono, ma non sono limitati a aggiungere, cambiare, cancellare, popolare, rivedere, aggiornare, assegnare e creare.</p>
<b>Processo elementare</b>	<p>Un <i>processo elementare</i> è la più piccola unità di attività che è significativa per l'utente.</p> <p>Comporre e/o decomporre i Requisiti Funzionali dell'Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"><li>• è significativa per l'utente,</li><li>• costituisce una transazione completa,</li><li>• è autonoma, e</li><li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale.</li></ul> <p><u>Per esempio</u>, l'utente richiede che l'inserimento di un impiegato comprenda le informazioni sul salario e le informazioni ad esso collegate. Se non sono inserite tutte le informazioni sull'impiegato, quest'ultimo non risulta ancora creato. Inserire solamente una parte delle informazioni comporta l'inserimento di un impiegato in uno stato di incoerenza. Se entrambe le informazioni sul salario e quelle ad esso collegate sono invece inserite, questa unità di attività è completata e l'applicazione è lasciata in uno stato coerente.</p>

## Procedura di conteggio delle funzioni di tipo dati

Questa sezione fornisce indicazioni su come applicare le regole per la misurazione delle funzioni di tipo dati (ILF e EIF).

Questa sintesi è riportata per illustrare le regole nell'ambito delle procedure di conteggio degli ILF ed EIF.

**Nota:** Il dettaglio sulle procedure di conteggio inizia alla pagina 6-4. Ulteriori illustrazioni ed esempi sono riportati nella Parte 3 File Logici.

Le procedure di conteggio di tipo dati includono i seguenti passi:

Passo	Azione
1	Identificare le funzioni di tipo dati.
2	Classificare ogni funzione di tipo dati come un ILF o EIF.
3	Determinare la complessità degli ILF e degli EIF ed il loro contributo alla dimensione funzionale.

## Regole di Identificazione delle funzioni di tipo dati

Una funzione di tipo dati rappresenta la funzionalità fornita all'utente per soddisfare i requisiti di archiviazione dei dati interni ed esterni. Una funzione di tipo dati è un file logico interno (ILF) o un file esterno d'interfaccia (EIF).

**Nota:** le funzioni di tipo dati sono più facilmente identificate utilizzando un modello logico dei dati; tuttavia, ciò non preclude l'utilizzo del processo di misurazione in ambienti in cui sono impiegati dati o tecniche alternative di modellazione degli oggetti. La terminologia di modellazione dei dati viene utilizzata per documentare le regole per le funzioni di tipo dati, ma lo stesso approccio può essere applicato con altre tecniche.

Al fine di identificare le funzioni di tipo dati, devono essere eseguite le seguenti attività:

- ❑ Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.
- ❑ Escludere le entità non mantenute da alcuna applicazione.
- ❑ Raggruppare le entità collegate dipendenti (vedi Parte 3, File Logici).

**Nota:** entità indipendenti sono considerate gruppi logici di dati distinti.

- ❑ Escludere le entità referenziate come dati di decodifica (vedi Parte 3, Dati di Decodifica).

- ❑ Escludere entità che non contengono attributi richiesti dall'utente.
- ❑ Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.

**Nota:** gli attributi di chiave esterna sono dati richiesti dall'utente per stabilire una relazione con un'altra funzione di tipo dati.

## Regole di classificazione delle funzioni di tipo dati

Classificare come ILF se il gruppo di dati è mantenuto dall'applicazione che si sta misurando.

Classificare come EIF, se il gruppo di dati:

- ❑ è referenziato, ma non mantenuto, dall'applicazione che si sta misurando, e
- ❑ è identificato come ILF in una o più ulteriori applicazioni.

**Nota:** se la funzione di tipo dati soddisfa entrambe le regole, va classificata come ILF.

## Definizioni e regole per la complessità e il contributo

Il numero di ILF, EIF e la loro relativa complessità funzionale determinano il contributo delle funzioni di tipo dati alla dimensione funzionale.

Assegnare ad ogni ILF e EIF identificato una complessità funzionale basata sul numero di tipi di elemento dati (DET) e tipi di elemento record (RET) associati con l'ILF o l'EIF.

Questa sezione fornisce una definizione per DET e RET ed include le regole di conteggio per ciascuno di essi.

**DET:**  
**Definizione** Un *tipo di elemento dati* (DET) è un campo unico riconoscibile dall'utente e non ripetuto.

**DET: Regole** Le seguenti regole si applicano quando si contano i DET

- ❑ Contare un DET per ciascun campo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di un qualsiasi processo elementare nell'ambito del conteggio.

Per esempio, il/i risultato/i di un calcolo effettuato da un processo elementare, come il valore calcolato della tassa sulle vendite relativo all'ordine di un cliente, mantenuto su un ILF, è contato come un DET sull'ILF degli Ordini Clienti.

Per esempio, reperire il prezzo di un articolo che è salvato in un archivio di Fatture o campi come il “*time stamp*”, se richiesti dal/dagli utente/i, sono contati come DET.

Per esempio, se la matricola di un impiegato appare due volte in un ILF o un EIF come: (1) la chiave primaria del record Impiegato e (2) una chiave esterna nel record Dipendente, contare il DET solo una volta.

Per esempio, nell’ambito di un ILF o EIF, contare un DET per i 12 campi dell’Importo Mensile di Budget. Contare un campo addizionale per identificare l’indicazione del mese.

- ❑ Quando due applicazioni mantengono e/o referenziano la stessa funzione di tipo dati, contare solo i DET effettivamente utilizzati dall’applicazione misurata.

**Nota:** Gli attributi che non sono referenziati dall’applicazione misurata non sono contati.

Per esempio, l’applicazione A può specificatamente identificare ed utilizzare un indirizzo come: via, città, stato e codice di avviamento postale. L’applicazione B può vedere l’indirizzo come un blocco dati senza necessità di individuare i singoli componenti.

L’applicazione A conterà 4 DET; l’applicazione B conterà 1 DET.

Per esempio, l’applicazione X mantiene e/o referencia un ILF che contiene il Codice Fiscale, il Nome, la Via, il Numero civico, la Città, lo Stato e il Codice di Avviamento Postale. L’applicazione Z mantiene e/o referencia il Nome, la Città e lo Stato. L’applicazione X conterà 7 DET; l’applicazione Z conterà 3 DET.

- ❑ Contare un DET per ogni singolo dato richiesto dall’utente per stabilire una relazione con un'altra funzione di tipo dati.

Per esempio, in un’applicazione Risorse Umane, le informazioni sull’impiegato sono mantenute su un ILF. Il tipo di lavoro dell’impiegato costituisce parte delle informazioni relative all’impiegato. Questo DET è contato perché è richiesta una relazione fra l’impiegato e i tipi di lavoro che esistono nell’organizzazione. Questo tipo di elemento dati è chiamato *chiave esterna*.

Per esempio, in un’applicazione Object Oriented (OO), l’utente richiede una associazione tra classi di oggetti, che sono stati identificati come ILF distinti. Il nome della sede è un DET nell’EIF Sede. Il nome della sede è richiesto per elaborare le informazioni relative all’impiegato: di conseguenza, viene anche contato come un DET nell’ambito dell’ILF Impiegato.

- ❑ Riesaminare gli attributi correlati per determinare se questi sono raggruppati e conteggiati come un singolo DET oppure se sono conteggiati come DET multipli; il raggruppamento dipenderà da come i

processi elementari utilizzano gli attributi nell'ambito dell'applicazione.

Per esempio, un numero di conto che è memorizzato su più campi fisici è contato come un solo DET.

Per esempio, una “before” o “after image” di un gruppo di 10 campi mantenuti per scopi di controllo, saranno contate come un DET per la “before image” (tutti i 10 campi), e come un DET per la “after image” (tutti i 10 campi), per un totale di 2 DET.

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

**RET: Regole** Per conteggiare un Tipo di Elemento Record (RET) per una funzione di tipo dati, devono essere svolte le seguenti attività:

- ❑ Conta un RET per ogni funzione di tipo dati (cioè, di default ogni funzione di tipo dati ha un sottogruppo di DET che deve essere conteggiato come un RET).
- ❑ Conta un RET aggiuntivo per ognuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contiene più di un DET:
  - entità associativa con attributi non-chiave,
  - entità sotto-tipo (ulteriore rispetto al primo dei sotto-tipi), e
  - entità attributiva, in una relazione 1-1 non obbligatoria.

**Nota:** Una relazione 1-1 obbligatoria riflette una relazione tra due entità in cui ciascuna è legata a una e una sola istanza dell'entità collegata.

Per esempio, nell'applicazione Risorse Umane, le informazioni relative ad un impiegato sono aggiunte inserendo alcune informazioni generali. Oltre alle informazioni generali, l'impiegato può essere stipendiato o retribuito a ore.

L'utente ha deciso che un impiegato deve essere o stipendiato o retribuito a ore. Entrambe le tipologie possono avere informazioni riguardanti i collaboratori. Per questo esempio, ci sono tre sottogruppi o RET, come mostrato di seguito:

- Impiegato stipendiato; include le informazioni generali
- Impiegato ad ore; include le informazioni generali
- Dipendente di un impiegato

**Nota:** Se non si dispone di un modello dati, cerca gruppi di dati ripetuti al fine di individuare i RET.

**Nota:** Ci sono due tipi di sottogruppi: opzionali ed obbligatori.

I *sottogruppi opzionali* sono quelli in cui l'utente ha la possibilità di utilizzare uno o nessuno dei sottogruppi nel corso di un processo elementare che aggiunge o crea un'istanza di dati.

*I sottogruppi obbligatori* sono quelli in cui l'utente deve utilizzarne almeno uno durante un processo elementare che aggiunge o crea un'istanza dei dati.

## Determinazione della complessità e del contributo

La complessità funzionale per ciascuna funzione di tipo dati deve essere determinata seguendo i seguenti passi:

Passo	Azione
1	Identificare e contare i RET e DET, la relativa complessità e le regole per il conteggio del relativo contributo usando le regole che iniziano a pagina 6-5.
2	La complessità funzionale di ciascuna funzione di tipo dati deve essere determinata sulla base del numero di RET e DET in conformità con la seguente matrice:

	1 - 19 DET	20 - 50 DET	51 o più DET
1 RET	Bassa	Bassa	Media
2 - 5 RET	Bassa	Media	Alta
6 o più RET	Media	Alta	Alta

Per esempio, una funzione con 51 DET and 2 RET si traduce in una complessità funzionale alta.

3	La dimensione funzionale di ciascuna funzione di tipo dati deve essere determinata usando il tipo e la relativa complessità funzionale in accordo con le seguenti tabelle.
---	--

Tabella di conversione per gli ILF: usare la seguente tabella per assegnare una dimensione funzionale a ciascun ILF.

Valore Complessità Funzionale	Function Point
Bassa	7
Media	10
Alta	15

Tabella di conversione per gli EIF: usare la seguente tabella per assegnare una dimensione funzionale a ciascun EIF.

Valore Complessità Funzionale	Function Point
Bassa	5
Media	7
Alta	10

Per esempio, una complessità alta per un EIF si traduce in 10 function point.

*Continua alla pagina seguente*



Passo	Azione
4	<p>È possibile calcolare il contributo degli ILF e EIF alla dimensione funzionale.</p> <p>Per esempio, la seguente tabella illustra il calcolo per un ILF ad complessità alta, due EIF a media complessità ed un EIF ad alta complessità.</p>

Tipo Funzione	Complessità Funzionale			Totale Complessità	Totale Tipi Funzione
<b>ILF</b>	0	Bassa	X 7 =	0	15
	0	Media	X 10 =	0	
	1	Alta	X 15 =	15	
<b>EIF</b>	0	Bassa	X 5 =	0	24
	2	Media	X 7 =	14	
	1	Alta	X 10 =	10	

In questo esempio non ci sono ILF a complessità basse o medie, pertanto la dimensione totale degli ILF è pari a 15. Per gli EIF non ci sono complessità basse, bensì 2 EIF a media complessità (14 punti) e uno ad alta complessità (10 punti), per un totale degli EIF pari a 24.

La dimensione funzionale include il totale complessivo per tutti i tipi di funzione. I contributi per gli ILF ed EIF possono essere sommati alla tabella che elenca tutti i tipi funzione. L'Appendice A riporta una tabella per registrare i totali di tutti i tipi funzione.

## Suggerimenti di Ausilio per il Conteggio

I seguenti suggerimenti possono essere di ausilio nell'applicare le regole per il conteggio di ILF ed EIF.

Tali suggerimenti *non sono* regole e non dovrebbero essere usati come regole.

- I dati sono un gruppo logico che soddisfa specifici requisiti utente?
  - Una applicazione può usare un ILF o EIF in vari processi, ma l'ILF o l'EIF viene contato una sola volta.
  - Un file logico (LF) non può essere contato sia come ILF che come EIF per la stessa applicazione. Se il gruppo di dati soddisfa entrambe le regole, contarlo come un ILF.
  - Se il gruppo di dati non è stato contato come un ILF o EIF di per sé, contare i suoi elementi dati come DET per l'ILF o l'EIF che comprende quel gruppo di dati.
  - Non supporre che un file fisico, una tabella o una classe di oggetti, coincida con un file logico quando i dati sono visti logicamente dal punto di vista dell'utente.
  - Sebbene alcune tecniche di memorizzazione come le tabelle in un DBMS relazionale o in un file sequenziale siano in stretta corrispondenza con gli ILF o EIF, non supporre che vi sia sempre una relazione fisico-logica uno-ad-uno (1:1).
  - Non supporre che tutti i file fisici debbano essere contati o inclusi come parte di un ILF o EIF.
- Dove sono mantenuti i dati? All'interno o all'esterno del confine dell'applicazione?
  - Considerare il diagramma di flusso.
  - Nella scomposizione funzionale del processo, identificare i punti in cui ci sono interfacce con l'utente o con altre applicazioni.
  - Considerare il diagramma dei processi per ricavare suggerimenti.
  - Attribuire gli ILF modificati da più di un'applicazione a ciascuna applicazione nel momento in cui è misurata.
  - Considerare solo i DET utilizzati da ciascuna applicazione oggetto di conteggio per dimensionare l'ILF/EIF.
- I dati in un ILF sono mantenuti attraverso un processo elementare dell'applicazione?
  - Una applicazione può usare un ILF o EIF più volte, ma si deve contare una sola volta l'ILF o l'EIF.
  - Un processo elementare può mantenere più di un ILF.
  - Considerare il diagramma dei processi per ricavare suggerimenti.

---

### Misurare le funzioni di tipo transazionale

---

**Introduzione** Questo capitolo fornisce la guida per l'applicazione delle regole per la misurazione degli input esterni (EI, External Input), output esterni (EO, External Output) e interrogazioni esterne (EQ, External Inquiry).

Una funzione di tipo transazionale è un processo elementare che fornisce funzionalità all'utente per l'elaborazione di dati. Una funzione di tipo transazionale è un input esterno, un output esterno o un'interrogazione esterna.

Per esempi che illustrano l'applicazione di tali regole, si rimanda alla Parte 4 – Esempi.

**Contenuti** Questo capitolo include le seguenti sezioni:

Argomento	Pagina
<b>Definizione e intento primario: EI, EO ed EQ</b>	<b>7-3</b>
External Input (EI)	7-3
External Inquiry (EQ)	7-3
External Output (EO)	7-3
Sommario delle funzionalità svolte da EI, EO ed EQ	7-4
Definizione dei termini utilizzati	7-5
Sommario del trattamento logico usato da EI, EO ed EQ	7-8
<b>Funzioni di tipo transazionale: regole di conteggio</b>	<b>7-9</b>
Procedure di conteggio	7-9
Identificare ciascun processo elementare	7-10
Determinare l'univocità dei processi elementari	7-11
<b>Classificare ciascun processo elementare</b>	<b>7-13</b>
Classificare come un EI	7-13
Classificare come un EO	7-13
Classificare come un'EQ	7-13

*Continua a pagina seguente*

<b>Argomento</b>	<b>Pagina</b>
Definizione e regole per complessità e contributo	7-14
FTR: definizione	7-14
FTR: regola	7-14
DET: definizione	7-14
DET: regole	7-14
EI: linee guida su complessità e contributo	7-15
FTR: guida per EI	7-15
DET: guida per EI	7-15
EO/EQ: linee guida su complessità e contributo	7-18
FTR: guida per EQ	7-18
FTR: guida per EO	7-18
DET: guida comune per EO ed EQ	7-18
Determinazione della complessità e del contributo	7-21
<b>Suggerimenti di ausilio per il conteggio di EI, EO ed EQ</b>	<b>7-23</b>
Suggerimenti aggiuntivi per contare EO ed EQ	7-25

## Definizione e intento primario: EI, EO ed EQ

Questa sezione include la definizione e l'intento primario di un input esterno (EI), di un output esterno (EO) e di una interrogazione esterna (EQ). Sono inoltre definiti i termini usati all'interno delle definizioni stesse, con alcuni esempi.

### External Input (EI)

Un input esterno (EI, External input) è un processo elementare che elabora dati o informazioni di controllo che provengono 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.

### External Inquiry (EQ)

Un'interrogazione esterna (EQ, External Inquiry) è un processo elementare che invia dati o informazioni di controllo all'esterno del confine dell'applicazione. L'intento primario di un'EQ è presentare informazioni ad un utente attraverso il reperimento di dati o informazioni di controllo. 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.

### External Output (EO)

Un output esterno (EO, External Output) è un processo elementare che invia dati o informazioni di controllo all'esterno del confine dell'applicazione e include ulteriori elaborazioni rispetto all'interrogazione esterna (EQ). 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 e/o alterare il comportamento del sistema.

## Sommario delle funzionalità svolte da EI, EO ed EQ

Come indicato nella Parte 1, la principale differenza tra le funzioni di tipo transazionale è rappresentata dal loro intento primario. La seguente tabella riassume le modalità di trattamento logico possibili per ciascuna funzione di tipo transazionale ed il loro intento primario.

<i>Funzionalità:</i>	Tipo Funzione Transazionale:		
	<b>EI</b>	<b>EO</b>	<b>EQ</b>
Alterare il comportamento del sistema	IP	P	V
Mantenere uno o più ILF	IP	P	V
Presentare informazioni a un utente	P	IP	IP

Legenda:

- IP l'intento primario del tipo di funzione transazionale,
- P una possibile funzionalità del tipo di funzione transazionale, ma non rappresenta l'intento primario ed è a volte presente,
- V la funzionalità è vietata al tipo di funzione transazionale

La differenza principale tra i tipi di funzione transazionali è data dal loro intento primario.

Tra le differenze riscontrabili tra EO ed EQ c'è che un EO può alterare il comportamento del sistema o mantenere uno o più ILF quando esegue l'intento primario di presentare informazioni all'utente. Ulteriori differenze sono individuate nella sezione sottostante, che riassume le modalità di trattamento logico usate da ciascuna funzione di tipo transazionale.

## Definizione dei termini utilizzati

Il seguente paragrafo include le definizioni riportate nella Parte 1 necessarie per determinare gli EI, gli EO e le EQ.

### Processo Elementare

Un *processo elementare* è la più piccola unità di attività che è significativa per l'utente.

Per esempio, un utente richiede la possibilità di aggiungere altri tipi di informazioni sull'impiegato (per es., indirizzo, informazioni sul salario e i suoi collaboratori), ma la più piccola unità di attività è quella di aggiungere un nuovo impiegato. In questo esempio, aggiungere un impiegato (senza aggiungere l'indirizzo, informazioni sul salario e i collaboratori) non soddisfa tutte le richieste. Una manutenzione successiva si preoccuperà di gestire informazioni sul salario e/o sui collaboratori in modo indipendente dalla gestione dell'impiegato.

### Informazioni di Controllo

Le *Informazioni di Controllo* sono dati che influenzano un processo elementare dell'applicazione oggetto di conteggio. Esse specificano *quale*, *quando* o *come* un dato deve essere elaborato.

Per esempio, qualcuno nel dipartimento del personale stabilisce cicli di pagamento per pianificare quando devono essere pagati gli impiegati di ciascuna sede. Il ciclo di pagamento, o schedulazione, contiene informazioni temporali che condizionano il momento in cui avverrà il processo elementare di pagamento degli impiegati.

### Mantenuto

Il termine *mantenuto* denota la capacità di aggiungere, modificare o cancellare dei dati tramite un processo elementare.

Gli esempi includono, ma non sono limitati a aggiungere, modificare, cancellare, popolare, rivedere, aggiornare, assegnare, e creare.

### Utente

Un *utente* è la persona o cosa che comunica o interagisce con il software in qualsiasi momento.

Per esempio: sono da considerare utenti sia le persone del dipartimento Risorse Umane (RU) che interagiscono con l'applicazione per gestire le informazioni relative all'impiegato, sia l'applicazione Benefici che interagisce con l'applicazione RU per ricevere le informazioni relative ai collaboratori del dipendente.

### Significativo

Riconoscibile dall'utente e che soddisfa i requisiti utente funzionali.

### Trattamento logico

Il *Trattamento logico* è definito come l'insieme specificatamente richiesti dall'utente per completare un processo elementare quali validazioni, algoritmi o calcoli e leggere o mantenere dati. Questi requisiti possono includere le seguenti azioni:

1. Eseguire validazioni.

Per esempio, quando si aggiunge un nuovo impiegato nell'organizzazione,

il processo convalida le informazioni (DET) che si stanno inserendo.

2. Eseguire formule matematiche e calcoli.

Per esempio, il processo che presenta una lista di tutti gli impiegati di un'organizzazione include il calcolo del numero totale degli impiegati salariati, quello degli impiegati a ore e di tutti gli impiegati.

3. Convertire valori equivalenti.

Per esempio, l'età dell'impiegato viene confrontata con l'età media.

4. Filtrare e selezionare dati utilizzando criteri specifici per confrontare insiemi multipli di dati.

Per esempio, per generare una lista di impiegati che svolgono un certo lavoro, un processo elementare utilizza il codice lavoro presente in ogni assegnazione di lavoro per selezionare e ottenere una lista di tutti gli impiegati con quella particolare assegnazione.

5. Analizzare condizioni per determinare quali siano applicabili.

Per esempio, il trattamento logico utilizzato dal processo elementare che aggiunge un nuovo impiegato sarà influenzato dalla condizione che esso sia stipendiato o pagato a ore. L'inserimento di DET (e il risultante trattamento logico) basato su una differente scelte (pagamento dello stipendio o su base oraria) in questo esempio è parte di un processo elementare.

6. Aggiornare uno o più ILF.

Per esempio, il processo elementare che aggiunge un nuovo impiegato aggiorna l'ILF Impiegato per mantenere i relativi dati.

7. Referenziare uno o più ILF o EIF.

Per esempio, nell'aggiungere un nuovo impiegato viene referenziato l'EIF Valuta per usare il corretto tasso di conversione relativo a dollari US al fine di determinare la paga oraria dell'impiegato.

8. Reperire dati o informazioni di controllo.

Per esempio, per visualizzare una lista dei possibili impiegati tali informazioni sono reperite da una funzione di tipo dati.

9. Creare dati derivati trasformando dati esistenti per ottenere ulteriori dati.

Per esempio, per determinare (derivare) un numero di registrazione di un paziente (per es. ROSCA01), si concatenano i seguenti dati:

- le prime tre lettere del cognome (per es. ROS per Rossi),
- le prime due lettere del nome (per es. CA per Carlo),
- un progressivo unico a due cifre (a partire da 01).



## 10. Alterare il comportamento del sistema.

Per esempio, il comportamento del processo elementare relativo al pagamento degli impiegati risulta alterato quando si apporta una modifica alla data di pagamento, da “ogni quindicesimo e ogni ultimo giorno del mese” a “ogni venerdì”; il risultato sarà quello di avere 26 periodi di pagamento all’anno rispetto a 24.

## 11. Preparare e presentare informazioni all’esterno del confine.

Per esempio, visualizzare una lista di impiegati all’utente.

## 12. Accettare dati o informazioni di controllo che entrano nel confine dell’applicazione.

Per esempio, un utente inserisce varie informazioni per aggiungere un nuovo ordine da parte di un cliente nel sistema.

## 13. Ordinare o riorganizzare dati.

Questa forma di trattamento logico non ha impatto sull’identificazione del tipo o contributo all’unicità di un processo elementare; ad es:  
l’orientamento dei dati non costituisce unicità di trattamento.

Per esempio, la lista degli impiegati è ordinata alfabeticamente oppure secondo la sede di appartenenza.

Per esempio, in una schermata di inserimento ordini, le informazioni di testata dell’ordine sono sistemate nella parte alta dello schermo e i dettagli sono sistemati nella parte bassa.

Un singolo processo elementare può includere varie alternative o occorrenze delle suddette azioni.

Per esempio: validazioni, filtri, riordinamenti, ecc.

## Sommario del trattamento logico usato da EI, EO ed EQ

La tabella seguente riassume quali forme di trattamento logico possono essere eseguite dagli EI, EO ed EQ. Per ciascun tipo di funzione transazionale, alcuni tipi di trattamento logico devono assolutamente essere utilizzati per il raggiungimento dell'intento primario previsto per tale tipo di funzione. Le 13 azioni di per sé non individuano processi elementari unici.

Tipo di Trattamento Logico	Tipo di Funzione Transazionale		
	EI	EO	EQ
1. Eseguire validazioni	p	p	p
2. Eseguire formule matematiche e calcoli	p	o*	v
3. Convertire valori equivalenti	p	p	p
4. Filtrare e selezionare dati utilizzando criteri specifici per confrontare insiemi multipli di dati	p	p	p
5. Analizzare condizioni per determinare quali siano applicabili	p	p	p
6. Aggiornare almeno un ILF	o*	o*	v
7. Referenziare almeno un ILF o EIF	p	p	o
8. Reperire dati o informazioni di controllo	p	p	o
9. Creare dati derivati	p	o*	v
10. Alterare il comportamento del sistema	o*	o*	v
11. Preparare e presentare informazioni all'esterno del confine	p	o	o
12. Accettare dati o informazioni di controllo che entrano nel confine dell'applicazione	o	p	P
13. Ordinare o riorganizzare un insieme di dati	p	p	p

Legenda:

- o** è **obbligatorio** che il tipo di funzione utilizzi questa forma di trattamento logico,
- o\*** è **obbligatorio** che il tipo di funzione utilizzi almeno una di queste forme (o\*) di trattamento logico
- p** è **possibile** che il tipo di funzione utilizzi questa forma di trattamento logico, ma ciò non è obbligatorio
- v** è **vietato** che il tipo di funzione utilizzi questa forma di trattamento logico.

## Funzioni di tipo transazionale: regole di conteggio

Questa sezione definisce le regole che si applicano nel conteggio degli EI, degli EO e delle EQ.

### Procedure di conteggio per le funzioni di tipo transazionale

Le procedure di conteggio di EI, EO ed EQ devono includere i seguenti passi:

Passo	Azione
1	Identificare ciascun processo elementare.
2	Determinare l'univocità dei processi.
3	Classificare ciascuna funzionalità transazionale come un Input Esterno (EI), Output Esterno (EO) o una Interrogazione Esterna (EQ).
4	Determinare la complessità funzionale per ciascuna funzione di tipo transazionale e il suo contributo alla dimensione funzionale.

Le regole della Parte 1 sono illustrate in dettaglio nei seguenti paragrafi.

## Identificare ciascun processo elementare

Per identificare ciascun processo elementare, è necessario eseguire le seguenti attività:

Comporre e/o decomporre i Requisiti Funzionali Utente nella più piccolo unità di attività, che soddisfa tutti i seguenti punti:

- ❑ è significativa per l'utente,

Per esempio, il requisito utente funzionale richiede la possibilità di aggiungere i dati di un nuovo impiegato nell'applicazione.

- ❑ costituisce una transazione completa

Per esempio, la definizione dell'utente di 'impiegato' include le informazioni sullo stipendio e sul dipendente stesso. Se il numero dei dipendenti è maggiore di zero, aggiungere un impiegato dovrà includere le informazioni sul dipendente. In questo esempio, aggiungere un impiegato (senza aggiungere l'indirizzo, stipendio e informazioni sul dipendente) non incontra tutti i criteri. Altri sistemi possono gestire la manutenzione dello dei dati su stipendio e/o informazioni sul dipendente in modo indipendente dai dati dell'impiegato..

- ❑ è autonoma, e

Per esempio, il processo di 'aggiunta' non è autonomo fintanto che tutte le informazioni obbligatorie siano inserite e tutti i passi elaborativi siano stati effettuati, es: validazioni, calcoli, aggiornamento degli ILF.

- ❑ lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale.

Per esempio, i requisiti utente per aggiungere i dati di un impiegato includono l'impostazione delle informazioni sullo stipendio e sul dipendente. Se tutte le informazioni sui dipendenti non vengono aggiunte, un dipendente non è stato ancora creato. L'aggiunta di alcune delle informazioni da sole lascia l'attività di aggiungere un dipendente in uno stato non coerente. Se sia il salario dei dipendenti e le informazioni del dipendente vengono aggiunte, allora questa unità di attività può risultare completata e il processo è lasciato in uno stato di coerenza funzionale.

Identificare un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.

## Determinare l'univocità dei processi elementari

Per determinare l'unicità dei processi elementari, è necessario eseguire le seguenti attività:

Dopo il confronto con un altro Processo Elementare (EP, Elementary Process) già identificato, contare i due EP simili come lo stesso EP se questi:

- ❑ utilizzano lo stesso insieme di DET, e
- ❑ utilizzano lo stesso insieme di FTR, e
- ❑ richiedono lo stesso insieme di trattamenti logici per completare il processo elementare.

**Nota:** Un processo elementare può includere variazioni minori di DET o FTR così come alternative multiple, variazioni o occorrenze del trattamento logico sottostante.

**Nota:** Una volta confrontati i due processi elementari e stabilito che essi contengono differenti DET, FTR o trattamenti logici, questi si identificano come processi elementari distinti se sono specificati dall'utente quali distinti requisiti funzionali .

**Nota:** La prova di univocità di cui sopra è destinato ad essere usato come mezzo per confrontare due EP già identificati e non quale giustificazione per suddividere un singolo EP in due come conseguenza delle variazioni. Suddividere un EP in due sulla base di variazioni indicherebbe che le regole per identificare un EP non sono state soddisfatte.

Per esempio, quando un EP 'Aggiungi Dipendente' richiede DET aggiuntivi per tenere conto tanto degli indirizzi di impiegati Europei quanto Statunitensi (codice di avviamento postale / CAP, paese / stato, paese numero di telefono e codice città). L'EP non viene suddiviso in due EP per spiegare le lievi differenze nei formati di indirizzo di un dipendente. Il PE è ancora 'Aggiungi Dipendente', e ci sono varianti nel trattamento logico e nei DET di cui tenere conto considerando i campi indirizzo e numero di telefono.

Per esempio, quando un EP 'Aggiungi Dipendente' sia stato identificato, non va diviso in due EP per tenere conto del fatto che un impiegato possa o non possa avere dipendenti. L'EP è ancora 'Aggiungi Dipendente', e non vi è variazione nel trattamento logico e nei DET relativi ai dipendenti.

Per esempio, quando i requisiti utente funzionali (FUR) specificano la necessità di avere due report simili (come quando il Report 1 contiene il nome del cliente, ID cliente e l'indirizzo e il Report 2 contiene invece Nome cliente, ID cliente, indirizzo e numero di telefono), i report sono identificati come EP separati poiché i FUR richiedono necessariamente DET diversi. I report non sono combinati in un singolo EP semplicemente perché hanno DET simili.

- ❑ Un processo elementare con molteplici forme multiple di trattamento logico non va diviso in più processi elementari distinti. Se un processo elementare viene impropriamente suddiviso, non sarebbero più soddisfatti i criteri (sopra elencati) per un processo elementare.

## Classificare ciascun processo elementare

Classificare ciascun processo elementare come un External Input (EI), un External Output (EO) o un'External Inquiry (EQ) in base al suo intento primario.

L'intento primario di un processo elementare deve essere identificato tra uno dei seguenti punti:

- ☐ alterare il comportamento del sistema,
- ☐ mantenere uno o più ILF,
- ☐ presentare informazioni all'utente.

Le forme di trattamento logico richieste per completare il processo elementare sono identificate dalla lista presentata a pagina 7-8.

### Classificare come un EI

Un EI ha l'intento primario di

- ☐ mantenere uno o più ILF, o
- ☐ alterare il comportamento del sistema,

e include nel trattamento logico la possibilità di ricevere dati o informazioni di controllo dall'esterno del confine dell'applicazione.

### Classificare come un EO

Un EO ha l'intento primario di presentare informazioni all'utente,

e include almeno una delle seguenti forme di trattamento logico:

- ☐ vengono eseguiti calcoli matematici,
- ☐ sono mantenuti uno o più ILF,
- ☐ sono create dati derivati, oppure \*
- ☐ risulta alterato il comportamento del sistema.

**Nota:** \* Campi calcolati rappresentano una forma di dati derivati, sebbene i dati derivati possano essere altresì creati senza eseguire calcoli.

### Classificare come un'EQ

Un'EQ ha l'intento primario di presentare informazioni all'utente, e:

- ☐ riferisce una funzione di tipo dati per reperire dati o informazioni di controllo, e
- ☐ non soddisfa i criteri per essere classificato come un EO.

## Definizioni e regole per complessità e contributo

Il numero di EI, EO ed EQ e le loro rispettive complessità funzionali determinano il contributo delle funzioni transazionali alla dimensione funzionale.

Si assegna ad ogni EI, EO e EQ identificato una complessità funzionale basata sul numero di *tipi file referenziati* (FTR – File Type Referenced) e *tipi elemento dati* (DET – Data Element Type).

Nell'applicare le regole per FTR/DET ai singoli tipi di funzioni transazionali, è importante altresì riconoscere le differenze nelle funzioni che ciascun tipo può eseguire in accordo con le proprie regole di classificazione (es: per regola, un EQ non può aggiornare un FTR; di conseguenza, la guida per contare un FTR per l'aggiornamento di un ILF non si applica). Per identificare le funzioni che un tipo funzione può eseguire, ci si riferisca alla tabella a pagina 7-8.

**FTR: definizione** Un *tipo file referenziato* (FTR – File Type Referenced) è una funzione dati letta e/o mantenuta da una funzione transazionale.

Un *FTR* include:

- un ILF letto o mantenuto da una funzione transazionale, oppure
- un EIF letto da una funzione transazionale

**FTR: regola** Contare un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)-

**DET: definizione** Un *tipo elemento dati* (DET – Data Element Type) è un attributo unico, riconoscibile dall'utente, non ripetuto.

**DET: regole** Per contare i DET di una funzione di tipo transazionale, è necessario eseguire le seguenti attività.

- ☐ Riesaminare ogni elemento che attraversa (entrando e/o uscendo) il confine.
- ☐ Contare un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.
- ☐ Contare un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli.
- ☐ Contare un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.



- ❑ Non contare i seguenti elementi come DET.
  - ❑ Costanti quali titolo del report, identificativi dello schermo, intestazione delle colonne o descrizione degli attributi.
  - ❑ Notazioni generate automaticamente dall'applicazione quali data e orario.
  - ❑ Numeri di pagina variabili o informazioni riguardanti il posizionamento (ad esempio "Righe da 37 a 54 di 211").
  - ❑ Aiuti navigazionali quali la possibilità di movimento all'interno di una lista usando "precedente", "prossimo", "primo", "ultimo" e oggetti grafici equivalenti.
  - ❑ Attributi generati all'interno del confine da una funzione transazionale e salvati all'interno di un ILF che non escono dal confine stesso.
  - ❑ Attributi reperiti o referenziati da un ILF o EIF necessari all'elaborazione, ma che non escono dal confine.

**EI: linee guida su complessità e contributo**

Questa sezione fornisce una guida da usare per determinare la complessità ed il contributo funzionale degli EI.

**FTR: guida per EI**

Riconoscendo che un EI deve sia aggiornare un ILF o controllare il comportamento del sistema, la seguente guida si applica nel conteggio degli FTR.

- ❑ Contare un FTR per ogni ILF mantenuto.
- ❑ Contare un FTR per ogni ILF o EIF letto.
- ❑ Contare un solo FTR per ogni ILF che è sia mantenuto che letto.

**DET: guida per EI**

Riconoscendo che un EI deve sia aggiornare un ILF o controllare il comportamento del sistema, la seguente guida si applica nel conteggio dei DET.

- ❑ Riesaminare ogni elemento che attraversa (entrando e/o uscendo) il confine.
- ❑ Contare un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.

Per esempio, nome lavoro e retribuzione sono due campi che l'utente fornisce quando inserisce un nuovo lavoro.

- ❑ Contare un solo DET per funzione di tipo transazionale per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli.  
Per esempio, se un utente tenta di aggiungere un impiegato già esistente nell'applicazione Risorse Umane, il sistema produce un messaggio di errore e il campo errato viene evidenziato. Contare un solo DET per tutti i messaggi di sistema che indicano condizioni di errore, che confermano che l'elaborazione
- ❑ Contare un solo DET per funzione di tipo transazionale per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.  
Per esempio, se un utente può avviare l'inserimento di un impiegato cliccando sul tasto OK oppure premendo un tasto funzionale, contare un solo DET per la capacità di avviare il processo.
- ❑ Non contare i seguenti elementi come DET.
  - ❑ Costanti quali titolo del report, identificativi dello schermo, intestazione delle colonne o descrizione degli attributi.
  - ❑ Notazioni generate automaticamente dall'applicazione quali data e orario.
  - ❑ Numeri di pagina variabili o informazioni riguardanti il posizionamento (ad esempio "Righe da 37 a 54 di 211").
  - ❑ Aiuti navigazionali quali la possibilità di movimento all'interno di una lista usando "precedente", "prossimo", "primo", "ultimo" e oggetti grafici equivalenti.
  - ❑ Attributi generati all'interno del confine da una funzione transazionale e salvati all'interno di un ILF che non escono dal confine stesso.  
Per esempio, al fine di mantenere la paga oraria in dollari US per gli impiegati ad ore che lavorano in paesi esteri con altre valute, la paga oraria in valuta è fornita dall'utente. Durante l'elaborazione di tutti gli attributi necessari per aggiungere un impiegato, un tasso di conversione è reperito dal sistema di gestione delle valute per calcolare la paga oraria in dollari US. La paga oraria calcolata in dollari US è mantenuta nell'ILF Impiegato come risultato dell'inserimento di un nuovo impiegato. La paga oraria in dollari US non sarà contata come un DET per l'EI perché non attraversa il confine dell'applicazione, ma è calcolata internamente (cioè è un dato derivato).
  - ❑ Attributi reperiti o referenziati da un ILF o EIF necessari alla elaborazione, ma che non escono dal confine.  
Per esempio, quando un ordine del cliente è aggiunto nel sistema, il prezzo unitario di ciascun articolo ordinato è automaticamente reperito e memorizzato nel record di fatturazione. Il prezzo unitario non è contato come un DET per l'EI in quanto non attraversa il confine dell'applicazione quando l'utente aggiunge l'ordine del cliente.

<b>EO/EQ: linee guida su complessità e contributo</b>	<p>Questa sezione fornisce una guida su FTR e DET da usare per determinare la complessità ed il contributo funzionale degli EO ed EQ.</p>
<b>FTR: guida per EQ</b>	<p>Riconoscendo che un'EQ non può aggiornare un ILF, la seguente guida si applica nel conteggio degli FTR per EQ:</p> <ul style="list-style-type: none"><li>□ Contare un FTR per ogni ILF o EIF letto</li></ul>
<b>FTR: guida per EO</b>	<p>Riconoscendo che un EQ può aggiornare un ILF, la seguente guida aggiuntiva si applica nel conteggio degli FTR per EO:</p> <ul style="list-style-type: none"><li>□ Contare un FTR per ogni ILF o EIF letto</li><li>□ Contare un FTR per ogni ILF mantenuto</li><li>□ Contare un solo FTR per ogni ILF che è sia mantenuto che letto</li></ul>
<b>DET: guida comune per EO ed EQ</b>	<p>La seguente guida si applica per contare i DET sia per EO che per EQ.</p> <ul style="list-style-type: none"><li>□ Riesaminare ogni elemento che attraversa (entrando e/o uscendo) il confine</li><li>□ Contare un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.</li></ul> <p><u>Per esempio (EO/EQ)</u>, per generare una lista di impiegati, il nome dell'impiegato è un campo che l'utente fornisce per individuare gli impiegati da elencare.</p> <p><u>Per esempio (EO/EQ)</u>, contare come un solo DET un messaggio testuale che può essere una singola parola, un periodo o una frase – una riga o un paragrafo incluso in un prospetto per indicare un commento esplicativo.</p> <p><u>Per esempio (EO/EQ)</u>, un numero di conto o una data, fisicamente memorizzato/a in campi multipli è contato/a come un solo DET quando è richiesto/a come singola informazione.</p> <p><u>Per esempio (EO/EQ)</u>, un diagramma a torta potrebbe avere un'etichetta di categoria e un valore numerico corrispondente in un output grafico. Contare due DET - uno per designare la categoria ed uno per il valore numerico.</p> <ul style="list-style-type: none"><li>□ Contare un solo DET per funzione di tipo transazionale per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli</li></ul> <p><u>Per esempio (EO/EQ)</u>, se un utente cerca di richiedere una lista, ma non ha accesso alle informazioni, contare un DET per la risposta del sistema.</p>

- ❑ Contare un solo DET per funzione di tipo transazionale per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.

Per esempio (EO/EQ), se l'utente può avviare la produzione di unprospetto cliccando sul bottone OK o premendo un tasto funzionale, contare un solo DET per la capacità di avviare il processo.

- ❑ Non contare i seguenti elementi come DET:

- ❑ Costanti quali titolo del report, identificativi dello schermo, intestazione delle colonne o descrizione degli attributi

Per esempio (EO/EQ), le costanti quali titoli dei prospetti, gli identificativi delle schermate o dei pannelli, le intestazioni delle colonne e i titoli dei campi.

- ❑ Notazioni generate automaticamente dall'applicazione quali attributi di data e ora.

Per esempio (EO/EQ), i campi data e ora se mostrati.

- ❑ Numeri di pagina variabili o informazioni riguardanti il posizionamento (ad esempio "Righe da 37 a 54 di 211").

Per esempio (EO/EQ), numeri di pagina che appaiono in un prospetto.

- ❑ Aiuti navigazionali quali la possibilità di movimento all'interno di una lista usando "precedente", "prossimo", "primo", "ultimo" e oggetti grafici equivalenti.

Per esempio (EO/EQ), pulsanti 'precedente' e 'successivo' che permettono all'utente di navigare e scorrere una lista di valori.

- ❑ Attributi generati all'interno del confine da una funzione transazionale e salvati all'interno di un ILF che non escono dal confine stesso.

Per esempio (EO), quando si stampa un cedolino, un indicatore di stato viene aggiornato sull'ILF Impiegato per indicare che il cedolino è stato stampato. Non contare questo indicatore come un DET poiché non attraversa il confine dell'applicazione.

**Nota:** Per regola, un EQ non può aggiornare un ILF, pertanto questa regole non si applica.

- ❑ Attributi reperiti o referenziati da un ILF o EIF necessari alla elaborazione ma che non escono dal confine.

Per esempio (EO/EQ), quando si crea un prospetto di 'Insoluti', il campo 'data pagamento dovuto' è referenziato per determinare se il conto è scaduto, ma non appare nel prospetto. Non contare il campo 'data pagamento dovuto' come un DET. poiché non attraversa il confine dell'applicazione

## Determinazione della complessità e del contributo

La complessità funzionale di ciascuna funzione transazionale dovrà essere determinate applicando i seguenti passi.

Passo	Azione
1	Per identificare e contare il numero di FTR e DET, usare le regole relative alla complessità e al contributo a partire da pagina 7-14.
2	La complessità funzionale di ciascuna funzione transazionale deve essere determinate usando il numero di FTR e DET in accordo alle seguenti matrici di complessità.

### External Input (EI):

	1 a 4 DET	5 a 15 DET	16 o più DET
0 a 1 FTR	Bassa	Bassa	Media
2 FTR	Bassa	Media	Alta
3 o più FTRs	Media	Alta	Alta

### External Output (EO) e External Inquiries (EQ):

	1 a 5 DET	6 a 19 DET	20 o più DET
0 a 1 FTR	Bassa	Bassa	Media
2 a 3 FTR	Bassa	Media	Alta
4 o più FTR	Media	Alta	Alta

Passo	Azione
-------	--------

- 3 La dimensione funzionale di ciascuna funzione di tipo transazionale deve essere determinata usando il tipo e la complessità funzionale in accordo alle seguenti tabelle.

**External Input (EI) ed External Inquiry (EQ):**

Complessità Funzionale	Function Point
Bassa	3
Media	4
Alta	6

**External Output (EO):**

Complessità Funzionale	Function Point
Bassa	4
Media	5
Alta	7

## Suggerimenti di ausilio per il conteggio di EI, EO ed EQ

I seguenti suggerimenti possono essere di ausilio nell'applicare le regole per il conteggio di EI, EO ed EQ ed effettuare la misurazione della dimensione funzionale.

Tali suggerimenti *non sono* regole e non dovrebbero essere usati come regole.

- I dati sono ricevuti dall'esterno del confine dell'applicazione?
  - Considerare il diagramma di flusso.
  - Nella scomposizione funzionale del processo, individuare i punti in cui ci sono interfacce con l'utente e con altre applicazioni.
- Il processo è la più piccola unità di attività dal punto di vista dell'utente?
  - Considerare i diversi moduli cartacei e interattivi usati.
  - Riesaminare gli ILF per individuare come l'utente raggruppa le informazioni.
  - Nella scomposizione funzionale del processo, individuare i punti in cui ci sono interfacce con l'utente e con altre applicazioni.
  - Considerare cosa è descritto nel manuale del sistema.
  - Tenere presente che un input fisico o un file di transazione o una schermata possono, quando sono considerati dal punto di vista logico, corrispondere a diversi EI, EO o EQ.
  - Tenere presente che due o più input fisici o file di transazione o schermate (es: tab su una schermata) possono corrispondere a singoli EI, EO o EQ.
  - Ricordare che due o più report fisici, schermate o file batch di output possono corrispondere ad un EO/EQ se il trattamento logico è identico.
- Il processo è autonomo e lascia l'applicazione in uno stato di coerenza funzionale?
  - Rivedere gli altri EI, EO e EQ per avere una visione generale di come l'utente lavora con le informazioni.
  - Seguire il diagramma dei processi per ricavare suggerimenti.
  - Considerare cosa è descritto nel manuale del sistema.
  - Controllare la consistenza rispetto ad altre decisioni.
- Identificare l'intento primario del processo elementare prima di classificarlo come un EI, un EO o un'EQ.
- L'identificazione dei processi elementari è basata su una comprensione o interpretazione dei requisiti condivisa tra gli utenti e gli sviluppatori.
- Ciascun elemento di una scomposizione funzionale potrebbe non corrispondere ad un unico processo elementare.

- L'identificazione dei processi elementari richiede una interpretazione dei requisiti utente.
- Il trattamento logico è unico rispetto ad altri EI, EO ed EQ?
  - Identificare gli input o gli output batch sulla base del trattamento logico richiesto.
  - Una transazione eseguita in numerosi input fisici, file di transazione o schermate, ma che presenta lo stesso trattamento logico, corrisponde tipicamente ad una funzione transazionale (EI, EO o EQ).
  - Ricorda che il solo ordinare o riorganizzare un insieme di dati non rende il trattamento logico unico.
- Gli elementi dati differiscono da quelli di altri EI, EO o EQ?
  - Se gli elementi dati sembrano essere un sottoinsieme degli elementi dati di un altro EI, EO, o EQ, assicurati che siano richiesti due processi elementari dall'utente: uno per l'insieme principale degli elementi dati e uno per i sottoinsiemi.
- Conta un solo FTR per ogni ILF/EIF referenziato anche se l'ILF/EIF ha più di un RET.



## Suggerimenti aggiuntivi per contare EO ed EQ

- Il processo è la più piccola unità di attività dal punto di vista dell'utente?
  - Un EO o un'EQ possono essere attivati da un processo interno al confine dell'applicazione.

Per esempio, l'utente richiede che un prospetto di tutte le paghe degli impiegati che sono state modificate venga inviato all'area Bilancio ogni 8 ore in base ad un orologio interno.

Situazione A. Il prospetto contiene il nome dell'impiegato, il codice fiscale e la sua paga oraria, che sono tutte informazioni reperite dal file Impiegato. Questa è la più piccola unità di attività dal punto di vista utente, non contiene formule matematiche o calcoli e nessun ILF è mantenuto dal processo. Questa è un'EQ.

Situazione B. Il prospetto contiene il nome dell'impiegato, il codice fiscale e la sua paga oraria che sono tutte informazioni reperite dal file Impiegato. Il prospetto include anche la percentuale di variazione della paga del dipendente, che è calcolata dai dati presenti nel file Impiegato. Questa è la più piccola unità di attività dal punto di vista utente e nessun ILF è mantenuto dal processo. D'altra parte, siccome il processo contiene una formula matematica, questo è un EO.

- I dati derivati per un EO non devono necessariamente essere mostrati in output.

Per esempio, ogni mese è generato un prospetto contenente tutti gli impiegati che dovranno sottoporsi alla valutazione nei successivi 30 giorni. I record sono selezionati attraverso il calcolo della successiva data di valutazione che è basata sulla ultima data di valutazione dell'impiegato, che è un campo presente nel file Impiegato, e sulla data corrente incrementata di 30 giorni. Questo processo dovrebbe essere contato come un EO e non come un'EQ.

Questa pagina è lasciata intenzionalmente vuota..

---

## Indice analitico – Parte 2

---

**A**

Analisi dei Function Point (FPA)

procedure per capitolo, 2-2

Applicazione

conteggio dei function point, 4-2

**C**

Ciclo di vita, 3-1, 3-8

Confine

regole, 5-5

suggerimenti, 5-6

Classifica ciascun processo elementare, 7-13

Complessità e contributo

definizione ILF/EIF, 6-5

Complessità e Definizioni Contributo e Regole, 7-14

Complessità e procedure per il contributo

file d'interfaccia esterni (EIF), 6-8

file logici interni (ILF), 6-8

Complessità e regole per il contributo

file d'interfaccia esterni (EIF), 6-5

file logici interni (ILF), 6-5

Conteggi finali, 4-4

Conteggio iniziale, 4-4

Conteggio in Function Point non pesati

funzioni di tipo transazionale, 7-1

**D**

Dimensione Funzionale, 3-10

Data element type (DET). Vedi DET

Definizioni

complessità e contributo, 6-5

file esterno d'interfaccia (EIF) , 6-2

file logici interni (ILF), 6-2

informazioni di controllo, 6-2

mantenuto, 6-3

processo elementare, 6-3

requisiti utente funzionali (FUR), 1-3

riconoscibile dall'utente, 6-3

DET

definizione, 6-5

regole per ILF/EIF, 6-5

Determinare l'univocità dei processi elementari, 7-11

Diagramma

Tipi di conteggio, 4-4

Differenza tra ILF e EIF, 6-2

Documentazione, 3-1, 3-3

Utile, 3-9

**E**

EI: Linee guida su complessità e contributo, 7-2, 7-15

EIF. Vedi External Interface File

EI, EO e EQ

definizione e intento primario, 7-3

informazioni di controllo, 7-5

trattamento logico, 7-5

processo elementare, 7-5

suggerimenti, 7-21

EO/EQ: Linee guida su complessità e contributo, 7-

17

**Esempi**

- ILF/EIF: sottogruppi obbligatorii, 6-7
- ILF/EIF: sottogruppi opzionali, 6-7
- informazioni di controllo, 6-2
- mantenuto per ILF/EIF, 6-3
- processo elementare per ILF/EIF, 6-3
- riconoscibile dall'utente per ILF/EIF, 6-3

**External Input (EI)**

- matrice di complessità, 7-19

**External Interface File (EIF)**

- definizione, 6-2
- DET: Regole, 6-5
- differenza con ILF, 6-2
- esempio di 'informazioni di controllo', 6-2
- esempio di 'processo elementare', 6-3
- esempio di 'mantenere', 6-3
- esemp di 'riconoscibile dall'utente', 6-3
- suggerimenti di ausilio per il conteggio, 6-10
- sottogruppi obbligatori per RET, 6-7
- sottogruppi opzionali per RET, 6-7
- tabella di conversione, 6-8

**F**

- File, 6-1

- File Logico Interno. Vedi ILF.

- File d'Interfaccia Esterno. Vedi EIF.

**Funzioni di tipo dati**

- introduzione, 6-1
- panoramica, 2-5

**Funzioni di tipo transazionale**

- panoramica, 2-6

**I**

- Input Esterno. Vedi External Input (EI)

- Interrogazione Esterna. Vedi External Inquiry (EQ)

- Intento Primario (tabella), 7-4

**Informazioni di Controllo**

- esempio per ILF/EIF, 6-2

- Identificare ciascun processo elementare, 7-10

- ILF. Vedi Internal Logical File

**Internal Logical File (ILF)**

- procedure per la complessità e il contributo, 6-8
- definizione, 6-2
- DET: Regole, 6-5
- differenza con EIF, 6-2
- esemp di 'riconoscibile dall'utente', 6-3
- esempio di 'informazioni di controllo', 6-2
- esempio di 'mantenere', 6-3
- esempio di 'processo elementare', 6-3
- matrice di complessità, 6-8
- procedure per la complessità e il contributo, 6-8
- regole per la complessità e il contributo, 6-5
- sottogruppi obbligatori per RET, 6-7
- sottogruppi opzionali per RET, 6-7
- suggerimenti di ausilio per il conteggio, 6-10
- tabella di conversione, 6-8

**M****Mantenuto**

- esempio per ILF/EIF, 6-3
- definizione, 6-3

- Matrici. Vedi Matrici di complessità

**Matrici di Complessità**

- external input (EI), 7-19
- external interface file (EIF), 6-8
- internal logical file (ILF), 6-8

**O**

- Output esterni. Vedi External Output (EO)

**P****Procedure**

- complessità e contributo degli EIF, 6-8
- ILF: complessità e contributo, 6-8
- passi, 2-2
- per capitolo, 2-2

**Progetto di sviluppo**

- function point count, 4-2

**Processo Elementare**

- esempio per EIF, 6-3
- esempio per ILF, 6-3

**Progetto di manutenzione evolutiva**

- conteggio in function point, 4-2

- Punto di vista dell'utente, 3-2

**R**

- Raccogliere la documentazione disponibile, 3-1

- Record Element Type. Vedi RET

**Regole**

- confine, 5-5
- complessità e contributo per ILF/EIF, 6-5
- DET per ILF/EIF, 6-5
- ILF/EIF: sottogruppi obbligatori, 6-7
- ILF/EIF: sottogruppi opzionali, 6-7

- Regole per la complessità. Vedi Complessità e regole per il contributo

- Regole per il contributo. Vedi Complessità e regole per il contributo.

**RET**

- definizione, 6-7

**RET regole**

- sottogruppi obbligatori per ILF/EIF, 6-7
- sottogruppi opzionali per ILF/EIF, 6-7

**Requisiti Utente Funzionali (FUR)**

- definizione, 1-3

**Riconoscibile dall'utente**

- definizione, 6-3
- esempi per ILF/EIF, 6-3

**S**

- Sottogruppi obbligatori, 6-7

- Sottogruppi opzionali, 6-7

**Standard Internazionali**

- ISO/IEC 14143-1:2007, 1-2

**Suggerimenti**

- confine, 5-6
- per contare EIF, 6-10

per contare ILF, 6-10

Suggerimenti di conteggio. Vedi Suggerimenti

## **T**

Tabelle di conversione

ILF/EIF, 6-8

Tipi di conteggio

conteggi iniziali e conteggi finali, 4-4

Trattamento Logico (tabella), 7-8

## **V**

Variazione dell'ambito (scope creep), 4-4

Questa pagina è lasciata intenzionalmente vuota.

---

## **Parte 3 – Prassi di conteggio**

---

---

Questa pagina è lasciata intenzionalmente vuota.

---



---

## Parte 3 Prassi di conteggio

---

**Introduzione** La parte 3 fornisce norme di conteggio approfondite ed esempi evoluti per assistere i professionisti nella comprensione di:

- Dati di decodifica, inclusa una descrizione dei vari tipi di dati, come distinguere i dati di business e di riferimento dai dati di decodifica, e come conteggiare i dati di decodifica.
- File Logici, inclusa una descrizione dei file logici e dei processi per raggruppare correttamente i dati in file logici, per conteggiare correttamente i tipi di elementi record, e per conteggiare correttamente i tipi di elementi dati.
- Dati condivisi, inclusa una descrizione dei dati condivisi e dei processi per identificare quando i dati condivisi sono funzioni dati (file logici interni, file esterni di interfaccia) o funzioni transazioni (input esterni, output esterni, interrogazioni esterne).
- Progetti di manutenzione evolutiva e attività di manutenzione, incluse descrizioni di progetti evolutivi e manutenzioni e quelle attività riconosciute dal processo di conteggio dei function point.

**Contenuto** La Parte 3 include i seguenti capitoli:

Capitolo	Pagina
Dati di decodifica	1-3
File logici	2-1
Dati condivisi	3-1
Progetti di manutenzione evolutiva e attività di manutenzione ordinaria	4-1
Attività di conversione dati	5-1

---

Questa pagina è lasciata intenzionalmente vuota.

---

## Dati di decodifica

---

**Introduzione** Questo capitolo usa i concetti di requisiti utente funzionali e non-funzionali (descritti nella Parte 1 e nel documento “*Framework for Functional Sizing*”, [IFPUG, 2003]) per identificare i dati di decodifica e determinare come dovrebbero essere considerati.

Questo capitolo esamina in modo specifico un numero di esempi relativi ai dati di decodifica. È chiaro che ci potrebbero essere altri esempi di dati di decodifica e questi potrebbero essere esaminati in una futura versione del CPM.

**Contenuto** Questo capitolo include le seguenti sezioni:

Argomento	Pagina
Tipi di entità di dati	1-4
Metodologia	1-9
Identificare i dati di decodifica	1-10
Esaminare i dati di decodifica e le relative transazioni	1-14
Bibliografia	1-15

## Tipi di entità di dati

Un riesame dei dati delle applicazioni e del loro scopo permette di comprendere le varie categorie di entità di dati. In generale, i conteggiatori possono distinguere tre categorie di entità di dati:

- dati di business,
- dati di riferimento,
- dati di decodifica.

Le prime due categorie di entità sono di solito identificate per soddisfare i Requisiti Utente Funzionali (FUR, Functional User Requirements) e come tali queste entità saranno analizzate per essere conteggiate come file logici (vedi Parte 3, Capitolo 2).

La terza categoria di dati, indicata nel seguito di questo capitolo come ‘dati di decodifica’, esiste invece per soddisfare solitamente requisiti utente non-funzionali (per requisiti di qualità, motivi di implementazione fisica e/o ragioni tecniche) piuttosto che funzionali. Le differenti categorie di dati sono descritte nel seguito per fornire supporto alla loro corretta identificazione.

### Dati di business

I dati di business possono anche essere denominati ‘dati fondamentali dell’utente’ o ‘oggetti di business’. Questo tipo di dati rispecchia le informazioni che devono essere memorizzate e reperite da parte delle aree funzionali coinvolte dall’applicazione. I dati di business solitamente rappresentano una percentuale significativa delle entità identificate. Essi presentano la maggior parte delle seguenti caratteristiche:

#### Caratteristiche logiche

Le caratteristiche logiche dei dati di business includono:

- sono obbligatori per le operazioni relative alle aree funzionali dell’utente,
- sono riconoscibili da parte dell’utente (generalmente da parte di un utente di business),
- sono manutenibili da parte dell’utente (generalmente da parte di un utente di business),
- contengono i dati fondamentali dell’utente a supporto delle transazioni di business,
- sono alquanto dinamici – nel corso delle normali operazioni di business essi sono referenziati regolarmente e, di routine, aggiunti, modificati o cancellati,
- sono oggetto di reportistica.

- Caratteristiche fisiche** Le caratteristiche fisiche dei dati di business includono:
- hanno campi chiave e, di solito, molti attributi,
  - possono avere da zero a infiniti record.

- Esempi** Esempi di dati di business includono:
- File Clienti, File Fatture, File Impiegati, File Lavoro
  - File Lavoro, all'interno del Sistema Gestione Lavori, potrebbe prevedere degli attributi come:
    - Identificativo Lavoro,
    - Nome del lavoro,
    - Identificativo Divisione,
    - Data Inizio Lavoro, ecc.

## Dati di riferimento

Questo tipo di dati è memorizzato per supportare le regole di business per la gestione dei dati di business; per es. in un'applicazione relativa agli stipendi sarebbero i dati relativi alle aliquote fiscali per ogni scaglione retributivo e la data a partire dalla quale ogni aliquota diventa effettiva. I dati di riferimento solitamente rappresentano una piccola percentuale delle entità identificate. Essi presentano la maggior parte delle seguenti caratteristiche:

- Caratteristiche logiche** Le caratteristiche logiche dei dati di riferimento includono:
- sono obbligatori per le operazioni relative alle aree funzionali dell'utente,
  - sono identificabili da parte dell'utente (generalmente da parte di un utente di business),
  - sono di solito manutenibili da parte dell'utente (generalmente da parte di un utente amministratore del sistema),
  - sono di solito valorizzati quando l'applicazione è installata per la prima volta e sono mantenuti saltuariamente,
  - contengono informazioni a supporto di attività fondamentali per l'utente,
  - sono poco dinamici – cambiamenti occasionali in risposta alle modifiche alle aree funzionali, ai processi funzionali esterni e/o alle regole di business,
  - le transazioni che elaborano i dati di business spesso necessitano di accedere ai dati di riferimento.

**Caratteristiche fisiche**

Le caratteristiche fisiche dei dati di riferimento includono:

- hanno campi chiave e pochi attributi,
- di solito almeno un record o un limitato numero di record.

**Esempi**

Esempi di dati di riferimento includono:

- Tariffe dei Lavori, Tariffe degli Sconti, Tariffe delle Imposte, Soglia di Impostazione.
- File Tariffe dei Lavori – memorizza le informazioni relative alle tariffe pagate per ogni tipo di lavoro e lo skill richiesto per fare quel tipo di attività:
  - Tipo di Lavoro,
  - Stato, Tariffa, Data Effettiva (1:n),
  - Descrizione Skill (1:n).

**Dati di decodifica**

L'utente non sempre specifica direttamente i dati di decodifica, talvolta indicati come 'dati delle liste' o 'dati di traduzione'. In altri casi sono identificati dallo sviluppatore in risposta a uno o più requisiti non-funzionali dell'utente. I dati di decodifica forniscono una lista di valori validi che un attributo di tipo descrittivo può possedere. Tipicamente, gli attributi dei dati di decodifica sono il Codice, la Descrizione e/o altri attributi 'standard' che descrivono il codice; per es. abbreviazione standard, data di inizio validità, data di fine validità, dati di *traccia di audit*, ecc.

Quando si fa uso di codici nei dati di business, è necessario avere un mezzo di traduzione per convertire il codice in qualcosa di maggiormente riconoscibile per l'utente. Allo scopo di soddisfare i requisiti tecnici, gli sviluppatori spesso creano una o più tabelle contenenti i dati di decodifica. Logicamente, il codice e la relativa descrizione hanno lo stesso significato. Senza la descrizione il codice non può essere sempre chiaramente compreso.

Le differenze chiave fra i dati di decodifica e i dati di riferimento sono:

- Con i dati di decodifica, si può sostituire il campo codice con il campo decodifica senza cambiare il significato dei dati di business; per es. Codice dell'Aeroporto anziché Nome dell'Aeroporto, Identificativo Colore anziché Descrizione Colore.
- Con i dati di riferimento, non si può effettuare tale scambio (per es. Codice Imposta anziché Tariffa Imposta).

I dati di decodifica presentano la maggior parte delle seguenti caratteristiche.

**Caratteristiche logiche**

Le caratteristiche logiche dei dati di decodifica includono:

- sono obbligatori per l'area funzionale, ma memorizzati opzionalmente come file di dati,
- non sono di solito identificati come parte dei requisiti funzionali; sono generalmente identificati come parte della progettazione per venire incontro ai requisiti tecnici,
- sono talvolta manutenibili da parte dell'utente (di solito da parte di una persona di supporto all'utente),
- contengono informazioni per standardizzare e facilitare le attività e le transazioni di business,
- sono essenzialmente statici – sono modificati solamente in seguito a una modifica della modalità stessa di funzionamento del business,
- le transazioni di business accedono ai dati di decodifica per migliorare la facilità del data entry, migliorare la consistenza dei dati, assicurare l'integrità dei dati, ecc.
- se riconosciuti dall'utente:
  - sono talvolta considerati come un gruppo della stessa tipologia di dati,
  - potrebbero essere mantenuti usando lo stesso trattamento logico.

**Caratteristiche fisiche**

Le caratteristiche fisiche dei dati di decodifica includono:

- consistono di un campo chiave e, solitamente, di uno o due attributi soltanto,
- tipicamente hanno un numero di record stabilito,
- possono rappresentare il 50% di tutte le entità in Terza Forma Normale,
- talvolta sono de-normalizzati e collocati in una tabella fisica con altri dati di decodifica,
- possono essere implementati in modi differenti (per es. tramite applicazioni separate, dizionario dati, o scritti all'interno del codice).

**Esempi**

Esempi di dati di decodifica includono:

- Stato
  - Codice stato,
  - Nome stato.
- Tipo Pagamento
  - Codice Pagamento,
  - Descrizione Pagamento.

## Origine dei dati di decodifica

Storicamente, la motivazione per i dati di decodifica è da rinvenirsi nella necessità di risparmiare spazio memorizzando un codice piuttosto che una lunga descrizione testuale. Per facilità di manutenzione, tali codici e le relative descrizioni sono state collocate in file o tabelle per evitare di dover modificare il software quando si rendevano necessari degli aggiornamenti di tali dati.

I dati di decodifica rappresentano una proprietà di un attributo descrittivo, anche chiamati ‘meta-dati’. Esempi sono valori validi, codici-descrizioni o tabelle di traduzione. Alcuni dati di decodifica sono sviluppati per soddisfare specifici requisiti utente e contenere dati che sono all’interno del dominio dell’utente. Altri dati di decodifica possono derivare da requisiti utente per restringere il range dei valori permessi. I dati di decodifica possono anche essere creati per cercare di ridurre la richiesta di spazio su disco. I requisiti utente possono includere anche la capacità di mantenere i dati di decodifica. Questi sono tutti requisiti non-funzionali.

I dati di decodifica rappresentano pertanto un’implementazione dei requisiti non-funzionali. Ne consegue che i dati di decodifica possono influenzare la dimensione non-funzionale del prodotto software, ma *non* la sua dimensione *funzionale* [“A Framework for Functional Sizing”, IFPUG 2003].



## Metodologia

L'impatto dei dati di decodifica relativi alla dimensione non-funzionale comporta che essi non possono essere contati né possono essere contate le transazioni che mantengono i dati di decodifica.

## Introduzione

La sezione “Identificare i dati di decodifica” fornisce un procedimento passo-passo per identificare cosa sia (o non sia) un dato di decodifica. La sezione è tipicamente referenziata nel passo “Identificare i file logici”, contenuto nella Parte 3, Capitolo 2, laddove i dati di decodifica vengono esclusi. Come detto in precedenza, i dati di decodifica non sono considerati parte della dimensione funzionale. Ciò determina una serie di conseguenze. Tali conseguenze sono riassunte nella presente Guida all'Implementazione al fine di non creare ambiguità.

## Conseguenze

1	<b>Non contare i dati di decodifica come file logici</b> Una conseguenza dell'esclusione dei dati di decodifica è che i dati di decodifica non possono essere considerati ILF o EIF.
2	<b>Non contare i dati di decodifica come RET o DET</b> Una conseguenza dell'esclusione dei dati di decodifica è che i dati di decodifica non possono essere considerati RET o DET in un ILF o EIF.
3	<b>Non contare i dati di decodifica come FTR</b> Una conseguenza dell'esclusione dei dati di decodifica è che un dato di decodifica non può essere considerato un FTR nella determinazione della complessità di una funzione di tipo transazionale (EI, EO, EQ), poiché esso non è un file logico.
4	<b>Non contare le funzioni di tipo transazionale riguardanti i dati di decodifica</b> Una conseguenza del fatto che i dati di decodifica appartenga ad un'altra dimensione (la dimensione non-funzionale, contrapposta a quella funzionale) è che il loro mantenimento o le funzionalità di presentazione non si considerano al momento della misurazione della dimensione funzionale di un'applicazione.

## Identificare i dati di decodifica

I tipi di dati di decodifica riassunti nelle sezioni “Quali dati sono dati di decodifica” e “Quali dati non sono dati di decodifica” possono essere usati come ausilio pratico per determinare se un’entità sia un dato di decodifica. Alcuni criteri in parte si sovrappongono. Non appena i criteri di una delle sottosezioni siano stati soddisfatti, l’entità dovrebbe essere considerata un Dato Di Decodifica e non essere quindi conteggiata.

Gli esempi forniti non sono totalmente esaustivi e potrebbero non coprire tutti i possibili casi. In caso di dubbio, si valutino i tipi entità nell’ambito della sezione “Tipi di entità dati”.

## Quali dati sono dati di decodifica

**Introduzione** Questi sono i principali differenti tipi di dati di decodifica, che ricadono in tre aree generali:

- Dati sostitutivi: forniscono un codice e un nome o una descrizione esplicativa per un attributo di un oggetto di business (la sostituzione è una condizione sufficiente, ma non è una condizione necessaria per essere considerati Dati di Decodifica).
- Dati statici o costanti: sono dati che cambiano raramente.
- Valori validi: forniscono una lista di valori disponibili per un attributo di uno o più oggetti di business.

### Tipi di dati di decodifica

Sostitutivi	Statici o costanti	Valori validi
Codice + descrizione	Una occorrenza	Valori validi
	Dati statici	Range di valori validi
	Valori di default	

Ciascuno di questi tipi di dati di decodifica può anche comprendere ulteriori attributi, quali data iniziale e data finale per definire il periodo temporale per il quale il valore è disponibile. Possono anche essere inclusi attributi di controllo (*audit*) di: data di creazione, utente che ha creato (userid), ultima data di aggiornamento, ultimo utente che ha aggiornato (user-id).

Inoltre, sono possibili alcune varianti (per es. codice + descrizione breve/lunga). La presenza di tali attributi addizionali non influenza il processo di classificazione, piuttosto gli ulteriori attributi vengono considerati in ogni caso parte dei Dati Di Decodifica.

## Dati sostitutivi

**Codice + descrizione** Questo tipo di dati di decodifica contiene un codice e un nome o una descrizione esplicativa. Può servire quale ausilio rapido di *data entry* per utenti esperti, il nome esplicativo/descrizione può essere utile invece agli utenti meno esperti o per elenchi nella reportistica. Questo tipo di dati di decodifica può anche essere implementato per risparmiare spazio in memoria o essere il risultato di una normalizzazione. Se i dati sono dati sostitutivi, sono dati di decodifica e non debbono essere quindi contati.

**Esempi**

- Nazione: Codice Nazione, Nome Nazione
- Colore: Codice Colore, Descrizione Colore

**Variazioni**

- Codice, Linguaggio, Descrizione (per descrizioni in linguaggi multipli)
- Codice, Descrizione Breve, Descrizione Lunga, Abbreviazione.

## Dati statici o costanti

**Dati a una occorrenza** Questo tipo di dati di decodifica contiene una e una sola occorrenza, a prescindere dal numero di attributi. Possiede una sola riga di dati e i suoi attributi sono relativamente costanti; possono cambiare, ma molto raramente.

**Esempi**

- Un'entità con dati su una particolare organizzazione; per es. nome e indirizzo.
- Software COTS (*Commercial-Off-The-Shelf*) con un nome di linea aerea personalizzabile da parte dell'organizzazione dell'utente

**Dati statici** Questo tipo di dati di decodifica contiene dati essenzialmente statici. Il numero di istanze dei dati statici può cambiare, ma molto raramente, e il contenuto di una istanza cambia raramente.

**Esempi**

- Un'entità con elementi chimici: simbolo, numero atomico, descrizione
- Le tabelle dei function point per valutare il tipo di funzionalità e i livelli di complessità.

**Valori di default (modelli)** Questo tipo di dati di decodifica contiene valori predefiniti (*default*) per (alcuni attributi di) nuove istanze di un oggetto di business.

## Valori validi

**Valori validi** Questo tipo di dati di decodifica fornisce una lista di valori validi per un attributo di uno o più oggetti di business. È implementato per soddisfare requisiti come la riduzione di errori e l'incremento dell'usabilità. È tipicamente usato per elencare valori disponibili per la selezione degli utenti e/o per validare gli input forniti dagli utenti. Contiene dati che sono sostanzialmente statici; se non è così, potrebbe trattarsi di dati di riferimento o dati di business.

### Esempi

- Nome nazione: contiene tutti i valori validi per l'attributo nome nazione.
- Codice nazione: contiene tutti i valori validi per l'attributo codice nazione.
- Colore: contiene tutti i valori validi per l'attributo colore di un oggetto di business.

### Range di valori validi

Questo tipo di dato di decodifica contiene dati che sono di base statici; se non è un dato di decodifica, potrebbe essere un dato di riferimento.

### Esempi

- Range di numeri di telefono permessi: il numero di telefono più basso, il numero di telefono più alto.
- Range di temperature per misurare il calore.

## Quali dati non sono dati di decodifica

Questa sezione descrive i dati che non sono considerati dati di decodifica perché o sono dati di business o dati di riferimento. La Parte 1 e la Parte 2 – Capitolo 2 (File logici) contengono le regole per questi tipi di entità. Talvolta alcune tabelle vengono denominate ‘tabelle di decodifica’, ma sono in realtà dati di riferimento o anche dati di business.

### Esempi

Esempi di dati di business o di riferimento che non dovrebbero essere considerati dati di decodifica:

- Tipi entità con importi finanziari, tasso di cambio, e valori di imposta, se non sono costanti. Questi dati non restringono valori validi; piuttosto, aggiungono significato a un valore all'interno di range particolari.
- Dati di controllo. I dati mantenuti dall'utente che contengono regole di business indicando che cosa l'applicazione deve fare o quale comportamento deve avere.
- Tabelle di tassi di cambio. I tassi di cambio contengono il paese e il tasso di cambio in dollari US. Non è possibile sostituire il codice con il tasso di cambio del paese, i dati sono essenzialmente non statici, e i dati supportano attività di business; perciò, questo è un esempio di Dati di Riferimento.
- Range di valori di imposta in un sistema a tassazione progressiva. Differenti tassi di imposta sono applicabili a differenti range di reddito. Questo contiene valori minimi e massimi per ogni imposta. Comunque non si può sostituire il valore per l'entità e i dati che supportano le attività di business. I valori di imposta non restringono il range del reddito. Perciò, questo è un esempio di dati di riferimento.

## Esaminare i dati di decodifica e le relative transazioni

I dati di decodifica, come identificato nella sezione “Quali dati sono dati di decodifica”, rappresentano l’implementazione di requisiti utente non-funzionali piuttosto che l’implementazione di un requisito funzionale. Di conseguenza, i dati di decodifica e le transazioni che mantengono i dati di decodifica non si conteggiano nella dimensione funzionale di un’applicazione.

Tuttavia, dati che esistono come parte di un requisito utente non-funzionale potrebbero essere misurati usando una misura separata per misurare la dimensione non-funzionale.

Ad oggi, non esistono metodi distinti per contare i dati di decodifica e le loro funzioni di gestione e presentazione al fine di ottenere una misura della dimensione non-funzionale.

## Bibliografia

Le seguenti fonti sono state consultate o citate nel presente capitolo:

Definitions and Counting Guidelines for the Application of Function Point Analysis: A Practical Manual, Version 2.2 (NESMA, 2003),

ISBN: 90-76258-17-1.

Nota: Questo manuale è anche denominato *NESMA Counting Practices Manual*. Esso descrive la metodologia standard dell'analisi dei function point (FPA – Function Point Analysis) e molti aspetti relativi all'applicazione della FPA. Può essere usato insieme al manuale IFPUG. Per ulteriori informazioni consultare il sito web del NESMA: [www.nesma.org](http://www.nesma.org).

IFPUG “A Framework for Functional Sizing”, IFPUG, 2003.

ISO/IEC 14143-1:2007 Information technology – Software measurement Functional size measurement – Part 1: definition of concepts, ISO/IEC, 2007.

Questa pagina è lasciata intenzionalmente vuota.



---

### File logici

---

**Introduzione** Questo capitolo applica le definizioni e le regole che interessano i file logici, viene utilizzato un processo descrittivo per l'identificazione e la classificazione delle funzioni di tipo dati.

Queste regole illustrano come identificare le funzioni di tipo dati da un modello dei dati normalizzato. Una panoramica sulla normalizzazione dei dati verrà fornita a supporto del seguente approccio. Comunque, questo non preclude l'uso di queste regole in ambienti dove alternativamente si applicano tecniche di modellazione ad oggetti o di dati.

**Contenuti** Questo capitolo include le seguenti sezioni:

Argomento	Pagina
Descrizione alla metodologia	2-2
Concetti di Modellazione dei dati	2-3
Passo 1: Identificare i file logici	2-8
Passo 2: Classificare i file logici	2-21
Passo 3: Identificare i tipi di elementi dati (DET)	2-21
Passo 4: Identificare i tipi di elementi record (RET)	2-31
Bibliografia	2-42

## Descrizione della metodologia

### Introduzione

Per stabilire l'insieme dei file logici (ILF ed EIF) viene seguito un processo a passi, dove a ogni passo si analizzano i dati a un livello di dettaglio crescente. Segue la descrizione di ciascun passo della metodologia.

Passo	Azione
1	<b>Identificare i file logici</b>  Per ogni entità logica di dati, stabilire come si debbano raggruppare in file logici le entità ad essa legate, secondo il 'punto di vista dell'utente'. Per esempio, occorre determinare se un'entità di dati corrisponda a un File Logico indipendente, o se più entità debbano essere raggruppate in un unico File Logico. Dopo l'identificazione dei File Logici, questi vengono classificati come ILF o EIF. Questo passo viene illustrato nella sezione "Passo 1: Identificare e Classificare i File Logici".
2	<b>Classificare i file logici</b>  Ogni File Logico identificato viene classificato come ILF o EIF. Questo passo viene illustrato nella Sezione "Passo 2: Classificare I File Logici".
3	<b>Identificare i tipi di elementi dati (DET)</b>  Per ogni file logico, identificare gli elementi dati (DET) utilizzati dall'applicazione oggetto di conteggio. Questo passo viene illustrato nella sezione "Passo 3: Identificare i Tipi di Elementi Dati".
4	<b>Identificare i tipi di elementi record (RET)</b>  Per ogni file logico, stabilire come si debbano raggruppare in tipi di elementi record (RET) i dati ad esso legati che riflettano il 'punto di vista dell'utente'. Tale passo viene illustrato nella sezione "Passo 4: Identificare i Tipi di Elementi Record".

Il passo con il maggiore impatto sul risultato del conteggio è il "Passo 1: Identificare i file logici", poiché l'identificazione del corretto numero di file logici è decisiva per il raggiungimento della consistenza dell'intero conteggio. I rimanenti passi hanno un grado di impatto considerevolmente minore nel conteggio poiché non hanno effetto sul numero dei file logici, ma esclusivamente sulla loro tipologia e complessità.

## Concetti di modellazione dei dati

### Introduzione

Un riepilogo delle definizioni utilizzate nell'analisi dei dati (compresa la modellazione logica e fisica dei dati), può essere utile sia per illustrare i fondamenti della teoria, sia per chiarire i riferimenti teorici presenti nelle regole di conteggio per l'identificazione dei file logici, dei tipi di elemento record e dei tipi di elemento dati. una valida conoscenza dei concetti dell'analisi dei dati, rimane comunque un requisito preliminare implicito per effettuare l'analisi dei function point, in particolare per contare correttamente le funzioni di tipo dati. Nella sezione “Terminologia della Modellazione dei Dati” si riassumono i principali termini utilizzati nella modellazione dei dati.

### Concetti Principali della Modellazione dei Dati

Un riepilogo delle definizioni utilizzate nel dominio dell'analisi dei dati (compresa la modellazione dei dati e i DBMS), può essere utile sia per illustrare i fondamenti della teoria, sia per chiarire i riferimenti teorici presenti nelle regole di conteggio per l'identificazione dei file logici, dei tipi di elementi record (RET) e dei tipi di elementi dati (DET). Una valida conoscenza dei concetti relativi ai dati rimane comunque un requisito implicito per applicare le linee guida descritte nei prossimi capitoli per contare in modo appropriato e corretto le funzioni di tipo dati.

### Entità

#### Definizione di entità (o ‘tipo di entità’)

- ❑ Ogni persona, posto, cosa, evento o concetto, identificabile e le cui informazioni sono memorizzate (Thomas Bruce, 1992).
- ❑ Ogni oggetto che può essere chiaramente identificato (Peter Chen, 1976).
- ❑ Ogni oggetto identificabile da rappresentare nel database (C. J. Date, 1986).
- ❑ Una entità di dati rappresenta ‘qualcosa’ da memorizzare per futura referenza. Il termine entità si riferisce alla rappresentazione logica dei dati (Clive Finkelstein, 1989).
- ❑ Con la parola entità ci riferiamo a qualsiasi cosa della quale si memorizzano informazioni (per es. cliente, fornitore, macchinario, impiegato, palo della luce, del telefono, prenotazione posto aereo, ecc.). Per ogni entità sono memorizzati determinati attributi (James Martin, 1989).
- ❑ Un'entità potrebbe anche implementare la relazione tra due o più entità, viene allora chiamata entità associativa (Michael Reingruber, 1994).
- ❑ Un'entità potrebbe rappresentare un sottoinsieme di informazioni di una istanza di un'entità, viene allora chiamata entità sottotipo (conosciuta anche come entità secondaria o di categoria) (Michael Reingruber, 1994).

Riassumendo, un'entità:

- ❑ è un oggetto dati principale, sul quale si raccolgono informazioni,
- ❑ è una persona, un luogo, una cosa o un evento dotato di informazioni,
- ❑ può avere un'istanza (un'occorrenza),
- ❑ è un componente fondamentale, rilevante per l'utente, di cui viene mantenuta una raccolta ordinata di informazioni; una associazione tra entità, dotata di attributi, è essa stessa un'entità,
- ❑ richiede informazioni, una rappresentazione di dati simili che condividono proprietà o caratteristiche,
- ❑ nel modello dei dati è spesso raffigurata con un rettangolo, all'interno del quale è riportato il nome dell'entità.

### Elemento dati

Nell'ambito della modellazione dei dati, l'elemento di base è denominato come *elemento dati* o *oggetto dati*. Esso è

- ❑ il componente fondamentale,
- ❑ la fondamentale particella atomica nell'universo del sistema informativo (Gary Schutt),
- ❑ la più piccola unità di dati che ha significato nel mondo reale dell'utente (Graeme Simsion).

### Attività di modellazione dei dati

La modellazione dei dati tratta di elementi dati, record logici e file. Un **File System** è composto di record e dati. I **Dati** sono definiti come la più piccola unità di dati con significato nel mondo reale. Un gruppo di Dati collegati, trattato come una unità, è conosciuto come **record**. Un **file** è un raggruppamento ordinato di record dello stesso tipo.

Nell'implementazione fisica dei dati tramite database relazionali, vengono usati i seguenti termini: un elemento di dati è chiamato 'attributo' o 'colonna', un record è chiamato 'riga' o 'tupla', e un file è chiamato 'tabella'. Questi termini non modificano il significato di base dei concetti.

### Corrispondenza tra concetti relativi ai dati e alla terminologia dei function point

È possibile mettere in relazione i concetti di cui sopra con l'analisi dei function point (FPA, Function Point Analysis) come mostrato nella seguente tabella:

Concetto di Modellazione dei Dati	Termine Modell. Dati	Termine Database Relazionali	Termine FPA	Concetto FPA
La più piccola unità di informazione, dotata di un nome, che ha significato nel mondo reale.	Dato	Attributo o Colonna	Tipo di Elemento Dati (DET)	Un tipo di elemento dati (DET) è un campo unico, non ripetuto, riconoscibile dall'utente.
Gruppo di dati collegati che sono trattati come un'unità.	Record	Riga o Tupla	Tipo di Elemento Record (RET)	Un tipo di elemento record (RET) è un sottogruppo di dati, riconoscibile dall'utente, all'interno di un ILF o EIF.
Raccolta ordinata di record dello stesso tipo.	File	Tabella	File Logico (Interno, ILF, oppure d'Interfaccia Esterno, EIF)	Il termine 'file' si riferisce a un gruppo di dati collegati logicamente e non alla loro implementazione fisica.

Una volta identificati tutti i dati necessari, gli analisti applicano le varie regole di normalizzazione per poi disegnare i Diagrammi di Entità/Relazioni. Una sintesi delle regole di normalizzazione si può trovare nella sezione "Terminologia della Modellazione dei Dati".

### Terminologia della modellazione dei dati

#### Entità (o Tipo entità)

- ❑ Principale oggetto di tipo dati le cui informazioni sono memorizzate.
- ❑ Informazioni su persone, luoghi, cose o eventi.
- ❑ Istanza di un'entità (occorrenza).
- ❑ Raffigurata attraverso un rettangolo, con il nome dell'entità riportato all'interno del rettangolo
- ❑ Un componente fondamentale, rilevante per l'utente, di cui viene mantenuta una raccolta ordinata di informazioni. Una associazione tra entità, che contiene attributi, è essa stessa un'entità.

#### Entità associativa

Un'entità che contiene ulteriori attributi che descrivono una relazione 'molti-a-molti' tra altre due entità.

**Entità attributiva** Un'entità che descrive ulteriormente una o più caratteristiche di un'altra entità.

**Sottotipo di entità** Un'entità derivata per suddivisione da un tipo di entità. Un sottotipo di entità (talvolta 'entità figlio') eredita tutti gli attributi e le relazioni dell'entità padre e può avere propri attributi e relazioni specifiche supplementari

**Relazione** Rappresenta le associazioni esistenti tra una o più entità.

- ☐ Uno-a-Uno.
- ☐ Uno-a-Molti.
- ☐ Molti-a-Molti.
- ☐ Rappresentata da una linea che connette le entità.
- ☐ Il nome della relazione è riportato accanto alla linea.

Le relazioni sono definite dal modo in cui sono connesse le entità:

- ☐ opzionale, utilizza il testo tra parentesi 1:(N), (1):(N),
- ☐ obbligatoria, non utilizza le parentesi 1:1, 1:N.

**Attributi** ☐ Una caratteristica di una entità. Generalmente l'attributo è equivalente al tipo di elemento dati (DET).

**Normalizzazione** I dati sono normalizzati tramite 5 regole:

1. Eliminare i gruppi ripetuti (Prima Forma Normale – 1NF).
2. Eliminare i dati ridondanti (Seconda Forma Normale – 2NF).
3. Eliminare le colonne non dipendenti da chiave (Terza Forma Normale – 3NF).
4. Isolare le relazioni multiple indipendenti (nessuna tabella deve contenere due o più relazioni di tipo 1:N o N:M) (Quarta Forma Normale – 4NF).
5. Isolare le relazioni multiple semanticamente collegate (vincoli di natura pratica possono giustificare la separazione di relazioni di tipo multi-a multi logicamente collegate) (Quinta Forma Normale – 5NF).

Nello svolgere l'analisi dei function point, è preferibile analizzare il modello logico dei dati in terza forma normale (3NF).

Ignora le entità multiple inserite per motivi tecnici (frequentemente nella quinta forma normale – 5NF).

## Concetti entità-relazioni (E-R)

Una volta identificati tutti i dati necessari, gli analisti applicano le varie regole di normalizzazione per poi rappresentare i dati in vari diagrammi Entità-Relazioni. La seguente tabella inoltre può aiutare a comprendere il concetto di tipo di elemento record (RET).

Concetto E-R	Termine E-R	Termine FPA	Definizione IFPUG
Oggetto dati principale, le cui informazioni sono memorizzate (persone, luoghi, cose o eventi); componente fondamentale, rilevante per l'utente, di cui viene conservata una raccolta ordinata di informazioni.	Entità, o Tipo di Entità	File Logico Interno (ILF) o File d'Interfaccia Esterno (EIF)	Il termine 'file' si riferisce a un gruppo di dati logicamente collegati e non alla loro implementazione fisica.
Un'entità che contiene altri attributi significativi per l'utente, oltre quelli che descrivono la relazione con altre entità.	Tipo di Entità Associativa	Tipo di Elemento Record (RET)	Sottogruppo di dati, riconoscibile dall'utente, contenuto in un ILF o EIF.
Un'entità che descrive ulteriormente una o più caratteristiche di un'altra entità.	Tipo di Entità Attributiva	Tipo di Elemento Record (RET)	Sottogruppo di dati, riconoscibile dall'utente, contenuto in un ILF o EIF.
Una entità derivata da un'altra entità, che eredita tutti gli attributi e relazioni dall'entità padre; può avere propri attributi e relazioni specifiche aggiuntive.	Sottotipo di Entità	Tipo di Elemento Record (RET)	Sottogruppo di dati, riconoscibile dall'utente, contenuto in un ILF o EIF.

## Passo 1: Identificare i file logici

### Premessa

Nella FPA un File Logico (LF) è un gruppo di dati logico come visto dall'utente. Un file logico è composto da una o più entità di dati. In questo capitolo si forniscono le regole per la corrispondenza tra le entità candidate identificate e uno o più file logici.

Il processo consiste nei seguenti passi, spiegati in dettaglio nei seguenti paragrafi di questa sezione:

### Passo 1

---

#### Passi

---

1. Identificare tutte le relazioni logiche e i dati riconoscibili dall'utente o informazioni di controllo all'interno dello scopo del conteggio
  2. Escludere le entità che non sono mantenute da nessuna applicazione
  3. Raggruppare in un unico file logico quelle entità relazionate che sono entità dipendenti
  4. Escludere le entità che si riferiscono a dati di decodifica
  5. Escludere le entità che non contengono attributi richiesti dall'utente
  6. Rimuovere le entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiavi esterne con le entità primarie
- 

Il passo più difficile è il raggruppamento dei dati (voce 3). Il raggruppamento finale dei dati in file logici è il risultato dell'effetto combinato di metodi di raggruppamento:

- ☐ Il metodo a) è *process-driven*, basato sulle transazioni utente nell'applicazione.
- ☐ Il metodo b) è *data-driven*, basato sulle regole applicative (di business).

In ogni caso, poiché anche le transazioni utente sono (o dovrebbero essere) basate sulle regole applicative, entrambi i metodi si supportano l'uno con l'altro. Questo doppio approccio può rivelare eventuali lacune nelle specifiche funzionali e rende il processo di identificazione dei file logici affidabile e ripetibile.

#### Passo 1.1 **Identificare tutte le relazioni logiche e i dati riconoscibili dall'utente o informazioni di controllo all'interno dello scopo del conteggio**

Prima di decidere quali entità debbano essere raggruppate insieme per creare i file logici, si dovrebbero determinare le entità candidate da prendere in considerazione per il loro successivo raggruppamento logico (passo 1.3 3) e quali di esse dovrebbero essere escluse. Tale punto aiuta ad identificare queste entità in modo ripetibile.



Il principio guida generale della Parte 1 – Contare le Funzioni di Tipo Dati è chiaro: vanno considerate solo le entità significative e richieste espressamente dall'utente. Va prestata particolare attenzione durante l'identificazione dei file logici da un modello dati (normalizzato):

- ❑ Non assumere che tutte le entità siano file logici; es., file indici, entità trovate su un modello fisico dei dati
- ❑ I file logici possono esistere da una prospettiva utente, ma possono essere non identificati in un modello dei dati (normalizzato) in alcuni casi; es., file storici contenenti dati aggregati. Non dimenticare di includere tali file logici nel resto del processo.

### **Passo 1.2 Escludere le entità non mantenute da alcuna applicazione**

Determinare quali entità non siano mantenute da un processo elementare in questa o in un'altra applicazione. Escludere tali entità da ulteriori considerazioni poiché esse non vengono contate.

### **Passo 1.3 Raggruppare le entità con relazioni che dipendono da altre entità in file logici**

Per ogni restante entità di dati logica, identifica come le entità relazionate sono raggruppate in file logici, riflettendo il 'punto di vista utente'; es., determina se le entità di dati sono loro stessi un file logico indipendente o se le entità relazionate potrebbero essere raggruppate in un singolo file logico. Identificare il punto di vista dell'utente ("*user view*", equivalente al "*business view*") per il raggruppamento di dati, esaminando:

- a) Il modo in cui i processi elementari all'interno del confine dell'applicazione accedono ai dati visti come gruppo (passo 1.3a, pagina 2-11)
- b) Le relazioni tra le entità e le loro interdipendenze sulla base delle regole applicative (passo 1.3b, pagina 2-12 e successive).

### **Passo 1.4 Escludere le entità che si riferiscono a dati di decodifica**

Escludere i dati di decodifica. Il dato di decodifica è identificato come una risposta ad un requisito non funzionale (NFR) da parte dell'utente (requisiti di qualità, implementazioni fisiche e/o tecniche). Il dato di decodifica è trattato in dettaglio nella Parte 3 - Capitolo 1 "Dati di decodifica".

### **Passo 1.5 Escludere le entità che non contengono attributi richiesti dall'utente**

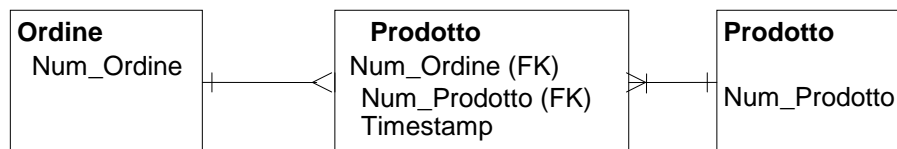
Determinare quali entità non contengano attributi riconoscibili e richiesti dall'utente, ma rappresentano attributi di tipo tecnico. Esempi di questi attributi tecnici sono tutti quegli attributi che esistono in quanto risultato di considerazioni di tipo implementativo o tecnico; per es. indici creati prestazioni morivi prestazionali, così come gli indici secondari (vedi la Parte 1 – Contare le Funzioni di Tipo Dati). Escludi queste entità da ulteriori considerazioni; esse non sono da contare come file logici né come RET.

## Passo 1.6 Rimuovere le entità associative che contengono attributi addizionali non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con entità primarie

**1.6.1** Determinare quali entità siano di tipo associativo. Un'entità associativa contiene le chiavi esterne delle entità collegate, in aggiunta ad altri attributi. Si noti che si possono presentare due casi:

- Gli attributi 'non-chiave' addizionali rappresentano il risultato di considerazioni implementative o esistono comunque per soddisfare esigenze tecniche (non richieste espressamente dall'utente; per es. un timbro relativo alla data/ora – timestamp – atto a consentire il recupero dei dati). Tali attributi tecnici non sono contati come tipi di elementi dati. Tratta queste entità come di tipo “chiave-a-chiave” (vedi nel seguito).
- Gli attributi 'non-chiave' addizionali sono necessari per soddisfare dei requisiti funzionali (FUR) ed espressamente richiesti dall'utente. Tali entità saranno valutate nelle prossime sezioni: Identificare i File Logici (passi 1.3a/1.3b).

**Esempio:**



*Il 'timestamp' è solitamente un attributo tecnico non riconosciuto dall'utente. In questo caso, per l'entità Ordine-Prodotto, si applica la linea guida 1.2.a. L'entità 'chiave-a-chiave' si risolve aggiungendo l'attributo Num\_Prodotto come chiave esterna all'entità Ordine e l'attributo Num\_Ordine all'entità Prodotto (passo 1.6).*

**1.6.2** Determinare quali entità siano una entità di tipo 'chiave-a-chiave' (intersezione); cioè, quelle che abbiano solo chiavi tra i loro tipi di elementi dati e non abbiano altri attributi non-chiave. Tali entità tipicamente rappresentano l'implementazione di relazioni di tipo molti-a-molti (N:M) in un modello dei dati normalizzato. Tali entità esistono solo per scopi di modellazione dei dati e di progettazione del database, non come risultato di un requisito utente.

Escludere tali entità da ulteriori considerazioni; esse non vanno contate come file logici né come RET. In accordo alle regole (Parte 1), gli attributi che risolvono i legami delle entità (chiavi-esterne) sono contati come un DET per entrambe le entità connesse dall'entità 'chiave-a-chiave'. Vedi anche le regole nella sezione “Passo 3: Identificare i Tipi di Elementi Dati (DET)”.

## Controllo finale

Controllare se tutte le rimanenti entità siano un risultato dei requisiti utente funzionali (FUR). Tali entità, e le relazioni e interdipendenze tra queste entità, saranno valutate nella prossima sezione: Identificare i file logici (sotto-passi 1.3a/1.3b).

## Identificare i file logici tramite il metodo dei processi elementari (passo 1.3a)

Il punto di vista di business dell'utente sui dati si riflette nel modo in cui le transazioni utente accedono ai dati.

Si esamini in che modo i processi elementari nel confine dell'applicazione, mantengono le entità. Se varie entità sono sempre create insieme e cancellate insieme, allora questa è un'indicazione sufficientemente robusta del fatto che tali entità dovrebbero essere raggruppate in un unico file logico. Si esaminino anche i processi elementari usati per estrarre i dati, verificando se il processo di estrazione acceda allo stesso gruppo di entità.

Nota: le transazioni che modificano dati, spesso insistono solo su un'entità del gruppo, così le transazioni di modifica non sono efficaci nell'aiutare nel raggruppamento dei dati, come le transazioni di creazione e cancellazione.

### Esempio

Un ordine d'acquisto di un cliente rappresenta un singolo gruppo di dati da una prospettiva applicativa dell'utente ed è composto dall'Intestazione dell'Ordine (cliente, indirizzo, data, ecc.) e dalle righe di dettaglio di ogni ordine. Da un punto di vista di business, un ordine non può essere creato se non ha almeno una riga ordine e, nella cancellazione dell'ordine, devono essere cancellati l'intestazione e tutte le righe. Comunque l'intestazione e le righe potrebbero anche essere gestite da transazioni indipendenti. Ad esempio, cambiare lo stato dell'ordine rappresenta una funzione distinta da quella che si occupa di modificare le righe ordine. Le funzioni di *creazione* e *cancellazione* indicano, da una prospettiva utente, che l'ordine è un unico file logico che raggruppa l'intestazione dell'ordine e le righe ordine.

Seguire quanto indicato nel passo 1.3a per validare i gruppi di dati logici candidati identificati.

## Identificare i file logici tramite il metodo della (in)dipendenza (passo 1.3b)

### Introduzione

Il Metodo della (In)Dipendenza dell'Entità, così come definito e trattato in questa sezione, supporta l'identificazione dei file logici da un modello dei dati, in modo corretto e ripetibile. Il termine 'entità' in questa sezione si riferisce ad un'entità in un modello dei dati normalizzato (usualmente nella terza forma normale).

La sezione "Tipi di Relazioni" spiega i vari tipi di relazione e la differenza tra i concetti 'relazione obbligatoria/opzionale' e 'entità dipendenti/indipendenti'.

La sezione "(In)Dipendenza dell'Entità illustrata per tutti i Tipi di Relazione" spiega il metodo in maggiore dettaglio per ogni tipo di relazione.

La sezione "Riepilogo: da Entità a File Logici tramite l'Analisi della (In)Dipendenza dell'Entità" riassume i diversi tipi di relazione e le condizioni per contare un File Logico.

Il Metodo delle Entità (In)Dipendenti raggruppa le entità valutandone le relazioni e le interdipendenze congiuntamente alle esigenze applicative. I principi guida sono l'indipendenza delle entità e la dipendenza delle entità.

### Indipendenza dell'entità

*Indipendenza dell'entità* significa che un'entità è significativa o importante per il business in sé e per sé senza la presenza di altre entità.

### Dipendenza dell'entità

*Entità dipendente* significa che un'entità *non* è significativa o importante per il business in sé e per sé senza la presenza di altre entità, tale che

- un'occorrenza dell'entità X deve essere collegata ad un'occorrenza dell'entità Y,
- dall'eliminazione di un'occorrenza nell'entità Y consegue l'eliminazione di tutte le occorrenze correlate nell'entità X.

### Nota

Non si confonda il concetto di entità indipendente/dipendente con il concetto di relazione opzionale / obbligatoria. L'esempio nella sezione "(In)Dipendenza dell'Entità illustrata per tutti i Tipi di Relazione" mostra chiaramente che questi sono concetti differenti.

**Determinare la dipendenza**

Per determinare se l'entità B è dipendente o indipendente da A, bisogna determinare:

*“B è significativa per il business anche senza l'occorrenza di A ad essa collegata?”*

Un semplice test per determinare la situazione (dipendenza o indipendenza dell'entità) è il seguente. Anche qualora non vi siano particolari requisiti utente sulla cancellazione, ci si ponga, comunque, la seguente domanda:

*“Si supponga di voler cancellare un'occorrenza ‘a’ dell'entità A, cosa succederebbe ad un'occorrenza ‘b’ dell'entità B che era collegata a ‘a’?”*

In funzione delle regole applicative, vanno distinti essenzialmente due casi:

**Caso 1**

Se l'occorrenza di B, in accordo con le regole applicative, non presenta una significatività/importanza indipendente per l'utente e quindi tanto vale cancellarla, allora l'occorrenza di B non ha evidentemente significato per l'utente senza essere collegata all'occorrenza di A. L'entità B va considerata dipendente da A. Le entità A e B dovrebbero essere raggruppate insieme in un unico file logico.

**Caso 2**

Se l'occorrenza di B conserva una sua significatività, anche senza essere più collegata all'occorrenza di A, le regole applicative non permetteranno di cancellare anche l'occorrenza di B. Le entità A e B vanno considerate come file logici separati.

La valutazione del modello dei dati di un sistema informativo, tramite la valutazione di tutte le coppie di entità collegate, dà come conseguenza l'identificazione dei file logici.

Nella sezione “(In)Dipendenza dell'Entità illustrata per tutti i Tipi di Relazione” questo metodo è esposto in maggiori dettagli per i differenti tipi di relazioni.

La sezione “Riepilogo: da Entità a File Logici tramite l'Analisi della (In)Dipendenza dell'Entità” riassume come contare nella FPA i diversi tipi di relazione.

## Tipi di relazione

Prima di riepilogare i principi sulla (In-)Dipendenza delle Entità per tutti i tipi di relazioni, si dovrebbero chiaramente comprendere i differenti tipi/generi di relazione. Questa sezione spiega tali differenze, così come i concetti di ‘obbligatorietà’ e ‘opzionalità’.

### Esempio

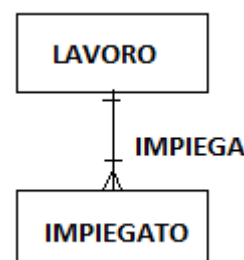
Due entità, per esempio Lavoro e Impiegato, possono essere connesse tra loro mediante una relazione; per es. ‘impiega’.

### Natura della relazione

La natura della relazione determina il modo in cui più impiegati possono svolgere uno stesso lavoro in base al modello dei dati (0, 1 o più) e quanti lavori possono essere svolti da un singolo impiegato (0, 1 o più).

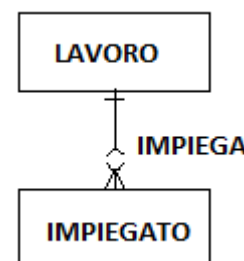
#### 1 : N

Si assuma che una regola applicativa stabilisca che per un lavoro possano essere utilizzati più impiegati (ma almeno uno), e che un singolo impiegato debba svolgere un lavoro (e uno solo). In tal caso si dice che la relazione tra Lavoro e Impiegato è del tipo 1:N. Nella figura al lato è mostrato questo tipo di relazione.



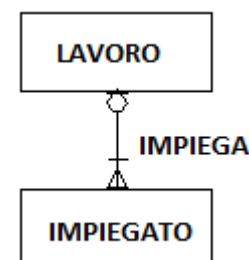
#### 1 : (N)

Più probabilmente le regole applicative stabiliscono che un lavoro può non essere assegnato né a uno né a più impiegati. In tal caso la relazione tra Lavoro e Impiegato è opzionale e viene indicata come 1:(N). Nella figura al lato è mostrato questo tipo di relazione.



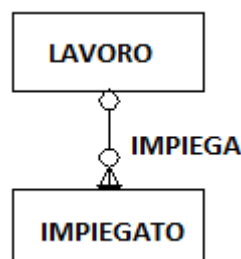
#### (1) : N

Se le regole applicative stabiliscono che può esistere un impiegato senza un lavoro, ma che un lavoro deve sempre essere assegnato ad un impiegato, si indica la relazione tra Lavoro e Impiegato come (1):N. Nella figura al lato è mostrato questo tipo di relazione.



#### (1) : (N)

Nel caso in cui un lavoro può non essere assegnato e in cui un impiegato può esistere senza un lavoro, entrambi i lati della relazione sono opzionali. La relazione tra Lavoro e Impiegato è indicata come (1):(N). Nella figura al lato è mostrato questo tipo di relazione.



**Confronto dei concetti di ‘obbligatorio/opzionale’ e ‘(in)dipendenza dell’entità’**

Per chiarire la differenza tra i concetti ‘relazione obbligatoria/opzionale’ e ‘entità dipendente/indipendente’ si prendano come esempio le relazioni derivate dalla regola di business “*non è possibile avere uno o più impiegati senza almeno un lavoro assegnato*” (relazioni di tipo 1:N e 1:(N)).

Se un lavoro diventa obsoleto, questo non implica che anche gli impiegati che lo svolgevano non siano più significativi. Un impiegato ha una sua importanza applicativa indipendentemente dal lavoro assegnato. Impiegato è un’entità indipendente dall’assegnazione del lavoro. A causa dell’obbligatorietà della relazione con il lavoro, però tutti gli impiegati che svolgevano il lavoro obsoleto devono essere riassegnati ad un nuovo lavoro, prima che il lavoro obsoleto possa essere cancellato.

Così può accadere che una occorrenza dell’entità B (per es. Impiegato) possa avere un legame obbligatorio a una occorrenza dell’entità A (per es. Lavoro) nella relazione A:B tra le entità A e B, ma nondimeno l’entità B gode di una propria indipendenza e significatività applicativa. In casi come questo, se si volesse cancellare un’occorrenza dell’entità A, si dovrebbe prima provvedere a correlare la relativa occorrenza di B a un’altra occorrenza di A.

**(In)Dipendenza dell’entità illustrata per tutti i tipi di relazione****(In)Dipendenza dell’entità in una relazione (1):(N)****(1) : (N)**

Se una relazione tra due entità è opzionale nei due lati, le entità possono esistere indipendentemente e (tutte le occorrenze di) A e B sono significative anche senza la correlazione con una o più occorrenze dell’altra entità.

Perciò A e B sono considerate entità indipendenti l’una dall’altra. La FPA conteggia le entità A e B come due file logici separati, così come indicato nella sezione “Riepilogo: da Entità a File Logici tramite l’Analisi della (In)Dipendenza dell’Entità”.

**(In)Dipendenza dell’entità in una relazione 1:(N)****1 : (N)**

In una relazione di tipo 1:(N) tra due entità A e B (vedere la figura 1), una occorrenza dell’entità A potrebbe esistere con nessuna o più occorrenze dell’entità B correlate. D’altra parte, ogni occorrenza di B deve essere correlata con un’occorrenza di A.

**Esempio**

Nella relazione 1:(N) tra Impiegato e Figlio (o Dipendente), nell’applicazione Risorse Umane (RU), un Impiegato potrebbe avere 0, 1 o più figli correlati, ma un figlio deve sempre essere ad un (e uno solo) impiegato (vedere la figura 2).

Poiché B deve essere correlato ad A, si pone la questione se B sia da considerarsi dipendente o indipendente da A

Per determinare se l'entità B sia dipendente o indipendente da A, occorre chiedersi:

*“B è significativa per il business a parte dalla A collegata ad essa?”.*

Un facile test per distinguere tra entità dipendenti e indipendenti è il seguente.

Anche qualora non vi siano particolari requisiti utente sulla cancellazione, ci si ponga la seguente domanda:

*“Supponiamo si voglia cancellare un'occorrenza dell'entità A, cosa succederebbe alle occorrenze dell'entità B che ha un collegamento obbligatorio con un'occorrenza di A?”.*

Le regole di business possono dar luogo a due possibilità:

### **Caso 1**

Se la cancellazione di A è permessa, allora devono essere cancellate anche tutte le occorrenze di B collegate, poiché il business non è più interessato alle occorrenze di B. Per esempio (vedi Fig. 2): l'applicazione RU mantiene informazioni sugli impiegati e loro figli. Si assuma che la regola applicativa dica che quando un impiegato (A) lascia l'azienda, non c'è più motivo di mantenere le informazioni sul figlio (B).

### **Caso 2**

La cancellazione di A non è permessa fintanto che sussistono dei collegamenti con B, in quanto B è ancora importante per le regole applicative, anche al di là del contesto in cui era collegato ad A. Per esempio (vedere la figura 3): Una organizzazione adotta dei bambini e assegna ogni bambino a un impiegato. Gli impiegati fanno da tramite tra l'azienda e i bambini. Nel caso in cui un impiegato lasci l'azienda, le informazioni del relativo bambino che gli era stato assegnato, sono ancora importanti per l'organizzazione. Così, prima di cancellare l'impiegato (A), bisogna prima associare il relativo bambino (B), ad un altro impiegato (A), in quanto il tipo di relazione non consente di avere bambini senza un impiegato associato.

Nel caso (1) si dice che B è un'entità *dipendente* da A, e nel caso (2) che B è un'entità *indipendente* da A.

La FPA conteggia le entità A e B come un unico file logico nel caso (1) (*dipendenza*), mentre nel caso(2), A e B sono ognuno un file logico separato (*indipendenza*), come indicato nella sezione “Riepilogo: da Entità a File Logici tramite l'Analisi della (In)Dipendenza dell'Entità”.



Ecco l'illustrazione delle relazioni di tipo 1:(N):

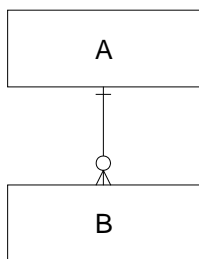


Fig. 1:  
Ogni occorrenza dell'entità A può essere collegata a 0, 1 o più occorrenze dell'entità B. Ogni occorrenza dell'entità B deve essere collegata esattamente a una occorrenza dell'entità A.

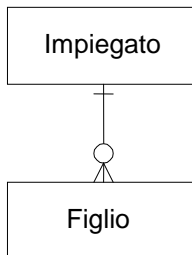


Fig. 2:  
L'applicazione RU mantiene informazioni sugli Impiegati e loro Figli.

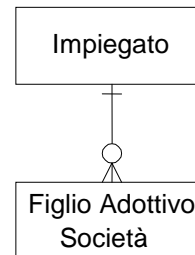


Fig. 3:  
L'applicazione RU mantiene informazioni sugli Impiegati e sui bambini adottati dall'organizzazione e assegnati a un impiegato.

Le figure 2 e 3 presentano modelli di dati simili, ma regole di business differenti danno luogo a file logici identificati differenti.

### (In)Dipendenza dell'entità in una relazione (1):N

#### (1) : N

Una relazione (1):N tra due entità A e B (vedere figura 4) può essere trattata in modo analogo. Questo tipo di relazione comunque si presenta raramente nella pratica.

In una relazione (1):N tra due entità A e B, ogni A deve essere assegnato a uno o più B. D'altra parte un B può, ma non necessariamente, essere assegnato ad un'occorrenza di A.

#### Esempio

Nella relazione (1):N tra Comitato e Membri dell'Organizzazione, un Comitato deve avere dei membri (almeno uno). Un membro dell'organizzazione potrebbe, ma non necessariamente, partecipare ad un Comitato (vedi figure 5 e 6).

Poiché un'occorrenza di A deve essere collegata una B, sorge la questione se A sia dipendente o indipendente da B.

Per determinare se l'entità A sia dipendente o indipendente da B, occorre chiedersi:

*“A è significativa per il business a parte dalla B collegata ad essa?”.*

Un facile test per distinguere tra entità dipendenti e indipendenti è il seguente.

Anche qualora non vi siano particolari requisiti utente sulla cancellazione, ci si ponga la seguente domanda:

*“Si assuma di avere un’occorrenza dell’entità A alla quale sono collegate una o più occorrenze dell’entità B. Si supponga di voler cancellare l’ultima occorrenza rimasta collegata dell’entità B, cosa succederebbe all’occorrenza nell’entità A che possiede un collegamento obbligatorio con almeno un’occorrenza di B?”.*

Le regole applicative possono dar luogo a due possibilità:

### Caso 1

Quando l’ultima B è cancellata, viene cancellata anche la A collegata poiché l’utente non è più interessato a quella A. Per esempio (vedi Figura 5): un’organizzazione ha dei comitati ai quali sono assegnati dei membri.

La regola applicativa è che un comitato *deve* avere dei membri, ma non è necessario che tutti debbano partecipare ad un comitato. Un’altra regola è che l’organizzazione cancella un comitato non appena non ha più partecipanti; si potrebbe dire che il comitato è visto come uno specifico gruppo di lavoro

In questo caso, quando l’ultimo membro lascia lo specifico gruppo di lavoro, *non* ha più senso mantenere le informazioni sul comitato. I dati sul comitato sono cancellati non appena l’ultimo membro lascia il comitato.

### Caso 2

La cancellazione dell’ultimo B *non* è permessa almeno fintanto che A è ancora collegato, perché l’utente è sempre interessato a quello specifico A, anche al di là del contesto che lo vede legato ai B. Per esempio (vedere figura 6), una organizzazione ha dei comitati ai quali sono assegnati dei membri.

La regola è che un comitato *deve* avere membri, ma non tutti i membri devono necessariamente partecipare ad un comitato. I comitati sono visti come parte della struttura organizzativa. Essi hanno significato per l’utente al di là dei membri che vi partecipano.

Prima che l’ultimo membro di un certo comitato lasci, deve essere assegnato un nuovo membro a quel comitato, in quanto il tipo di relazione non consente l’esistenza di un comitato senza membri.

Nel caso (1), A è evidentemente *non* significativo per il business, a meno che non sia collegato a uno o più B, mentre nel caso (2) è comunque significativo.

Nel caso (1), si dice che A è un’*entità dipendente* da B, e nel caso (2) che A è un’*entità indipendente* da B.

La FPA conteggia le entità A e B come un unico file logico nel caso (1) (*dipendenza*), mentre nel caso (2), A e B sono ognuno un file logico separato (*indipendenza*), come indicato nella sezione “Riepilogo: da Entità a File Logici tramite l’Analisi della (In)Dipendenza dell’Entità”.

Ecco l'illustrazione delle relazioni di (1):N:

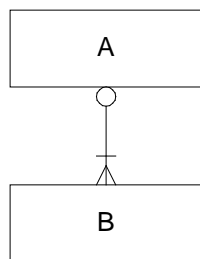


Figura 4:  
Ogni occorrenza dell'entità A deve essere collegata a 1 o più occorrenze dell'entità B. Ogni occorrenza dell'entità B può essere collegata, ma non necessariamente, a una occorrenza dell'entità A.

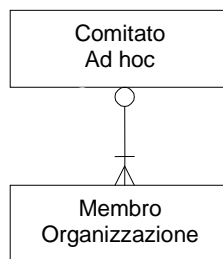


Figura 5:  
I membri di un'organizzazione possono (ma non necessariamente) partecipare a un certo comitato. Un comitato deve avere (uno o più) membri partecipanti.

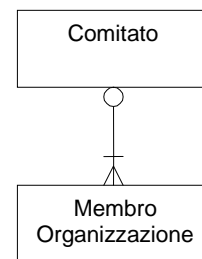


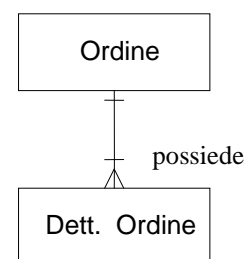
Figura 6:  
I membri di un'organizzazione possono (ma non necessariamente) partecipare a un certo comitato. Un comitato deve avere (uno o più) membri partecipanti.

Le figure 5 e 6 presentano modelli di dati simili, ma regole di business differenti danno luogo a file logici identificati differenti.

### (In)Dipendenza dell'entità in una relazione 1:N

#### 1 : N

In una relazione 1:N tra due entità A e B, ogni entità B deve essere assegnata ad una ed una sola A, e ogni A deve essere assegnata almeno ad una B. Si applicano le stesse regole viste per la valutazione della dipendenza e indipendenza.



#### Caso 1

Se B *non* è significativa per l'utente, senza il collegamento con A, allora B si considera come *entità dipendente* da A.

#### Caso 2

Se B è significativa per l'utente, a prescindere dal suo collegamento con A, allora B si considera come entità indipendente da A.

La FPA conteggia le entità A e B come un unico file logico nel caso (1) (*dipendenza*), mentre nel caso (2), A e B sono ognuno un file logico separato (*indipendenza*), come indicato nella sezione “Riepilogo: da Entità a File Logici tramite l'Analisi della (In)Dipendenza dell'Entità”.

**Riepilogo: da entità a file logici tramite l'analisi della (in)dipendenza dell'entità**

Nella tabella seguente, A e B sono due entità di un modello dei dati (normalizzato) che sono state individuate in accordo con quanto riportato nella sezione “Identificare le Entità da Considerare nel Conteggio” e che sono interconnesse tramite una relazione. La tabella riassume come le varie situazioni sono contate.

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

**Legend**

LF = File Logico (Logical File, ILF o EIF).  
 (..) = Lato opzionale della relazione.

**Note**

1. In caso di dubbio, scegli entità indipendenti.
2. In alcuni casi, anche più di due entità possono costituire un file logico.

## Passo 2: Classificare i file logici

I file logici identificati devono essere controllati in base alle regole di conteggio sugli ILF/EIF della Parte 1.

Si classifica un file logico come un File Logico Interno (ILF) se sono stati contati dei processi elementari, all'interno del confine dell'applicazione, che mantengono (tramite operazioni di inserimento, modifica o cancellazione) i dati nel file.

Si classifica un file logico come un File Logico Esterno (EIF) se sono stati contati dei processi elementari, all'interno del confine dell'applicazione, che referenziano solamente i dati nel file, e il file logico è mantenuto da un processo elementare di un'altra applicazione.

Se un file logico identificato *non* è mantenuto da un processo elementare (o all'interno dell'applicazione o in altre), allora il file logico non viene conteggiato affatto.

## Passo 3: Identificare i tipi di elementi dati (DET)

L'elemento dati è il più piccolo elemento significativo per l'utente e rappresenta uno specifico fatto riguardante il business, per esempio:

Nome DET	Valore DET
Tariffa	\$900
Data di Nascita	15 Gennaio 1965
Nome	InfoMerge

## Elementi dati: termini e definizioni

Quando si esamina un modello logico dei dati, si comincia con il considerare tale elementi dati come attributi. Un attributo rappresenta una specifica caratteristica di una entità o relazione. Nella seguente tabella le entità sono mostrate in CARATTERI MAIUSCOLI, gli attributi in caratteri minuscoli:

Rappresentazione	Esempio
ENTITÀ_attributo	CORSO_tariffa
ENTITÀ.attributo	AZIENDACLIENTE.nome

Gli elementi dati/attributi si possono riscontrare:

- ❑ nelle visualizzazioni per l'utente (prospetti, maschere),
- ❑ nei dizionari dei dati (modelli di business, modelli dei dati),
- ❑ nei file effettivi (strutture dei record nei programmi, impaginazione dei file).

Nell'esaminare i dati, l'analista dei dati segue un'assunzione di base: ogni tipo di elemento dati, conosciuto dall'utente, deve essere trattato come un attributo, quindi deve essere visibile in qualche specifica entità.

Gli attributi potrebbero avere alcune proprietà come: nome dell'attributo (alias), scopo (utilizzo), valore iniziale, dominio di valori ammessi, valore (struttura), unità di misura e dipendenze. Andremo ad approfondire tali proprietà prima di esaminare la corrispondenza tra elementi dati/attributi e DET nell'analisi dei function point.

### Nome dell'attributo

Un nome univoco che sintetizza le caratteristiche da rappresentare. Possiede i seguenti elementi:

**posizione** (entità/relazione) *seguita da un punto*

**descrizione** (qualificativo dell'attributo) *seguita da un trattino*

**classe** (attributo base)

Molte tecnologie non accettano gli spazi, così i nomi multipli sono concatenati da trattini.

Esempio:

AZIENDA\_CLIENTE.spedizione-indirizzo

SEMINARIO\_ISCRIZIONE.efficacia-valutazione

### Caratteristica

La proprietà del contesto oggetto di misurazione o rappresentazione. Il nome dell'entità che contiene la caratteristica viene sempre dichiarato; per esempio, spedizione-indirizzo: l'indirizzo dove spedire il materiale per l'AZIENDA-CLIENTE

### Scopo

L'attributo è giustificato dalla risposta a questa domanda: “*a cosa serve l'attributo nella mia applicazione?*”.

Esempio: CORSO.Data-qualificazione

Scopo: Usato nella schedulazione delle sessioni di verifica

**Dipendenze** I casi in cui i valori di altri attributi nel modello influenzano o vincolano il valore dell'attributo in esame. Per es. il CORSO.voto-finale non può esistere prima dell'attributo fine-semester, ma deve necessariamente esistere al termine del corso.

**Attributi chiave** Forniscono il legame tra un'entità e l'altra. Esistono diversi tipi di chiave in un modello dei dati, per es. chiavi primarie, chiavi secondarie e chiavi esterne.

Una **Chiave Primaria (PK – Primary Key)** è l'identificativo univoco di un'entità.

Le **Chiavi Secondarie (SK – Secondary Keys)** rappresentano attributi implementati per consentire un veloce accesso alle informazioni, come per esempio:

LIBRO\_DI\_TESTO.costo (SK)

LIBRO\_DI\_TESTO.nome-editore (SK)

Le Chiavi Secondarie non sono rappresentate nel modello dei dati delle informazioni (modello logico dei dati), ma sono usate principalmente come un aiuto per l'accesso ai dati (implementazione fisica).

Le **Chiavi Esterne (FK – Foreign Keys)** sono attributi usati per risolvere le relazioni tra un'entità e l'altra.

**Attribuzione** L'ultimo concetto di modellazione dei dati da considerare prima di passare all'analisi dei DET è quello dell'Attribuzione, che prescrive/describe dove risiede l'attributo, o all'interno di un'entità, o all'interno di una relazione. Esistono alcune regole generali sull'attribuzione seguite nella modellazione dei dati:

1. Un attributo è assegnato al suo specifico e miglior 'posto', indicato nell'apposita proprietà nella MASCHERA DI DEFINIZIONE ATTRIBUTI.
2. L'identificativo univoco (chiave primaria) di un'entità sarà attribuito anche a ogni relazione dell'entità stessa.

Altre regole per l'attribuzione dell'attributo nella giusta entità:

1. Se la definizione dell'attributo si riferisce a un'entità, si assegni l'attributo a quella entità.
2. Se la definizione di un attributo si riferisce a più entità, allora si crei una relazione e si assegni l'attributo alla relazione o all'entità relativa.

## Corrispondenza tra elementi dati e tipi di elementi dati (DET) dell'Analisi dei Function Point

Ora che sono state esaminate le basi degli elementi dati e degli attributi dal punto di vista della modellazione dei dati, si possono mettere in relazione tali concetti con le definizioni e le regole dei function point IFPUG:

Concetti di Modellazione dei Dati	Termini Modell. Dati	Termini Database Relazionali	Termini FPA	Concetti FPA
La più piccola unità di informazione che ha significato nel mondo reale.	Elemento Dati	Attributo o Colonna	Tipo di Elemento Dati (DET)	Il Tipo di Elemento Dati (DET) è un campo unico, non ripetuto, riconoscibile dall'utente.
Gruppo di dati collegati che sono trattati come un'unità.	Record	Riga o Tupla	Tipo di Elemento Record (RET)	Il Tipo di Elemento Record (RET) è un sottogruppo di dati, riconoscibile dall'utente, all'interno di un ILF o EIF.
Raccolta ordinata di record dello stesso tipo.	File	Tabella	File Logico (File Logico Interno – ILF, o File d'Interfaccia Esterno – EIF)	Il termine 'file' si riferisce a un gruppo di dati collegati logicamente e non alla loro implementazione fisica.

I **Tipi di Elemento Dati (DET)** sono campi o attributi unici, non ripetuti, riconoscibili dall'utente.

Le seguenti regole vanno applicate nel conteggio dei DET in File Logici:

- Contare un DET per ogni campo unico, riconoscibile dall'utente, non ripetuto, mantenuto o reperito da un file logico, attraverso l'esecuzione di tutti i processi dentro lo scopo del conteggio
- Se due o più applicazioni mantengono e/o referenziano lo stesso file logico, si contano solo i rispettivi DET utilizzati per ogni applicazione.
- Contare un DET per ogni porzione di dati richiesta dall'utente allo scopo di stabilire una relazione con un altro file logico.
- Analizzare gli attributi correlati per determinare se loro sono raggruppati e contati come singoli DET o se sono contati come multipli DET; i gruppi dipenderanno da come il processo elementare usa gli attributi nell'applicazione



Non contare gli attributi che esistono unicamente per soddisfare un requisito tecnico e non siano stati specificati espressamente dall'utente. Esempi di tali attributi tecnici sono quelli risultanti da considerazioni di progettazione o implementative.

Esempio: Data\_ORDINE è contato come un DET su Ordine, poiché deve essere mantenuto per rispondere ai requisiti utente. Mentre l'ora e il marchio della data su ogni record servono per soddisfare l'integrità e l'affidabilità dei dati. La soluzione tecnica a questi requisiti di qualità è stata infatti quella di salvare il database con la possibilità di recuperare i dati sulla base delle informazioni sulla data e ora. Di conseguenza l'ora e il 'timestam' non potranno essere contate come DET.

## Casi vari

Seguono degli esempi di conteggio dell'elemento di base qui considerato, il tipo di elemento dati (DET).

### Attributi

Attributi che sono composti di diversi dati correlati e memorizzati separatamente.

Si dovrebbero contare più DET o uno solo? Le regole sui DET nella Parte 1 recitano: "Si conta ogni campo riconoscibile dall'utente".

Ma come determinare se sia riconoscibile come un unico oggetto o come unione di più oggetti? è necessario riesaminare le transazioni dell'applicazione per determinare se l'attributo è trattato come uno o più campi.

Si considerino i seguenti punti nel prendere una decisione:

- a) Se l'attributo è sempre usato per intero, allora si conta come un singolo elemento dati (DET). Non dovrebbero esserci dei casi in cui un componente individuale dell'attributo è usato senza gli altri. Sulla base di questa modalità di utilizzo, l'attributo è contato come un singolo DET.
- b) Se in alcuni casi, viene utilizzata solo una parte dell'attributo (per es. il cognome), allora si dovrebbe contare più di un DET. Si osservi l'utilizzo dei componenti nell'ambito dell'applicazione per determinare le parti riconoscibili. Non necessariamente devono essere usati uno per volta o tutti insieme. Per esempio, in base all'osservazione, potrebbe essere appropriato contare solo due DET, anche se sono presenti cinque pezzi fisici che compongono l'attributo.

- c) Si verifichi l'esistenza di ordinamenti o di requisiti per stabilire i criteri di ordinamento o di ricerca. Se delle liste o stampe sono ordinate o filtrate su un singolo componente dell'attributo, questo suggerisce l'indipendenza del componente secondo il punto di vista utente.

## Conteggio dei nomi

Nome (nome, iniziale secondo nome e cognome)

In molte applicazioni si usano e tracciano i nomi di persone. I nomi composti da più campi si dovrebbero contare come DET multipli o come un singolo DET?

È necessario riesaminare quelle transazioni dell'applicazione nelle quali occorre determinare come considerare il nome, se come unico elemento, o se costituito da più elementi distinti. Per esempio si osservi in che modo viene trattato il Nome Impiegato nelle varie transazioni nei Casi di Studio 1, 2, e 3. Viene sempre utilizzato il nome per intero o talvolta ne viene utilizzata solo una parte?

Nei Casi di Studio il Nome Impiegato viene sempre utilizzato nella sua interezza. Non ci sono schermate o stampe dove è utilizzata una singola parte del nome a meno delle altre. Non ci sono neanche casi come ordinamenti, modifiche o criteri di filtro dove è utilizzata una singola parte del nome. Quindi il modo in cui è utilizzato nell'ambito dell'applicazione, suggerisce che Nome Impiegato sia considerato come un singolo tipo di elemento dati (DET).

## Conteggio degli indirizzi

Indirizzo (via, città, stato e codice di avviamento postale)

In molte applicazioni si usano e tracciano gli indirizzi. Gli indirizzi composti da più campi si dovrebbero contare come DET multipli o come un singolo DET?

Si rivedano quelle transazioni dell'applicazione nelle quali occorre determinare come considerare l'indirizzo, se come unico elemento, o se come costituito da più elementi distinti. Per esempio si osservi come viene trattato l'Indirizzo nelle varie transazioni nei Casi di Studio 1, 2, e 3.

Le transazioni referenziano sempre l'indirizzo per intero o talvolta ne viene utilizzata solo una parte?

Nei Casi di Studio l'Indirizzo viene sempre utilizzato nella sua interezza. Non ci sono schermate o stampe dove è utilizzata una singola parte dell'Indirizzo a meno delle altre. Non ci sono neanche casi come ordinamenti, modifiche o criteri di filtro dove è utilizzata una singola parte dell'Indirizzo. Quindi il modo in cui è utilizzato nell'ambito dell'applicazione, suggerisce che l'Indirizzo sia considerato come un singolo DET. Se ci fosse una funzionalità per l'elenco dei siti che consente all'utente di elencare tutti i siti per una particolare città, stato o codice di avviamento postale, allora dovrebbe essere contato più di un tipo di elemento dati per l'Indirizzo. In base alle informazioni fornite, è chiaro che città, stato e codice di avviamento postale sono riconoscibili dall'utente come parti separate dall'indirizzo stesso. Quindi dovrebbero essere contati quattro DET per l'indirizzo (Via, Città, Stato e Codice di Avviamento Postale).

### Conteggio dei campi ripetuti

Spesso nelle applicazioni vengono mantenute occorrenze multiple di un tipo di elemento dati. In accordo con le regole per i DET nel Capitolo 6 della Parte 2, i campi ripetuti si contano una volta sola.

Dopo aver ridotto i campi ripetuti ad un singolo DET, verificare che i requisiti applicativi siano ancora soddisfatti. Si considerino i seguenti esempi:

#### **Esempio 1: ID Impiegato**

Nei Casi di Studio 1,2 e 3, si vedano i requisiti per Gestione Impiegato e le relazioni tra Impiegato e Dipendenti nel Diagramma Entità-Relazioni (ERD). In accordo con le conclusioni dei Casi di Studio, Impiegato è un file logico che include Dipendente. I File Logici Impiegato e Dipendente conterranno ciascuno l'ID Impiegato.

Nell'applicare la regola dei DET ripetuti contiamo ID Impiegato come un singolo DET per l'ILF Impiegato. Ora dobbiamo determinare se i requisiti utenti sono ancora soddisfatti.

I requisiti richiedevano di mantenere Dipendente come facente parte delle informazioni di Impiegato. I requisiti utente rimangono soddisfatti quindi si conferma il conteggio di un DET per l'ID Impiegato.

#### **Esempio 2: Ore Giornaliere Lavorate**

Un sistema di 'time reporting' tipicamente mantiene il numero di ore lavorate da una persona ogni giorno. Da un esame dei dati, si vede che le Ore Lavorate sono memorizzate separatamente per ogni giorno della settimana (Ore lavorate Lunedì, Ore lavorate Martedì, ecc.).

Applicando la regola dei DET ripetuti, contiamo un solo DET Ore Lavorate. Determiniamo se i requisiti utente sono sempre soddisfatti. Se l'applicazione tiene traccia delle sole Ore Lavorate, può soddisfare il requisito applicativo di mantenere le ore lavorate per ogni giorno?

No, non è possibile. Per soddisfare quel requisito è necessario contare un DET Giorno della Settimana. Ora l'applicazione ha la possibilità di tracciare separatamente le ore lavorate per ogni giorno (Ore Lavorate Lunedì, Ore Lavorate Martedì, ecc.). Quindi si contano due DET (Ore Lavorate e Giorno della Settimana).

### Conteggio dei campi di stato

Le applicazioni frequentemente tengono traccia dello stato corrente di un dato (per es. Attivo, Inattivo, Sospeso, Approvato, ecc.). Lo stato viene tipicamente aggiornato da varie transazioni all'interno dell'applicazione. Questo campo di stato potrebbe, o non potrebbe, essere fisicamente visibile all'utente attraverso le transazioni dell'applicazione. Si considerino i seguenti esempi:

#### **Esempio 1: Stato Inattivo**

Nei Casi di Studio 1, 2 e 3, ci si riferisce ad un indicatore di Stato. Quando viene cancellato un lavoro o un impiegato, i requisiti utente indicano che ogni relativo lavoro assegnato dovrebbe essere aggiornato ponendo lo stato a 'inattivo'.

Ad un esame della schermate relative all'assegnazione lavoro, risulta che lo Stato non è mai visualizzato e non è mai direttamente modificabile su ogni maschera. Nonostante la mancanza di visibilità esterna, lo Stato è comunque contato come DET per l'ILF Assegnazione Lavoro. Il fatto che esso appaia nel modello logico dei dati e nei requisiti utente suggerisce che esso è riconoscibile dall'utente nel File Logico, ma non nelle transazioni. Quindi, per il campo Stato, si conta un DET nell'ILF.

#### **Esempio 2: Stato Non Contato**

L'utente richiede la possibilità di cancellare gli Impiegati. Il personale tecnico per il momento non intende cancellare fisicamente i record; per questo motivo si realizza un flag di stato su Impiegato. Quando l'utente cancella un Impiegato, lo Stato viene impostato a 'inattivo'. L'utente non si rende conto del campo Stato su Impiegato. Pertanto il campo Stato non dovrebbe essere contato come DET.

## Conteggio delle date di sistema

Le applicazioni spesso conservano la data di sistema associata ai dati applicativi come riferimento sul periodo di validità dei dati. La Data di Sistema potrebbe essere memorizzata con diversi nomi (ultimo aggiornamento, ultima approvazione, ecc.) e spesso accompagnata dall'ID utente (ultimo aggiornamento a cura di, ultima approvazione a cura di). Tali date di sistema sono tipicamente aggiornate automaticamente dalle varie transazioni all'interno dell'applicazione. In molti casi la data di sistema deve essere mantenuta per scopi applicativi. Ci sono anche dei casi in cui la data di sistema viene mantenuta solo per ragioni tecniche.

Si considerino i seguenti esempi:

### **Esempio 1: Data di Validità**

Nei Casi di Studio 1, 2 e 3, ci si riferisce ad una Data di Validità. Quando viene cancellato un lavoro o un impiegato, i requisiti utente indicano che la data di fine validità, di ogni relativo lavoro assegnato, dovrebbe essere impostata alla corrente data di sistema.

Ad un esame della schermate relative all'assegnazione lavoro, risulta che la Data di Validità non è mai visualizzata e non è mai direttamente modificabile su ogni maschera. Nonostante la mancanza di visibilità esterna, la Data di Validità è comunque contata come DET per l'ILF Assegnazione Lavoro. Il fatto che essa appaia nel modello logico dei dati e l'esplicito riferimento nei requisiti utente suggeriscono che essa è riconoscibile dall'utente. Quindi, per il campo Data di Validità, si conta un DET nel File Logico.

### **Esempio 2: Recupero alla Data**

Il *tool* di Backup/Ripristino dati utilizzato dall'applicazione sfrutta la data di sistema memorizzata nelle tabelle per recuperare i dati da un particolare punto in poi. In questo caso il campo *recupera-alla-data* non è riconoscibile dall'utente e non dovrebbe essere contato.

### **Esempio 3: Data di Controllo (Audit)**

Il personale tecnico ha deciso che si dovrebbe mantenere traccia della data di sistema e dell'ID utente ogni volta che vengono modificati i dati, per risolvere ogni futura questione riguardante quando e da chi sia stata fatta la variazione e per essere in grado in seguito di fornire tali informazioni. In questo caso, la data di sistema non è riconoscibile dall'utente e non dovrebbe essere contata.

## Conteggio delle chiavi esterne

Spesso le applicazioni si occupano di mantenere le relazioni tra un'entità e l'altra. In alcuni casi le relazioni esistono per soddisfare requisiti di validazione dei dati, ma in altri casi esse rappresentano una interazione di business tra le due entità. Il concetto di Attribuzione della modellazione dei dati è ben rappresentato dall'assegnazione di chiavi esterne.

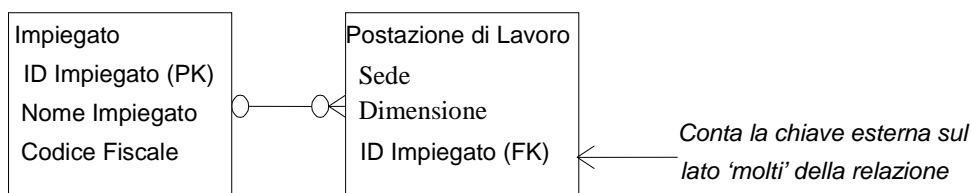
**Esempio 1:  
Sede  
(N):(1)**

I Casi di Studio 1, 2 e 3 includono i requisiti utente per l’Inserimento e la Modifica degli Impiegati: “La *sede* deve essere una sede valida nel Sistema Patrimoniale (FAS – Fixed Assets System)”.

Anche il diagramma ER rafforza questo requisito di business rappresentando la relazione che sussiste tra l’entità Impiegato e l’entità Sede. In accordo con il diagramma ER, un Impiegato può appartenere ad una sola Sede, ed una Sede può contenere molti Impiegati. In questo caso, il nome della Sede dovrebbe essere incluso tra gli attributi nelle tabelle logiche e fisiche. Si conta l’attributo Nome della Sede come un tipo di elemento dati (DET) per Impiegato.

**Esempio 2:  
Postazione  
di Lavoro  
(1):(N)**

Si consideri una variante del precedente esempio. Un impiegato potrebbe avere un numero illimitato di postazioni di lavoro. Una postazione di lavoro può essere occupata da non più di un impiegato alla volta. La postazione di lavoro deve avere un ID valido, così come identificato nella tabella POSTAZIONE\_DI\_LAVORO. Il diagramma ER rafforzerebbe di nuovo il requisito di business rappresentando la relazione tra l’entità POSTAZIONE\_DI\_LAVORO e l’entità IMPIEGATO. In accordo con il diagramma ER, un Impiegato può avere molte postazioni di lavoro, ma una Postazione di Lavoro può essere occupata da un solo impiegato. In questo caso l’ID Impiegato è un attributo di Postazione di Lavoro.

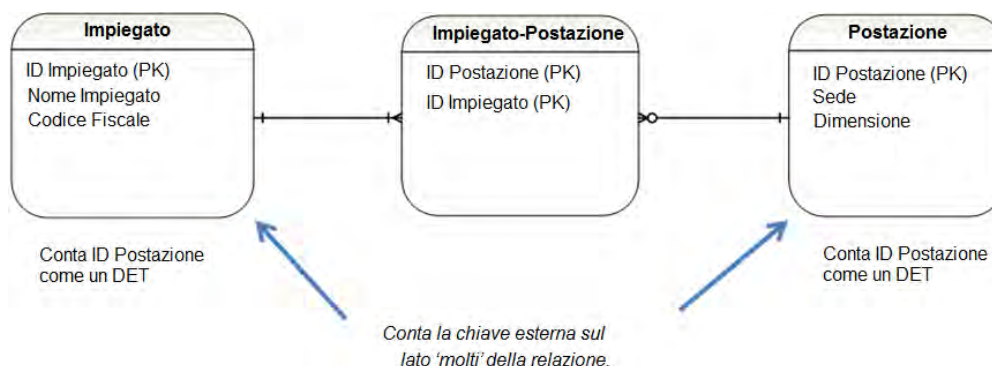


Nota: Non tutti gli attributi sono riportati nell’entità.

La relazione invece si riflette nella tabella POSTAZIONE\_DI\_LAVORO tramite l’identificazione dell’Impiegato che occupa il posto di lavoro. L’ID Impiegato è contato come un tipo di elemento dati (DET) in POSTAZIONE\_DI\_LAVORO.

**Esempio 3:  
Postazione  
di Lavoro  
N:M**

Si consideri una ulteriore variante all’esempio precedente. Un impiegato deve avere almeno una postazione di lavoro, ma potrebbe anche avere un numero illimitato di postazioni di lavoro. Una postazione di lavoro può essere occupata anche da più di un impiegato alla volta. La postazione di lavoro deve avere un ID valido, così come identificato nella tabella POSTAZIONE\_DI\_LAVORO. Di nuovo il diagramma ER rafforzerebbe ancora una volta il requisito di business tramite la rappresentazione della relazione tra l’entità POSTAZIONE\_DI\_LAVORO e l’entità IMPIEGATO. In accordo con il diagramma ER, un Impiegato può avere molte postazioni di lavoro e una Postazione di Lavoro può essere occupata da molti impiegati.



Nota: Non tutti gli attributi sono riportati nell'entità.

La relazione si riflette nella tabella *Impiegato – Postazione\_di\_Lavoro*, che contiene un'occorrenza per ogni associazione tra impiegato e postazione di lavoro. Questa conterrebbe l'ID Impiegato e l'ID Postazione di Lavoro come chiavi primarie. Come spiegato nelle sezioni “Identificare e Classificare i File Logici” e “Identificare i Tipi di Elementi Record”, la tabella *Impiegato – Postazione\_di\_Lavoro* non è contato come file logico, né come RET. L'ID Posto di Lavoro è contato come elemento dati (DET) in *Impiegato* e l'ID Impiegato è contato come elemento dati (DET) nell'entità ‘Postazione di Lavoro’.

## Osservazioni

L'identificazione del corretto numero di DET non influenza il *numero* di File Logici, ma esclusivamente la loro complessità. Pertanto questo passo influenza in modo limitato il risultato di un conteggio di function point, considerevolmente di meno rispetto al “Passo 1: Identificare e Classificare i File Logici”).

## Passo 4: Identificare i tipi di elementi record (RET)

I tipi di elementi record (RET – Record Element Type) rappresentano il punto di vista utente di *sottogruppi* coerenti di dati contenuti nel file logico identificato, i file logici sono stati trattati nella precedente sezione “Passo 1: Identificare i file logici”.

I tipi di elementi record tipicamente corrispondono a quelle diverse entità che sono state raggruppate all'interno di uno stesso file logico, come visto nella sezione “Passo 1: Identificare e Classificare i File Logici”. Comunque è necessaria una attenta revisione volta ad assicurare che l'utente li veda effettivamente come sottogruppi logici, per contarli quindi come RET.

## Tipo di Elemento Record: termini e definizioni

In questo capitolo si fa riferimento al modello logico dei dati nella terza forma normale (3NF) e non vengono considerate entità multiple create per ragioni tecniche; se non si dispone di un modello dei dati in questa forma, si dovrebbe provare a (de)normalizzare i dati per ricondurre il modello dei dati alla terza forma normale.

Le definizioni seguenti riprendono i concetti di modellazione dei dati già descritti nella sezione “Concetti di Modellazione dei Dati”.

<b>Entità Associativa</b>	Una entità che contiene ulteriori attributi che descrivono una relazione ‘multi-a-molti’ tra altre due entità. È nota anche come ‘intersezione tra le entità’.
<b>Entità Attributiva</b>	Una entità che contiene ulteriori attributi che descrivono una o più caratteristiche di un’altra entità.
<b>Sottotipo di Entità</b>	Un’entità derivata per suddivisione da un tipo di entità. Un sottotipo di entità (talvolta ‘entità figlio’) eredita tutti gli attributi e le relazioni dell’entità padre e può avere propri attributi e relazioni specifiche supplementari.

### Corrispondenza tra Concetti di Modellazione dei Dati e Terminologia Function Point

Si possono mettere in relazione i concetti della modellazione dei dati con l’analisi dei function point (FPA – Function Point Analysis) come mostrato nella seguente tabella:

<b>Concetti di Modellazione dei Dati</b>	<b>Termini Modell. Dati</b>	<b>Termini Database Relazionali</b>	<b>Termini FPA</b>	<b>Concetti FPA</b>
Gruppo di dati collegati che sono trattati come un’unità.	Record	Riga o Tupla	Tipo di Elemento Record (RET)	Il <i>Tipo di Elemento Record</i> (RET) è un sottogruppo di dati, riconoscibile dall’utente, all’interno di un ILF o EIF.
Raccolta ordinata di record dello stesso tipo.	File	Tabella	File Logico (File Logico Interno – ILF, o File Esterno di Interfaccia – EIF)	Il termine ‘file’ si riferisce a un gruppo di dati collegati logicamente e non alla loro implementazione fisica.



Inoltre la seguente tabella può aiutare a meglio comprendere il concetto di RET.

<b>Concetto Entità-Relazioni</b>	<b>Temine E-R</b>	<b>Termine FPA</b>	<b>Concetto FPA</b>
Oggetto dati principale, le cui informazioni sono memorizzate (persone, luoghi, cose o eventi); componente fondamentale, rilevante per l'utente, di cui viene conservata una raccolta ordinata di informazioni	Entità, o Tipo di Entità	File Logico	Il termine 'file' si riferisce a un gruppo di dati logicamente collegati e non alla loro implementazione fisica; se non ci sono altri sottogruppi, nel file si conta un unico Tipo di Elemento Record (RET).
Entità che contiene altri attributi significativi per l'utente, oltre quelli che descrivono la relazione con altre entità	Tipo di Entità Associativa	Potrebbe corrispondere a un file logico o a un Tipo di Elemento Record (RET); si veda la sezione "Analizzare le Entità Associate per Determinare i RET" per maggiori dettagli	Sottogruppo di dati, riconoscibile dall'utente, contenuto in un ILF o EIF, può essere opzionale o obbligatorio.
Un'entità che contiene ulteriori attributi che descrivono una più caratteristiche di un'altra entità.	Tipo di Entità Attributiva	Candidato Tipo di Elemento Record (RET); si veda la sezione "Analizzare le Entità Associate per Determinare i RET" per maggiori dettagli	Sottogruppo di dati, riconoscibile dall'utente, contenuto in un ILF o EIF, può essere opzionale o obbligatorio.
Un'entità derivata da un'altra entità, che eredita tutti gli attributi e relazioni dall'entità padre. Potrebbe anche avere, in aggiunta, propri attributi e relazioni	Sottotipo di Entità	Candidato Tipo di Elemento Record (RET); si veda la sezione "Analizzare le Entità Associate per Determinare i RET" per maggiori dettagli	Sottogruppo di dati, riconoscibile dall'utente, contenuto in un ILF o EIF, può essere opzionale o obbligatorio.

L'analisi dei function point si rivolge alle associative, attributive e ai sottotipi, in quanto sottogruppi di dati. Tali tipologie saranno esaminate nella successiva discussione su come contare i RET.

## Analizzare le entità associative per determinare i RET

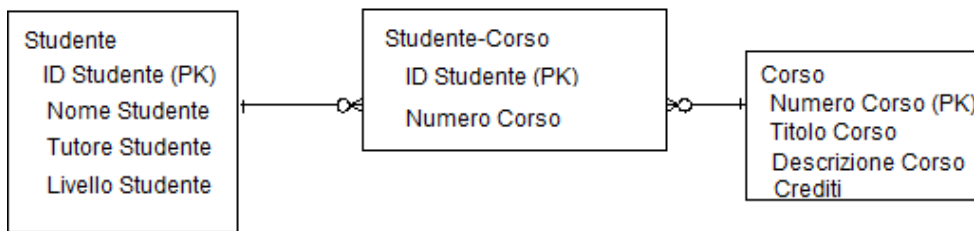
Le entità associative sono utilizzate come un mezzo per rappresentare le relazioni multi-a-molti. Questo tipo di entità viene spesso creata dal responsabile del modello dei dati per risolvere alcune delle regole applicative che mettono in relazione due diverse entità.

Esistono tre possibilità da prendere in considerazione nell'analisi delle entità associative.

### Caso 1

#### L'entità associativa *non* è contata come RET

Un impiegato ha il compito di gestire tutti gli studenti iscritti ai corsi e conoscere i corsi già frequentati dallo studente. Corso è un'entità e Studente è un'entità. Il responsabile del modello dei dati crea un'entità associativa chiamata Studente- Corso, come intersezione tra le due, questa entità associativa contiene solo le chiavi di ogni entità.

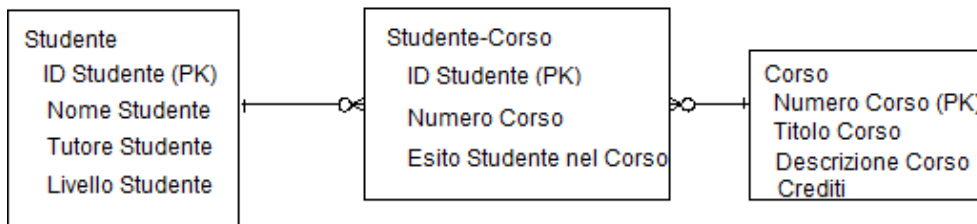


L'entità Studente – Corso non è considerata come RET né si conta come un file logico separato in quanto non contiene alcun elemento aggiuntivo diverso dalle due chiavi primarie (PK) delle entità che sono intersecate.

Studente è un file logico con 1 RET (Studente) e Corso è un file logico con 1 RET (Corso).

**Caso 2**L'entità associativa è contata come RET

Un impiegato ha il compito di gestire tutti gli studenti iscritti ai corsi. Inoltre necessita dell'informazione aggiuntiva sull'esito del corso sostenuto dallo studente(i). Corso è un'entità e Studente è un'entità. Il responsabile del modello dei dati crea un'entità associativa chiamata Studente-Corso, come intersezione tra le due, questa entità associativa contiene le chiavi di ogni entità e, in aggiunta, l'esito del corso sostenuto dallo studente.



Non esistono regole applicative che richiedono di trattare l'entità Studente-Corso in maniera indipendente, quindi Studente-Corso non soddisfa le regole per essere conteggiato come un file logico separato. In questo caso l'entità Studente-Corso è considerata come un RET in quanto contiene almeno un attributo riconoscibile dall'utente (\*), oltre le due chiavi primarie delle entità di cui rappresenta l'intersezione.

(\*) Nella sezione “Passo 1.5 Escludere le entità che non contengono attributi richiesti dall'utente”, gli attributi non-chiave che sono il risultato di considerazioni implementative, o che soddisfano requisiti tecnici, non sono considerati tipi di elementi dati.

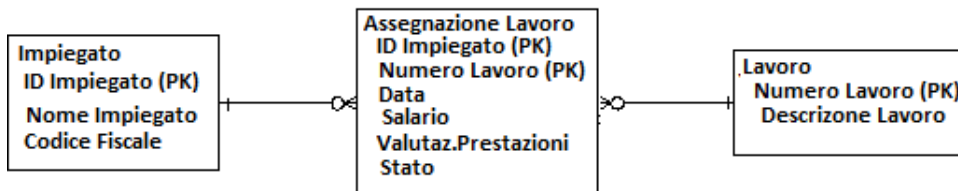
Un RET aggiuntivo in un file logico, per essere contato come tale, deve sempre contenere uno o più attributi univoci, in aggiunta alle chiavi primarie. Studente è un file logico con 2 RET (Studente e Studente-Corso) e Corso è un file logico con 2 RET (Corso e Studente-Corso).

Se i requisiti utente indicano che la relazione associativa tra le entità appartiene solo a uno dei file logici, allora il RET si dovrebbe contare solo per quel file logico.

**Caso 3**

L'entità associativa è contata come un *file logico* con un RET

L'ufficio RU mantiene le informazioni sugli Impiegati, sui Lavori e sulle Assegnazioni Lavori. Le Assegnazioni Lavori devono comunque essere mantenute, anche se un Impiegato non dovesse essere più associato necessariamente con un Lavoro o se un Lavoro non dovesse essere più adatto per l'assegnazione.



Nota: Non tutti gli attributi sono riportati nell'entità.

Sebbene Assegnazione Lavoro sia un'entità associativa, è qualcosa di più di una mappatura 'chiave-a-chiave' tra due entità, ed è anche più di un RET associato ad un file logico. Se una regola applicativa richiede che le informazioni di Assegnazione Lavoro siano conservate in maniera indipendente, allora Assegnazione Lavoro va considerato come file logico, come descritto nella sezione "Identificazione dei File Logici tramite il Metodo della (In)Dipendenza dell'Entità" (passo 1.3b).

Se invece non esistono regole applicative che richiedano che le informazioni di Assegnazione-lavoro siano conservate in maniera indipendente, allora questa entità va contata come descritto nella precedente Situazione 2.

## Analizzare le entità attributive per determinare i RET

Un'entità attributiva è un tipo di entità che contiene ulteriori attributi che descrivono una o più caratteristiche di un'altra entità. Per definizione è una logica estensione di un'altra entità; nell'analisi dei function point un'entità attributiva rappresenta un Tipo di Elemento Record di quella entità.

Un'entità attributiva si considera o come un RET del file logico che è stato ulteriormente definito (Caso 1) o come un'estensione del file logico (Caso 2).

### Caso 1

#### Entità attributiva *opzionale*

Un impiegato potrebbe sottoscrivere delle indennità. Nel nostro modello dei dati Impiegato è un'entità. Indennità-Impiegato è un'entità attributiva di impiegato e contiene informazioni sul tipo di indennità che l'impiegato ha deciso di sottoscrivere. Indennità-Impiegato non può esistere senza Impiegato ed è logicamente correlata in questo modo:



Indennità-Impiegato è contato come un RET in quanto si tratta di un'entità attributiva opzionale. Impiegato è un file logico con 2 RET, Impiegato e Indennità-Impiegato.

### Caso 2

#### Entità attributiva *obbligatoria*

Un sistema di gestione vendite deve mantenere le informazioni su ciascun prodotto e relativi dati sul prezzo. Prodotto è un'entità. Informazioni Prezzi Prodotti è un'entità attributiva relativa al prezzo che contiene: prezzo precedente, prezzo corrente, prezzo futuro prospettato e data di validità dei prezzi. Informazioni Prezzi Prodotti non ha senso senza Prodotto ed è logicamente correlata in questo modo:



Informazioni Prezzi Prodotti *non* è contato come un RET. Prodotto è un file logico con 1 RET che contiene Prodotto e Informazioni-Prodotto-Prezzo.

## Analizzare i sottotipi di entità per determinare i RET

Un sottotipo di entità (o ‘entità figlio’) è un’entità derivata per suddivisione da un tipo di entità. Un sottotipo di entità eredita tutti gli attributi e le relazioni dell’entità padre e può avere propri attributi e relazioni specifiche supplementari. Le regole di modellazione dei dati affermano che un’entità può avere associato a se stessa un numero qualsiasi di sottotipi indipendenti, che possono essere opzionali o obbligatori. Ogni sottotipo (‘figlio’) può avere solo un generico padre. Nella modellazione dei dati, sebbene padre e sottotipo (‘figlio’) siano rappresentati tramite entità separate, esse fanno logicamente parte della stessa entità.

Nell’analizzare i sottotipi di entità in un modello di dati, esamina la relazione richiesta con l’entità padre, come mostrato nelle seguenti situazioni.

### Caso 1

Sottotipo di entità che rappresenta un sottogruppo e quindi è contata come RET

Un impiegato deve essere o un dipendente fisso o un impiegato a tempo determinato, ma non può essere entrambi. I dati comuni all’impiegato sono pertinenti per tutti gli impiegati e sono obbligatori. In più questi dati comuni sono ereditati dai sottotipi obbligatori di entità, dipendenti fissi e impiegati a tempo determinato.



Analizzando questo modello dal punto di vista dei function point, si identificano due sottogruppi per il file logico Impiegato:

- ❑ I dati sui dipendenti fissi includono sia le informazioni relative ai dipendenti fissi, sia le informazioni comuni a tutti gli impiegati.
- ❑ I dati sugli impiegati a tempo determinato includono sia le informazioni relative agli impiegati a tempo determinato, sia le informazioni comuni a tutti gli impiegati.

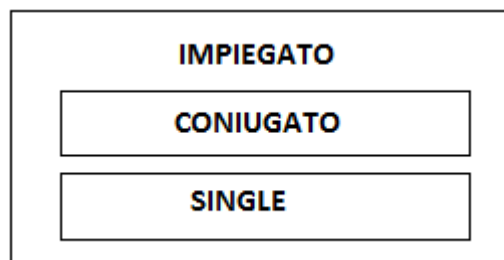
In questo caso c’è un solo file logico (Impiegato) con due RET, dipendenti fissi e impiegati a tempo determinato.

**Caso 2**

Sottotipo di entità che non rappresenta un sottogruppo e quindi non è contata come RET

Se c'è un unico attributo di differenza tra sottotipi di entità, occorre considerare seriamente se veramente esiste un sottogruppo separato che potrebbe costituire un Tipo di Elemento Record (RET). Un solo ed unico attributo opzionale potrebbe non risultare come RET dal punto di vista dell'utente, anche se nel modello logico dei dati è raffigurato come un sottotipo di entità.

Lo stato civile di un impiegato potrebbe essere coniugato o non coniugato. Se coniugato viene memorizzato il nome del coniuge. Nonostante il nome del coniuge possa essere raffigurato come un sottotipo di entità nel modello dei dati, in realtà il nome del coniuge è solo un attributo opzionale all'interno del file logico Impiegato.



Un solo attributo di differenza in questo caso non costituisce una significativa differenza tra impiegati coniugati e non, dal punto di vista del business.

**Suggerimenti**

Esamina attentamente il modello dei dati. In caso di dubbio si verifichi con l'utente il vero intento dei diversi sottotipi di entità. L'analista dei dati crea un modello dei dati raffigurando la sua visione del mondo dell'utente. In pratica dipende dal punto di vista dell'utente se i sottotipi di entità siano rilevanti per l'utente stesso; nel qual caso dovrebbero essere conteggiate come RET.

Se per questi sottotipi di entità ci sono transazioni di inserimento/modifica distinte con attributi differenti, questa è un'indicazione che si potrebbero avere dei RET separati a fronte di quelle entità.

## Casi vari

Se non si dispone di un modello dei dati su cui lavorare, si potrebbero incontrare alcuni dei casi seguenti. Di seguito sono forniti alcuni suggerimenti aggiuntivi per il conteggio.

### Gruppi / dati ripetuti

I gruppi ripetuti sono istanze di occorrenze multiple degli stessi dati, che potrebbero essere ripetuti più volte all'interno dello stesso file logico.

#### Caso 1

##### I gruppi ripetuti sono contati come RET

Il gruppo di dati relativo all'Ordine è composto dall'Intestazione Ordine e può avere diverse occorrenze relative alle Riga Ordine. La Riga Ordine contiene qualcosa di più di un unico attributo. Intestazione Ordine e Riga Ordine rappresentano due sottogruppi separati. Si contano 2 RET per il file logico Ordine.

#### Caso 2

##### I dati ripetuti non sono contati come RET

Un campo replicato (DET) non dovrebbe dare luogo a un sottogruppo separato o RET. Per esempio, un Impiegato potrebbe avere diversi numeri di conto bancari. Ma questo *non* implica che ci siano 2 RET per Impiegato ('tutti i dati meno i numeri di conto bancari' e 'i numeri di conto bancari').

### Osservazioni

In caso di dubbio *non* contare un sottogruppo di informazioni come RET.

L'identificazione del corretto numero di RET non influisce sul *numero* di file logici identificati, ma ha influenza solo sulla complessità dei file logici. Perciò, sebbene questo passo influenzi di fatto il risultato di un conteggio di function point, lo fa in grado minore rispetto all'identificazione dei file logici nella sezione "Passo 1: Identificazione e Classificazione dei File Logici".



## Considerare i tipi di elemento record congiuntamente ai file logici tramite la (in)dipendenza dell'entità

Dopo aver discusso sia DET che RET, è possibile esplodere la tabella mostrata in precedenza nella sezione “Riepilogo: da Entità a File Logici tramite l'analisi della (In)Dipendenza dell'Entità” includendo pertanto anche le indicazioni su come e quanti DET e RET contare.

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, 1RET e DET ciascuno
1 : N	B è un'entità dipendente da A	1 LF, 2RET, somma DET
	B è un'entità indipendente da A	2 LF, 1RET e DET ciascuno
1 : (N)	B è un'entità dipendente da A	1 LF, 2RET, somma DET
	B è un'entità indipendente da A	2 LF, 1RET e DET ciascuno
(1) : N	A è un'entità dipendente da B	1 LF, 2RET, somma DET
	A è un'entità indipendente da B	2 LF, 1RET e DET ciascuno
(1) : (1)	(A e B sono indipendenti)	2 LF, 1RET e DET ciascuno
1 : 1	(A e B sono dipendenti)	1 LF, 1RET, somma DET
1 : (1)	B è un'entità dipendente da A	1 LF, 1 o 2 RET, somma DET
	B è un'entità indipendente da A	2 LF, 1RET e DET ciascuno
(N) : (M)	(A e B sono indipendenti)	2 LF, 1RET e DET ciascuno
N : M	B è un'entità dipendente da A	1 LF, 2RET, somma DET
	B è un'entità indipendente da A	2 LF, 1RET e DET ciascuno
N : (M)	B è un'entità dipendente da A	1 LF, 2RET, somma DET
	B è un'entità indipendente da A	2 LF, 1RET e DET ciascuno

### Nota

- “1 RET e DET ciascuno” significa: valutare entrambe le entità per conto proprio
- “Somma DET” significa: contare tutti gli attributi unici, non ripetuti delle entità collegate insieme
- Contare la chiave esterna sul lato “molti” delle relazioni.
- In alcuni casi, anche più di due entità possono costituire un file logico; allora si dovrebbero contare più di 2 RET.

### Legenda

LF = File Logico (Logical File – ILF or EIF).  
 (..) = Lato opzionale della relazione  
 RET = Tipo di Elemento Record  
 DET = Tipo di Elemento Dati

## Bibliografia

Le seguenti fonti sono state consultate o citate in questo capitolo:

Booch, Grady, James Rumbaugh, Ivar Jacobson. The Unified Modeling Language User Guide. Reading: Addison-Wesley, 1994. ISBN: 0-2015-7168-4.

NESMA. Definitions and Counting Guidelines for the Application of Function Point Analysis: A Practical Manual, Version 2.2. (NESMA, 2003). ISBN: 978-90-76258-17-1.

Nota: Tale manuale è anche chiamato NESMA Counting Practices Manual che descrive lo standard metodologico FPA, e molti degli aspetti correlati all'applicazione dei FPA. Si può usare insieme al manuale IFPUG. Per ulteriori informazioni, vedere il sito della NESMA: [www.nesma.org](http://www.nesma.org).

Garmus, David, David Herron. Function Point Analysis: Measurement Practices for Successful Software Projects. Boston: Addison-Wesley Information Technology Series, 2001. ISBN: 0-201-69944-3.

Martin, James, Carma McClure. Diagramming Techniques for Analyst and Programmers. Englewood Cliffs: Prentice-Hall, Inc., 1985. ISBN: 0-132-087944.

Modern Language Association of America. MLA Handbook for Writers of Research Papers, Fifth Edition. Boston: Addison Wesley, 1999.

Reingruber, Michael C. and William W. Gregory. The Data Modeling Handbook: A Best- Practice Approach to Building Quality Data Models. Canada: John Wiley & Sons, Wiley-QED Publication, 1994. ISBN: 0-471-05290-6.

Silverman, Len, W. H. Inmon, Kent Graziano. The Data Model Resource Book: A Library of Logical Data Models and Data Warehouse Design. Boston: Addison-Wesley, Inc. Out of Print: AISN: 0-471-15364-8.

Simsion, Graeme. Data Modeling Essentials: Analysis, Design, and Innovation. Boston: International Thomson Computer Press, 1994. ISBN: 1-850-932877-3.

Schuldt, Gary. "Information Modeling for Information Systems Analysts", A workshop at AT&T Bell Laboratories. Holmdel, N.J., May, 1992.

Teorey, Toby J. Database Modeling & Design: The Fundamental Principles, Second Edition. San Francisco: Morgan Kaufmann Publishers, Inc., 1994. ISBN: 1-558-60291-1.

Questa pagina è lasciata intenzionalmente vuota.

---

### Dati condivisi

---

**Introduzione** Questo capitolo fornisce linee guida aggiuntive a supporto dell'identificazione dei file d'interfaccia esterni (EIF) e dei file logici interni (ILF), così come dei file di movimenti (*transaction file*), laddove due o più applicazioni interagiscono (ovvero, assistenza, chiarimenti e indicazioni per contare i dati condivisi tra più applicazioni).

**Contenuti** Questo capitolo include le seguenti sezioni:

Argomento	Pagina
<b>Contare i Dati Condivisi tra Applicazioni</b>	<b>3-2</b>
<b>Scenari di conteggio – Gruppo 1</b>	<b>3-7</b>
Scenario 1: Lettura	3-7
Scenario 2: Copia immagine statica	3-9
Scenario 3: Copia / Carica immagine – senza ulteriore elaborazione	3-11
Scenario 4: Copia Immagine / Carica unica tabella fisica – senza ulteriore elaborazione	3-13
Scenario 5: Copia e unione	3-15
Scenario 6: Cattura da schermo	3-17
<b>Scenari di conteggio – Gruppo 2</b>	<b>3-18</b>
Scenario 7: Aggiornamento del medesimo archivio dati	3-18
Scenario 8: Dati di transazione standard	3-20
<b>Riepilogo</b>	<b>3-22</b>

## Contare i dati condivisi tra applicazioni

### Applicazioni che condividono dati

Le applicazioni che condividono dati con altre applicazioni:

- referenziano o utilizzano i dati per completare una transazione all'interno dell'applicazione che riceve o accede ai dati, oppure
- mantengono i file logici interni all'interno dell'applicazione che riceve o accede ai dati.

### Metodi per condividere i dati

I dati condivisi, utilizzati dai processi elementari all'interno di un'applicazione per mantenere dati in un file logico interno o per presentare dati all'utente, possono essere trasferiti tramite:

- schermate online (per es. cattura dati da schermo),
- accesso diretto a file di dati di altre applicazioni,
- trasferimento di file,
- richieste di informazioni online in tempo reale,
- applicazioni web.

Per analizzare correttamente queste implementazioni, gli utenti devono considerare l'intento primario ed avere una comprensione comune dei termini che rappresentano le varie tecniche implementative.

### Intento Primario

Il concetto di intento primario è utile per l'identificazione dei file logici interni e dei file d'interfaccia esterni in quanto è legato all'uso dei dati all'interno dell'applicazione oggetto di analisi. L'intento primario si riferisce all'obiettivo più significativo o importante che la funzione intende perseguire. La definizione di intento primario è "il primo intento per importanza". Perciò è importante, nella discussione di ciascuno scenario, **determinare l'intento primario**. L'implementazione fisica non ha impatto sull'intento primario e non dovrebbe quindi influenzare l'analisi.

### Definizione & intento primario (ILF/EIF)

#### File Logico Interno:

Un file interno logico (ILF) è un gruppo di dati logicamente collegati o di informazioni di controllo, riconoscibili dall'utente, mantenuti all'interno del confine dell'applicazione sottoposta al conteggio. L'intento principale di un ILF è di contenere dati mantenuti da uno o più processi elementari dell'applicazione sottoposta al conteggio.

Nota: Il termine mantenere si riferisce alla capacità di aggiungere, cambiare o cancellare i dati attraverso un processo elementare. Gli esempi comprendono, ma non sono limitati a aggiungere, cambiare, cancellare, popolare, rivedere, aggiornare, assegnare e creare.

### File d'Interfaccia Esterno:

Un file esterno di interfaccia (EIF) è un gruppo di dati logicamente collegati o di informazioni di controllo, riconoscibili dall'utente, referenziati dall'applicazione ma mantenuti all'interno del confine di un'altra. L'intento principale di un EIF è quello di contenere dati referenziati da uno o più processi elementari situati nei confini dell'applicazione sottoposta al conteggio. Un EIF contato per un'applicazione deve essere un ILF in un'altra applicazione.

## Termini comuni

In questo documento si fa uso dei seguenti termini comuni per descrivere tecniche di implementazione fisica:

Termine	Utilizzo nel Documento
<b>Copia</b>	Definizione IEEE <sup>1</sup> : “(1) Lettura di dati da una fonte, senza modifica dei dati di origine, e scrittura degli stessi dati in altra posizione, in un formato fisico che può differire da quello di origine. Per es. copiare dati da un disco magnetico su un nastro magnetico. (2) Il risultato di un processo di copia come descritto sopra. Per es. una copia di un file di dati”.
<b>File</b>	Definizione IEEE: “[...] insieme di record collegati, trattati come un'unità. Per es. un file potrebbe consistere di un insieme di record di fatture.”
<b>Immagine (Image)</b>	Una replica esatta di un altro oggetto, file, o tabella, creata solitamente per mezzo di una funzione di “utilità”.
<b>Caricamento (Load)</b>	Definizione IEEE: “[...] copiare istruzioni o dati dalla memoria esterna alla memoria interna [...]”
<b>Unione (Merge)</b>	File multipli con gli stessi elementi dati, consolidati in un unico file.
<b>Aggiornamento (Refresh)</b>	Il processo che ricrea un insieme di dati per renderli attuali rispetto alla fonte di origine.

<sup>1</sup> Institute of Electrical and Electronics Engineers (IEEE) Std 610.12-1990, *IEEE Standard Glossary of Software Engineering Terminology*

## Organizzazione degli scenari di conteggio

L'analisi dei function point si basa spesso sulle descrizioni degli sviluppatori delle caratteristiche fisiche di un'applicazione installata. Questo capitolo affronta le descrizioni fisiche frequentemente incontrate da un analista dei function point per fornire supporto alla corretta interpretazione di molte di queste interfacce tra applicazioni.

Questo capitolo utilizza vari scenari come supporto all'analisi di varie situazioni di dati condivisi.

### Approccio

Questo capitolo utilizza il seguente approccio nella discussione dei dati condivisi:

- **Descrizione** – Una descrizione ad alto livello dell'esempio discusso.
- **Scenario** – Si presenta un esempio che spesso descrive un'attività o transazione fisica riguardante dei file trasferiti tra due applicazioni; per es. condivisione di dati.
- **Diagramma dello Scenario** – Lo scenario è raffigurato graficamente, in supporto al confronto con una situazione o scenario simile. La freccia nei diagrammi rappresenta la direzione del flusso di dati, non l'applicazione che inizializza l'interfacciamento.
- **Interpretazione del Conteggio** – Viene fornita un'interpretazione del conteggio dello scenario, che include una discussione dell'esempio e come l'esempio dovrebbe essere contato, insieme con qualsiasi assunto fatto sull'intento primario.
- **Diagramma della Soluzione** – La soluzione è raffigurata graficamente.
- **Riepilogo del Conteggio** – Si riepilogano in una tabella le funzioni di tipo dati e di tipo transazionale applicabili per ciascuna applicazione.
- **FAQ/Varianti** – Se applicabili, alcune varianti tipiche dello scenario possono essere riportate a fine discussione.

### Simboli usati nei diagrammi delle soluzioni

Nei diagrammi delle soluzioni si trovano i seguenti simboli:

☑ sopra un tipo di component, indica che la componente è **contata** per l'applicazione;

☒ sopra un tipo di component, indica che la componente **non andrebbe contata** per lo scenario;

⇐ nel diagramma, indica la direzione del flusso di dati, non l'applicazione che inizializza l'interfacciamento.



### Convenzioni terminologiche degli scenari

Per omogeneità, tutti gli scenari utilizzano le seguenti convenzioni terminologiche:

Termine	Descrizione
<b>Applicazione A</b>	L'applicazione sorgente dei dati di riferimento/transazione.
<b>Applicazione B</b>	L'applicazione destinataria dei dati di riferimento/transazione.
<b>File X</b>	Un ILF contato nell'Applicazione A.
<b>File X' ("X primo")</b>	Un EIF contato nell'Applicazione B, che è un sottoinsieme dei dati del File X.
<b>File Y</b>	Un ILF contato nell'Applicazione B.
<b>File Z</b>	Un file di trasferimento dei dati. Questo file è generato dall'Applicazione A e letto (elaborato) dall'Applicazione B.
<b>Confini</b>	L'Applicazione A e l'Applicazione B rappresentano due applicazioni <u>distinte</u> , e quindi rappresentano due confini distinti.

### Sintesi degli scenari

I seguenti scenari non rappresentano una lista completa dei vari modi in cui i dati condivisi sono implementati, ma forniscono una guida per molte delle situazioni incontrate. Comprendere questi esempi faciliterà la comprensione degli ulteriori scenari che si possono incontrare.

Gli scenari si concentrano sulle situazioni in cui i dati richiesti per completare i processi elementari all'interno dell'Applicazione B sono ottenuti a partire dall'Applicazione A. L'Applicazione B è l'applicazione oggetto di misurazione. Gli scenari si dividono logicamente in due gruppi, ognuno dei quali ha varie implementazioni:

**GRUPPO 1** L'intento primario è che l'Applicazione B referenzi i dati mantenuti dall'Applicazione A. Si affrontano due aree, quella funzionale e quella non-funzionale.

**Funzionale** Per Ragioni Funzionali (requisiti aziendali), le applicazioni condividono dati nei seguenti scenari:

Numero	Scenario	Descrizione Sintetica
<b>1</b>	<b>LETTURA</b>	L'Applicazione B accede fisicamente ai dati dell'applicazione A. Questo esempio è attualmente documentato nel CPM.

Numero	Scenario	Descrizione Sintetica
<b>2</b>	<b>COPIA IMMAGINE STATICA</b>	L'Applicazione A genera un'immagine copia di un archivio di dati, che riflette lo stato attuale dei dati in un certo momento e rimane all'interno del suo confine.

**Non-Funzionale** Per Ragioni Non-Funzionali (prestazioni, sicurezza, ecc.), l'Applicazione B deve utilizzare i dati dell'Applicazione A e lo fa nei seguenti modi:

Numero	Scenario	Descrizione Sintetica
<b>3</b>	<b>COPIA / CARICA IMMAGINE, senza elaborazione</b>	L'Applicazione A genera un'immagine copia senza alcun trattamento logico ulteriore e la invia all'Applicazione B; l'Applicazione B carica la copia senza alcun trattamento logico ulteriore.
<b>4</b>	<b>COPIA / CARICA IMMAGINE, sottoinsieme di un ILF</b>	L'Applicazione A genera una copia immagine di un sottoinsieme (per es. un RET) senza alcun trattamento logico aggiuntivo e la invia all'Applicazione B. L'Applicazione B carica il RET senza alcun trattamento logico ulteriore.
<b>5</b>	<b>COPIA UNIONE "Refresh"</b>	Archivi memorizzati in due applicazioni sono copiati come immagini e uniti per formare un file che viene caricato in una terza applicazione.
<b>6</b>	<b>CATTURA DA SCHERMO</b>	L'Applicazione B accede alle transazioni a schermo dell'Applicazione A per referenziare/ottenere dati necessari per elaborare una transazione.

**GRUPPO 2** L'intento primario è che l'Applicazione B mantenga i propri dati a partire dai dati mantenuti dall'Applicazione A.

Numero	Scenario	Descrizione Sintetica
<b>7</b>	<b>MANTENERE ARCHIVIO DATI IN COMUNE</b>	Lo stesso archivio di dati è mantenuto da due applicazioni distinte. Questo esempio è attualmente documentato nella Parte 4, Capitolo 1, Esempi di Conteggio delle Funzioni di Tipo Dati.
<b>8</b>	<b>DATI DI TRANSAZIONE STANDARD</b>	Dati di transazione sono forniti dall'applicazione sorgente.

## Scenari di conteggio – Gruppo 1

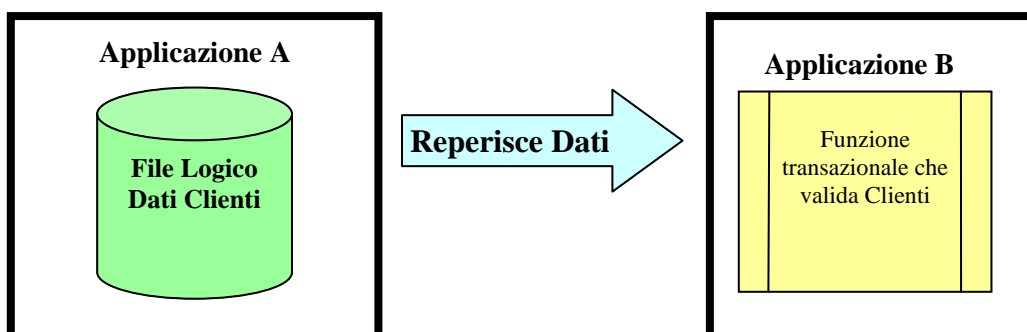
In ognuno degli scenari da 1 a 6, l'intento primario è che un'applicazione referenzi i dati mantenuti da un'altra o altre applicazioni; esso può essere implementato come segue:

### Scenario 1: Lettura

**Descrizione** L'Applicazione B accede fisicamente ai dati dell'applicazione A per eseguire un'interrogazione.

**Scenario** Una transazione elaborata dall'Applicazione B richiede informazioni da un archivio di dati mantenuto all'interno dell'Applicazione A. L'Applicazione B è responsabile per l'accesso ai dati nell'Applicazione A, e mantiene il software per tale accesso.

#### Diagramma

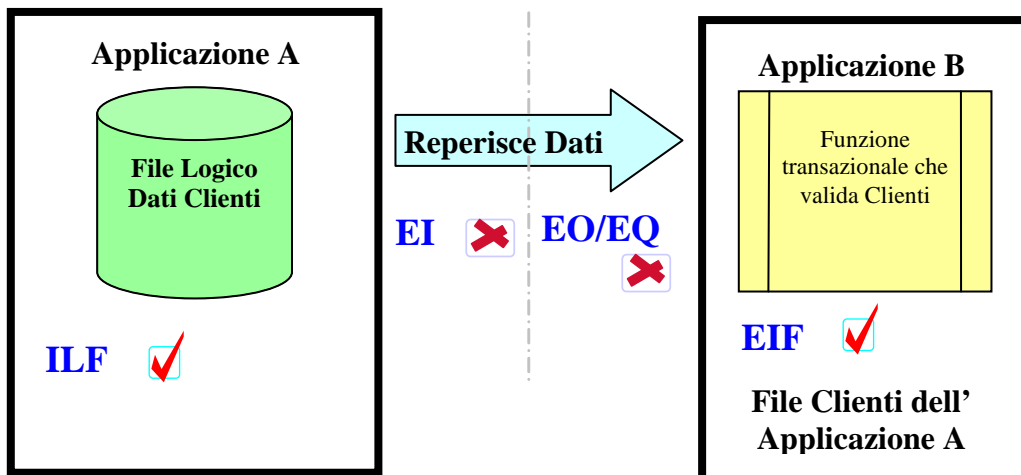


#### Interpretazione del conteggio

**Applicazione A:** Dal punto di vista dell'Applicazione A, non c'è alcun requisito di invio di dati. I dati sono disponibili nell'Applicazione A. Non si riconosce alcun merito all'Applicazione A per la transazione eseguita dall'Applicazione B, sebbene il file di dati sia un ILF per l'Applicazione A.

**Applicazione B:** Dal punto di vista dell'Applicazione B, sia logico che fisico, c'è un solo archivio di dati coinvolto. L'Applicazione B conta l'archivio di dati, che risiede nell'Applicazione A, come un EIF. L'Applicazione B conta anche questo file di dati come un FTR nella transazione.

**Diagramma della soluzione**



**Riepilogo del conteggio**

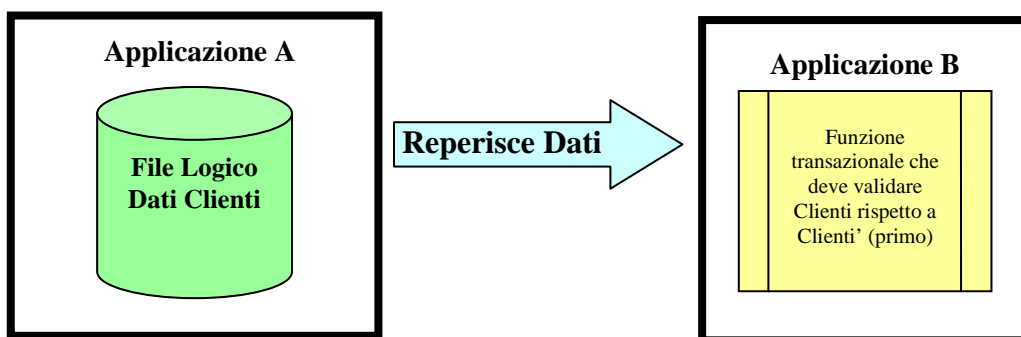
	<b>ILF</b>	<b>EIF</b>	<b>EI</b>	<b>EO/EQ</b>	<b>Note</b>
Applicazione A	✓				
Applicazione B		✓			Clienti è anche contato come un FTR in una funzione transazionale

## Scenario 2: Copia immagine statica

**Descrizione** L'Applicazione A genera un'immagine copia di un ILF, che riflette lo stato attuale dei dati in un certo momento e rimane all'interno del suo confine.

**Scenario** Nell'industria bancaria, si stabiliscono quotidianamente delle transazioni finanziarie tra tutti gli istituti finanziari. Le conseguenti transazioni finanziarie dei clienti sono validate rispetto ai bilanci finanziari dei clienti in funzione di questa impostazione. A supporto di questo requisito aziendale, l'Applicazione A crea periodicamente una copia dei dati del file logico Clienti nel file Clienti' ("primo") affinché altre applicazioni possano referenziarlo. Clienti' ("primo") rimane all'interno del confine dell'Applicazione A. Possono esserci delle differenze fra i dati attuali del file Clienti e i dati del file Clienti'. Il Sistema B utilizza il file Clienti' ("primo").

### Diagramma

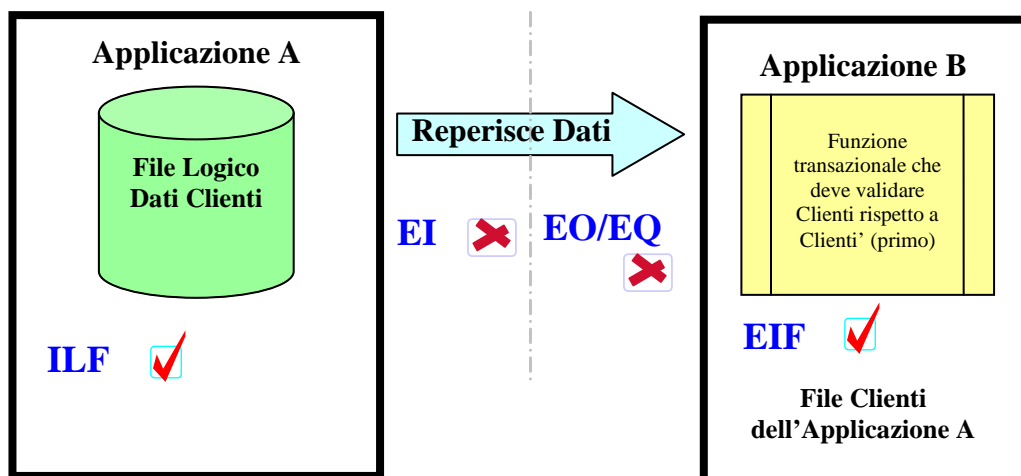


### Interpretazione del conteggio

**Applicazione A:** Dal punto di vista dell'Applicazione A, Clienti è un file logico interno. Clienti' (primo) non è contato come un distinto ILF, nè è contato come un RET di Clienti. Clienti' (primo) è solo una fotografia di Clienti in un certo momento.

**Applicazione B:** Dal punto di vista dell'Applicazione B, Clienti è un file d'interfaccia esterno ed è anche contato come un FTR per la transazione dell'Applicazione B.

**Diagramma della soluzione**



**Riepilogo del conteggio**

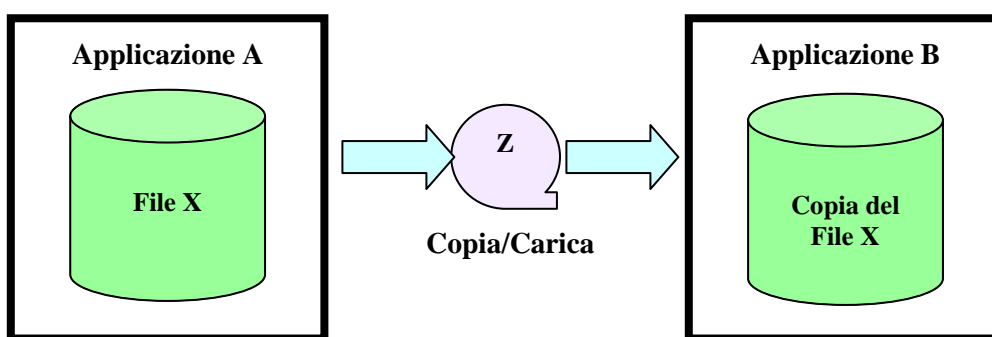
	ILF	EIF	EI	EO/EQ	Note
Applicazione A	☑				
Applicazione B		☑			Clienti è anche contato come un FTR in una funzione transazionale

### Scenario 3: Copia / Carica immagine – senza ulteriore elaborazione

**Descrizione** L'Applicazione A genera un'immagine copia senza alcun trattamento logico ulteriore e la invia all'Applicazione B; l'Applicazione B carica la copia senza alcun trattamento logico ulteriore.

**Scenario** L'Applicazione B richiede la possibilità di accedere al File X dell'Applicazione A solo per validazione e referenza. L'Applicazione B richiede (per es. per motivi prestazionali, ecc.) che l'Applicazione A invii un file completo all'Applicazione B. L'archivio di dati esistente nell'Applicazione B è “refreshato” ogni volta con questa copia.

#### Diagramma



#### Interpretazione del conteggio

Dal punto di vista dell'Applicazione A, questo trasferimento di dati è una soluzione tecnica escogitata per soddisfare il requisito aziendale che l'Applicazione B debba avere accesso, per scopi di reperimento dati, al File X dell'Applicazione A.

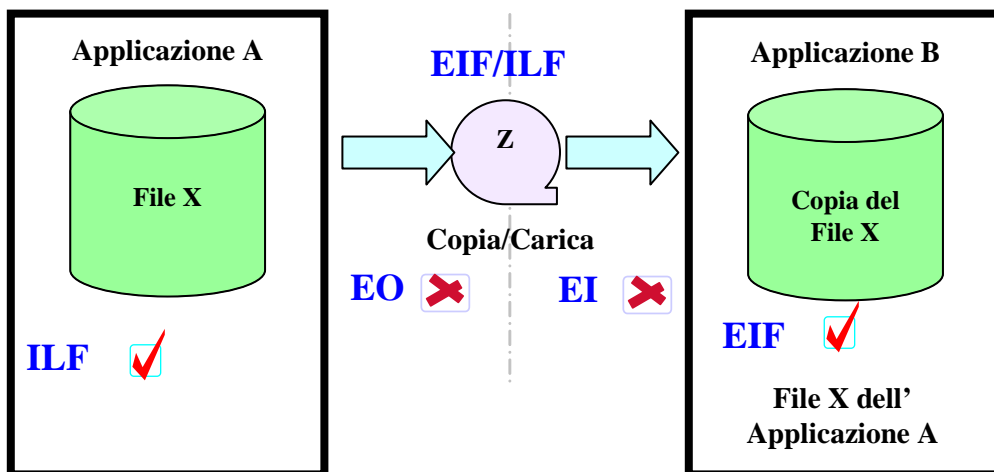
Dal punto di vista logico l'archivio di dati rimane nell'Applicazione A. In questo caso, copiare l'archivio di dati da un'applicazione all'altra è la soluzione di un requisito utente non-funzionale (per es. i dati dell'Applicazione A non sono disponibili quando sono richiesti dall'Applicazione B).

L'*intento primario* per B è di referenziare i dati che sono dal punto di vista logico in A. Un'ulteriore indicazione sarebbe il fatto che il file nell'Applicazione B è “refreshato” ogni volta con la copia. Inoltre, non è svolto alcun trattamento logico né nell'Applicazione A né nell'Applicazione B.

*Transazioni* – Le transazioni di trasferimento dei dati, scaricamento dall'Applicazione A e caricamento nell'Applicazione B, fanno parte della soluzione tecnica e non sono contati in nessuna delle due applicazioni. In pratica, quando si conta l'Applicazione A a sè stante, potrebbe non essere evidente all'Analista dei Function Point che esiste per soddisfare un requisito utente non-funzionale, e potrebbe essere contata, non correttamente, come un EO/EQ. Nessuna delle due applicazioni conta il File Z come una funzione transazionale.

*File* – È coinvolto un unico file logico. L'Applicazione A conta il File X come un ILF. L'Applicazione B conta la propria versione copia del File X come un EIF. Nessuna delle due applicazioni conta il File Z come una funzione di tipo dati.

### Diagramma della soluzione



### Riepilogo del conteggio

	ILF	EIF	EI	EO/EQ	Note
Applicazione A	☑				
Applicazione B		☑			

### FAQ, Ulteriori varianti

**Domanda:** Cosa cambia se un file logico dell'Applicazione A è composto di molteplici tabelle fisiche e l'Applicazione A fornisce copie individuali di più di una tabella all'Applicazione B?

**Risposta:** L'EIF è identificato allo stesso modo dello scenario sopra descritto; solo i campi utilizzati sono contati come DET.

**Domanda:** Cosa cambia se si ha un archivio di dati partizionato?

**Risposta:** È segmentato per motivi prestazionali, ma è ancora un solo file di dati dal punto di vista logico (implementazione fisica).

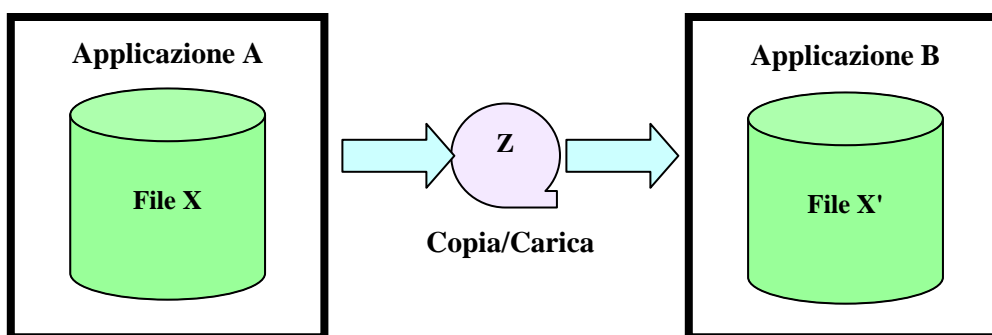


## Scenario 4: Copia Immagine / Carica unica tabella fisica – senza ulteriore elaborazione

**Descrizione** L'Applicazione A genera una copia immagine di un sottoinsieme di una tabella fisica contenuta in suo file logico senza alcun trattamento logico aggiuntivo e la invia all'Applicazione B. L'Applicazione B carica la tabella fisica senza alcun trattamento logico ulteriore.

**Scenario** L'Applicazione B richiede (per es. per motivi prestazionali, ecc.) la possibilità di accedere a una porzione del File X dell'Applicazione A solo per validazione e referenza. L'Applicazione A invia una tabella fisica contenuta in un file logico all'Applicazione B. La vista esistente di tale tabella fisica nell'Applicazione B è “refreshata” ogni volta con questa copia.

### Diagramma



### Interpretazione del conteggio

Poichè i dati sono una copia immagine dei dati dell'Applicazione A, la Tabella del File X' fa parte del file logico X dell'Applicazione A. L'Applicazione B conta il File X' (con solo gli elementi dati utilizzati nella Tabella del File File X') come un EIF.

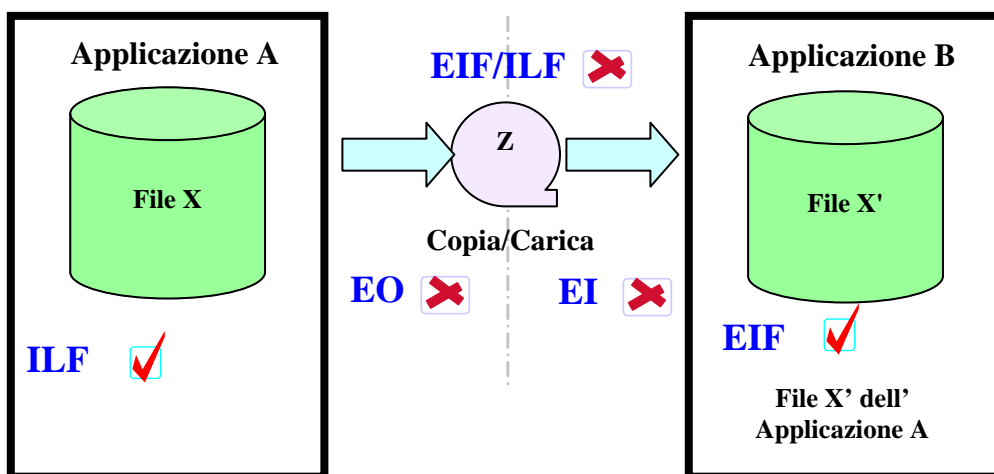
Dal punto di vista logico l'archivio di dati rimane nell'Applicazione A. In questo caso, copiare l'archivio di dati da un'applicazione all'altra è la soluzione di un requisito utente non-funzionale (per es. i dati dell'Applicazione A non sono disponibili quando sono richiesti dall'Applicazione B).

L'*intento primario* è che l'Applicazione B referenzi i dati che dal punto di vista logico esistono nell'Applicazione A.

*Transazioni* – Poichè non vi sono funzioni transazionali logiche nell'inviare o caricare la copia, nessuna transazione è contata per nessuna delle due applicazioni in supporto alla copia e al caricamento dei dati condivisi. Perciò, l'Applicazione A non conta la copia verso l'Applicazione B come un EO/EQ, e l'Applicazione B non conta un EI. Un'ulteriore indicazione sarebbe il fatto che il file nell'Applicazione B è “refreshato” ogni volta con la copia.

*Files* – È coinvolto un unico file logico. L'Applicazione A conta il File X come un ILF. L'Applicazione B conta la propria tabella copiata dal File X come un EIF.

### Diagramma della soluzione



### Riepilogo del conteggio

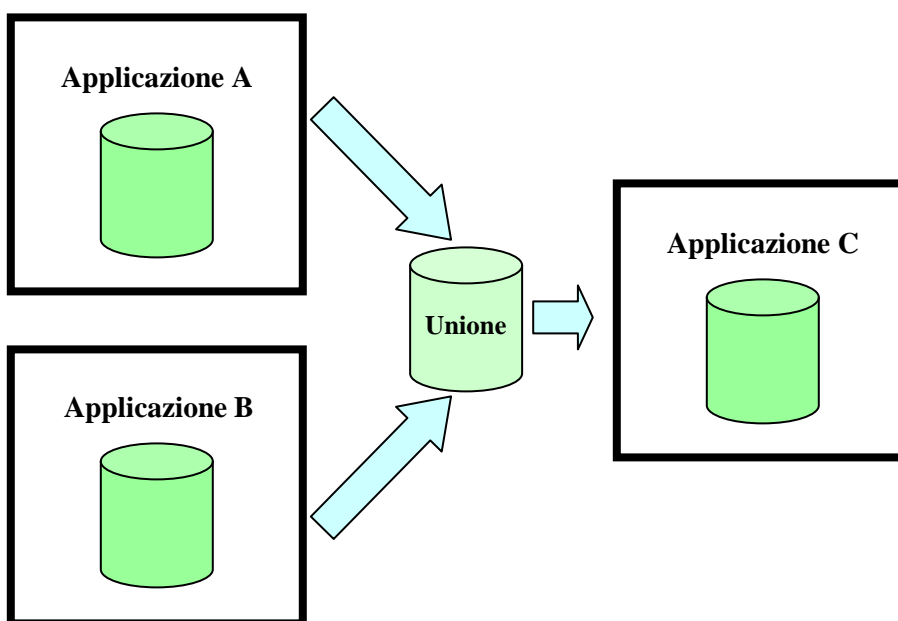
	ILF	EIF	EI	EO/EQ	Note
Applicazione A	☑				
Applicazione B		☑			

## Scenario 5: Copia e unione

**Descrizione** Gli archivi memorizzati in due applicazioni sono copiati come immagini e uniti per formare un file che viene caricato in una terza applicazione. Molteplici file con gli stessi elementi dati vengono consolidati in un singolo file.

**Scenario** Per evitare l'appesantimento dell'Applicazione C, dovuto alla ricerca dinamica dei dati da entrambe le Applicazioni A e B, i dati vengono copiati dall'Applicazione A e dall'Applicazione B e uniti in un nuovo archivio di dati nell'Applicazione C. L'utente richiede che le informazioni provenienti dalle applicazioni A e B siano aggiornate quotidianamente solo per gli scopi di validazione e referenza. Si fa uso di utilità di scaricamento, unione e caricamento. Non è coinvolto alcun trattamento logico applicativo. Questa è tipicamente una soluzione tecnica dove due applicazioni hanno differenti istanze degli stessi dati logici richiesti da una terza applicazione.

### Diagramma

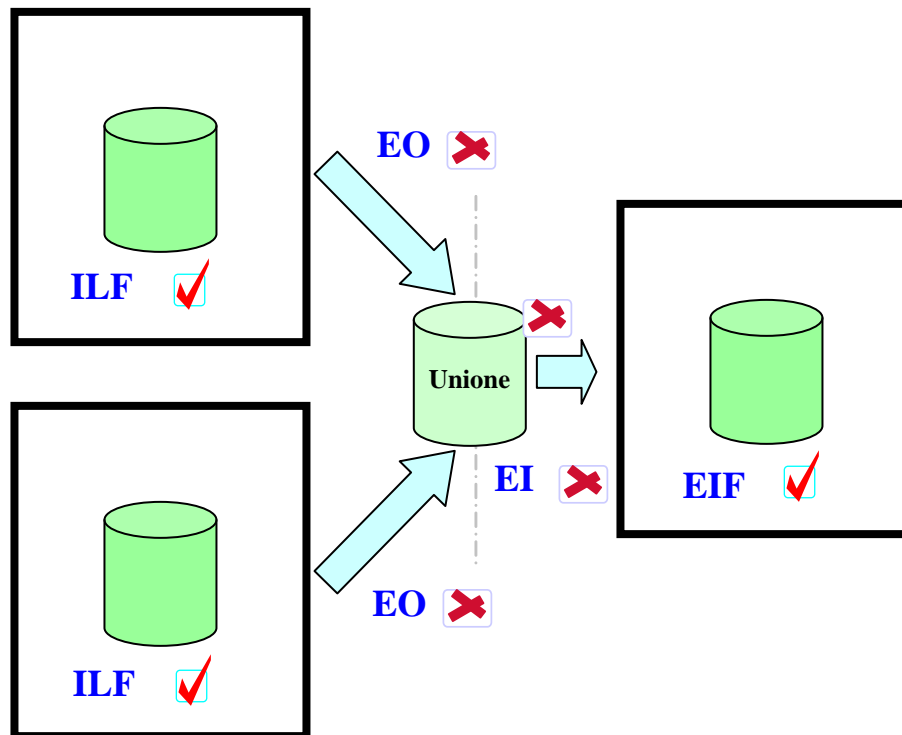


### Interpretazione del conteggio

Dal punto di vista logico, gli archivi di dati rimangono nelle applicazioni A e B. Unire i dati in unico archivio di dati non crea di per sé un nuovo ILF dell'Applicazione C. Inoltre, si deve valutare l'intento primario nell'utilizzo dei dati nell'Applicazione C. Poiché i dati sono usati solo per referenza o validazione, è contato come un EIF. Poiché non c'è alcun trattamento logico ulteriore, non è contata alcuna transazione per nessuna delle applicazioni.

I dati dell'Applicazione C devono essere valutati come indicato alla sezione 3.4 della Parte 1 – Misurare le Funzioni di Tipo Dati e nella Parte 3, Capitolo 2 – File Logici. Anche se i dati provengono da due applicazioni distinte, i tipi di elementi dati sono esattamente gli stessi (si veda la definizione di “unione (merge)”). Perciò, si identifica un unico file logico per l'Applicazione C, come EIF.

### Diagramma della soluzione



### Riepilogo del conteggio

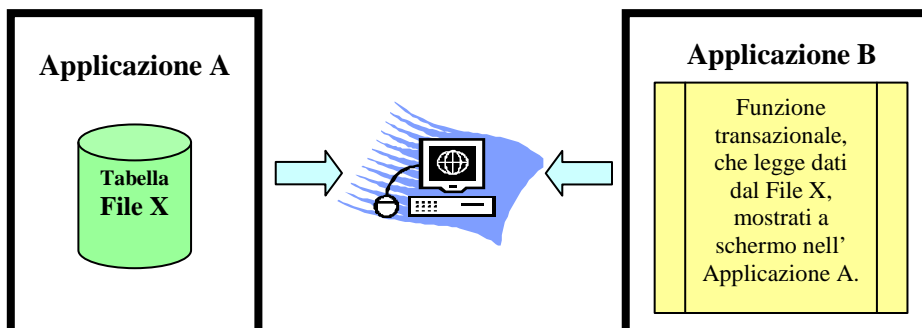
	ILF	EIF	EI	EO/EQ	Note
Applicazione A	<input checked="" type="checkbox"/>				
Applicazione B	<input checked="" type="checkbox"/>				
Applicazione C		<input checked="" type="checkbox"/>			Conta anche come FTR in una funzione transazionale

## Scenario 6: Cattura da schermo

**Descrizione** Accedere alle transazioni a schermo di un'altra applicazione per referenziare/ottenere dati o per aggiornare i dati di quell'applicazione.

**Scenario** L'Applicazione B "legge" il contenuto di una schermata di interrogazione dell'Applicazione A e utilizza questi dati nell'elaborare una funzione transazionale.

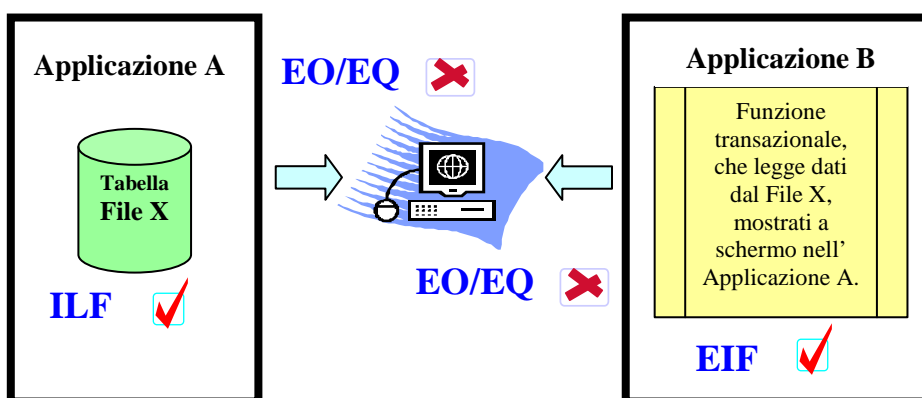
**Diagramma**



**Interpretazione del conteggio**

Dal punto di vista logico, l'Applicazione B sta leggendo i dati dell'Applicazione A. L'Applicazione A ha già contato i dati visualizzati come un EQ/EO (perciò non si conta ulteriormente in questa sede), mentre l'Applicazione B conta i dati come un EIF. Da un punto di vista transazionale, l'Applicazione A è passiva e non conta nulla di più. Per l'Applicazione B la cattura da schermo fa parte del processo elementare della propria transazione ed è contato come un FTR (EIF), poiché i dati in origine sono stati reperiti da un ILF dell'Applicazione A.

**Diagramma della soluzione**



**Riepilogo del conteggio**

	ILF	EIF	EI	EO/EQ	Note
Applicazione A	<input checked="" type="checkbox"/>				
Applicazione B		<input checked="" type="checkbox"/>			

## Scenari di conteggio – Gruppo 2

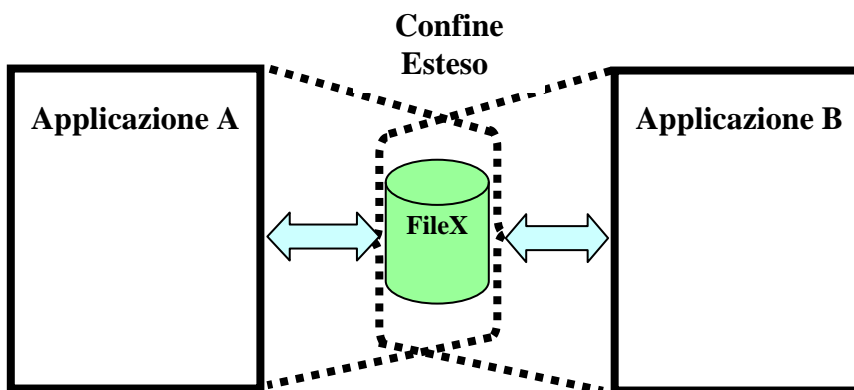
In ognuno degli scenari 7 e 8, l'intento primario è l'Applicazione B mantenga i propri dati a partire dai dati mantenuti dall'Applicazione A; esso può essere implementato come segue:

### Scenario 7: Aggiornamento del medesimo archivio dati

**Descrizione** Il medesimo archivio di dati è mantenuto da due applicazioni distinte.

**Scenario** Entrambe le Applicazioni A e B mantengono lo stesso ILF. Ciascuna applicazione ha la propria vista unica dei dati. Vi sono alcuni elementi dei dati in comune, mentre altri sono unici per ciascuna applicazione.

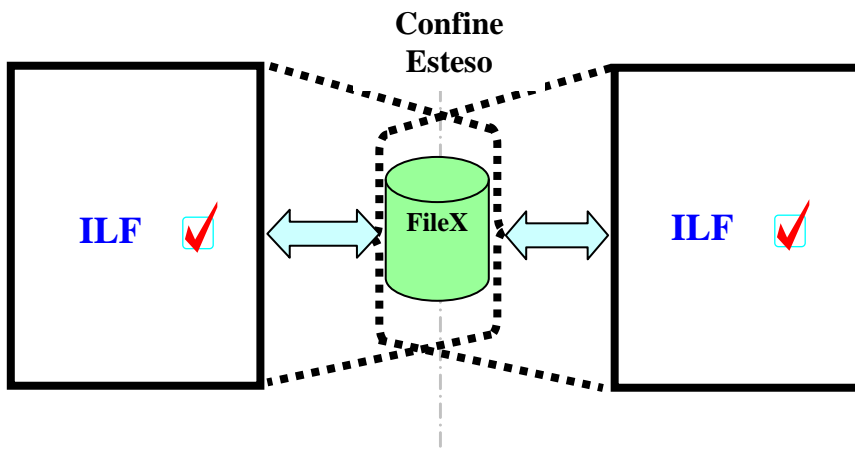
**Diagramma**



**Interpretazione del conteggio**

Si conta un ILF per entrambe le applicazioni, poichè ciascuna possiede delle transazioni per mantenerlo. Entrambe le Applicazioni A e B mantengono dei dati nel medesimo ILF. Ciascuna applicazione conta solo i RET e i DET mantenuti, utilizzati o referenziati da se stessa.

**Diagramma della soluzione**



**Riepilogo del conteggio**

	ILF	EIF	EI	EO/EQ	Note
Applicazione A	☑				
Applicazione B	☑				

## Scenario 8: Dati di transazione standard

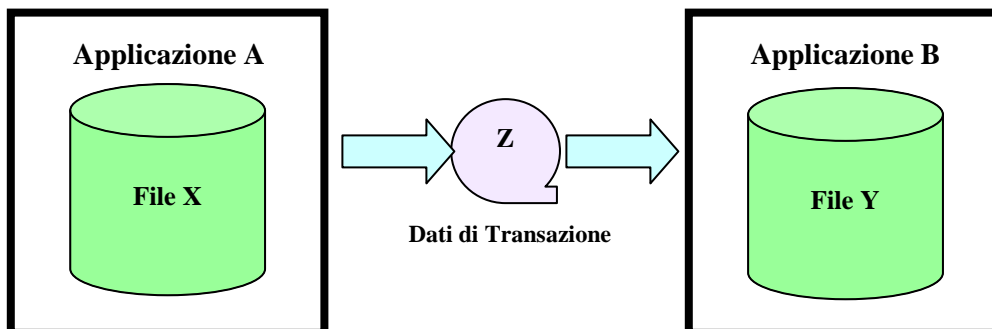
**Descrizione** Dati transazionali sono forniti dall'applicazione sorgente.

**Scenario** L'Applicazione A produce un file transazione delle variazioni, il File Z, che è caricato nell'Applicazione B. I record sono solitamente di più tipi. L'Applicazione B elabora le transazioni in input in base al tipo transazione contenuto nei record del File Z, prima di aggiornare i record del File interno Y. I DET del File X dell'Applicazione A e quelli del File Y dell'Applicazione B sono differenti. Per esempio, il File X è un Catalogo Principale dei Materiali, mentre il File Y è un Elenco dei Prodotti locale. L'elaborazione prevede i seguenti tipi di transazione:

- Inserisci
- Modifica
- Elimina

Questo trasferimento di dati è un requisito utente applicativo. Entrambe le applicazioni A e B hanno un requisito per accedere ad una versione del File X, tuttavia i DET dei due file sono differenti. L'Applicazione A invia i dati relativi soltanto alle variazioni. L'Applicazione B legge i record del File Z e basandosi sul tipo transazione avvia trattamenti logici distinti.

### Diagramma



### Interpretazione del conteggio

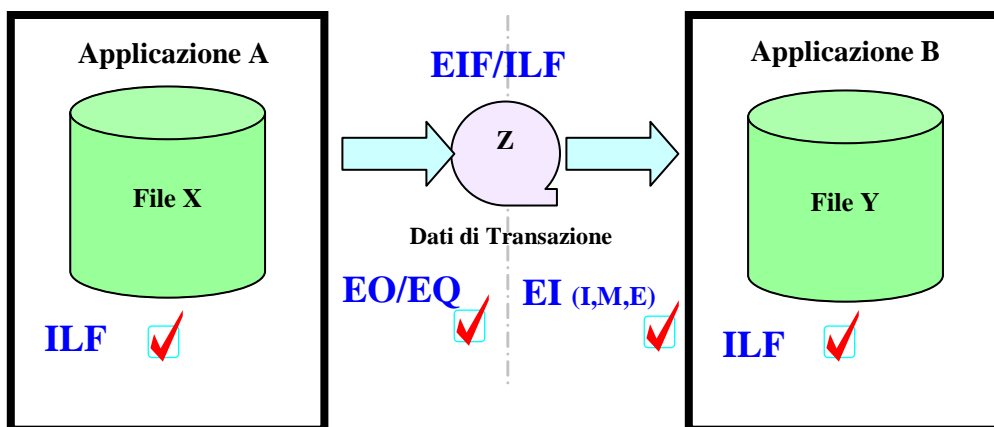
Se ogni record scritto dall'Applicazione A nel File Z è elaborato nello stesso modo, si conta un solo EO/EQ. Soltanto quando sono coinvolti trattamenti logici differenti è possibile avere molteplici funzioni transazionali (per es. EO/EQ) all'interno di un singolo file.

L'Applicazione B conta un EI per ciascuna funzione di manutenzione unica del File Y. Solitamente, il numero di Tipi Transazione del File di transazione Z determina il numero di queste funzioni, ma non necessariamente sempre così. Si devono dimostrare trattamenti logici differenti.

Due file sono coinvolti. L'Applicazione A conta il File X come un ILF. L'Applicazione B conta il File Y come un ILF. Nessuna delle due applicazioni conta il File Z come un file logico.



**Diagramma della soluzione**



**Riepilogo del conteggio**

	ILF	EIF	EI	EO/EQ	Note
Applicazione A	☑			☑	
Applicazione B	☑		☑		

## Riepilogo

In questo capitolo, gli scenari si sono concentrate sull'utilizzo dei dati all'interno dell'applicazione oggetto di misurazione e sull'applicazione delle regole di identificazione degli ILF e degli EIF che fanno riferimento al loro "intento primario".

Questo capitolo non illustra tutte le possibili implementazioni della condivisione dei dati tra le applicazioni. Fornisce, comunque, sufficienti esempi affinché un Analista dei Function Point possa applicare le regole di identificazione degli ILF e degli EIF in modo omogeneo concentrandosi sull'intento primario per l'utilizzo dei dati e sui confini delle applicazioni coinvolte nella misurazione.

# Progetti di manutenzione evolutiva e attività di manutenzione ordinaria

---

**Introduzione** Questo capitolo fornisce una guida addizionale per identificare e misurare le modifiche funzionali apportate a un'applicazione installata. Esso non fa riferimento alla relazione fra la dimensione funzionale della manutenzione evolutiva e l'impegno richiesto per la sua implementazione. Per ulteriori approfondimenti su questo argomento, si rimanda il lettore, tra le altre, alla pubblicazione “*Function Point Analysis for Software Enhancement*” del NESMA (2001, [www.nesma.org](http://www.nesma.org)).

Questo capitolo descrive inoltre le diverse attività di manutenzione e supporto che possono presentarsi durante il ciclo di vita di un'applicazione e per le quali la misurazione della dimensione funzionale fornisce un'utile base per la stima e l'allineamento dei costi.

**Contenuti** Questo capitolo include le seguenti sezioni:

Argomento	Pagina
Misurare i progetti di manutenzione evolutiva	4-2
Considerazioni e suggerimenti	4-10
Elementi per la misurazione dei progetti di manutenzione	4-12
Procedura	4-13
Esempio di un progetto di manutenzione evolutiva	4-14
Considerazioni su manutenzione evolutiva e manutenzione ordinaria	4-20
Sommario	4-24

## Misurare i progetti di manutenzione evolutiva

La dimensione funzionale di un progetto di manutenzione evolutiva misura le modifiche ad un'applicazione pre-esistente già installata, le quali aggiungono, modificano o cancellano funzioni utente. Le modifiche delle funzionalità potrebbero derivare da requisiti utente nuovi o rivisti, da cambiamenti nelle leggi/regolamenti o dai nuovi utenti.

### Ambito e confine di un progetto di manutenzione evolutiva

La dimensione funzionale di un progetto di manutenzione evolutiva include tutte le funzioni che vengono aggiunte, modificate e cancellate. Il confine della/e applicazione/i toccata/e dalla manutenzione evolutiva rimane lo stesso. Il funzionamento dell'applicazione/i rispecchia l'impatto delle funzioni aggiunte, cambiate o cancellate.

Ci potrebbe essere più di un'applicazione inclusa nell'ambito del conteggio. In tal caso, dovrebbero essere identificati molteplici confini applicativi e una dimensione funzionale di un progetto di manutenzione evolutiva separato per ogni applicazione coinvolta.

Qualora venga richiesta la dimensione totale del progetto di manutenzione evolutiva, questa è calcolata sommando le dimensioni funzionali dei progetti di manutenzione evolutiva di tutte le applicazioni incluse nell'ambito del conteggio.

### Misurare le funzioni di tipo dati in progetti di manutenzione evolutiva

L'aggiunta di nuovi file logici interni (ILF) o di file esterni di interfaccia (EIF) in un progetto di manutenzione evolutiva sono generalmente identificati facilmente e sono misurate secondo le regole definite nella Parte 1. Il Capitolo 6 della Parte 2 (Contare le Funzioni di Tipo Dati) ed il Capitolo 2 della Parte 3 (File Logici) contengono ulteriori indicazioni per misurare le Funzioni di Tipo Dati oltre a definizioni e termini collegati.

Tuttavia, dovrebbe essere preso in considerazione quanto segue:

- Se il cambiamento riguarda solo l'aggiunta di nuovi record in un file logico o una nuova valorizzazione di un attributo esistente all'interno di quel file logico, non c'è motivo di contare le funzioni di tipo dati come modificate.
- Se una funzione di tipo dati è modificata perché si aggiunge un attributo e tale attributo non è usato dall'applicazione che si sta misurando, allora non si hanno modifiche per tale applicazione.
- Affinché una funzione di tipo dati sia contata come una funzione modificata, la linea guida generale è tale funzione debba essere modificata strutturalmente (per es. aggiungendo o rimuovendo un attributo o cambiando le caratteristiche dell'attributo).

**Nota:** Un nuovo testo per l'Help è spesso aggiunto alla Funzione di Tipo Dato Help utilizzato da una nuova transazione. Poiché non ci sono cambiamenti strutturali alla Funzione di Tipo Dati Help, questa non deve essere contata come modificata.

- Se si richiede che un'applicazione usi (referenzi o mantenga) un attributo esistente, che non era usato in precedenza, le relative funzioni di tipo dati sono considerate modificate per tale applicazione. Questo può accadere senza alcuna variazione fisica del file
- Se si aggiungono nuovi attributi a un ILF, si devono cercare le funzioni di tipo transazionale, nuove o modificate, che mantengono l'attributo in tale ILF a conferma che è avvenuta una modifica.
- Se si aggiunge un attributo a un ILF, che è mantenuto da due applicazioni, e se un'applicazione mantiene il nuovo attributo mentre l'altra lo referencia solamente, allora entrambe le applicazioni devono contare l'ILF modificato. Tuttavia, la seconda applicazione non avrà alcuna funzione transazionale nuova o modificata che mantiene l'attributo di tale ILF.
- Se un'applicazione non mantiene né referencia un attributo nuovo o modificato, allora per essa non può essere contata alcuna funzione di tipo dati modificata.
- Se un file fisico è aggiunto da un progetto di manutenzione evolutiva, non necessariamente deve risultare in un nuovo file logico. Prima di tutto occorre determinare se il nuovo file fisico rappresenta una modifica di un file logico esistente, con DET aggiuntivi e eventualmente un RET aggiuntivo, o un nuovo file logico. Per ulteriori informazioni sulla misura dei file logici, vedi Parte 2, Capitolo 6, e Parte 3, Capitolo 2.

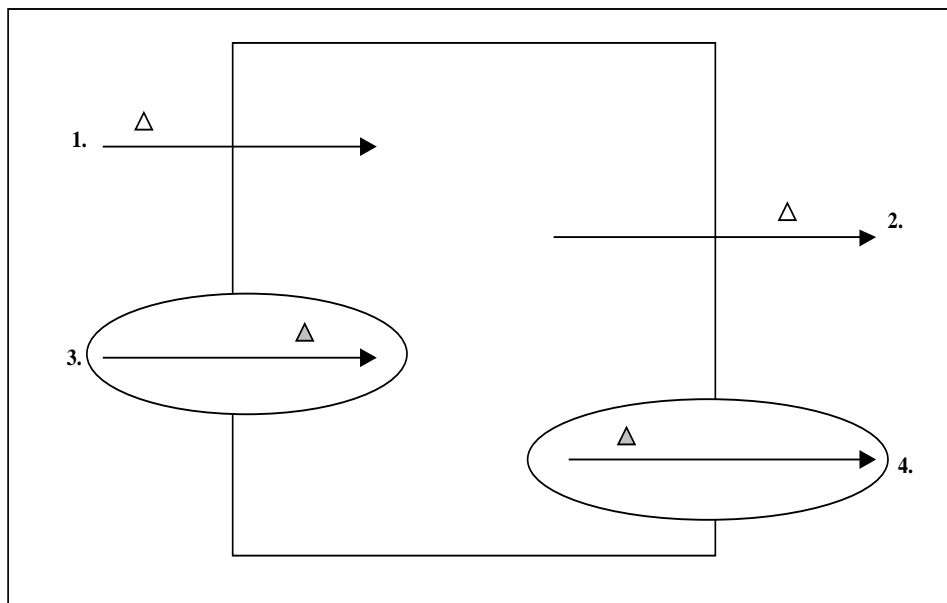
## Misurare le funzioni di tipo transazionale in manutenzione evolutiva

L'aggiunta di nuove funzioni di Tipo Transazionale da parte di un progetto di manutenzione evolutiva sono generalmente facili da identificare e misurare secondo le regole definite nella Parte 1. Il Capitolo 7 della Parte 2 (Misurare le Funzioni di Tipo Transazionale) contiene ulteriori indicazioni per misurare le Funzioni di Tipo Transazionale, oltre a definizioni e termini collegati.

L'identificazione di Funzioni di Tipo Transazionale che sono state modificate dall'aggiunta o dalla rimozione di Tipi di Elementi Dati è ovvia. Non è altrettanto ovvio quando i requisiti utente prevedono cambiamenti nel trattamento logico, come descritto nella Parte 1. Quando un trattamento logico è stato modificato all'interno di un'applicazione per venire incontro ai requisiti di business, il processo elementare che include tale trattamento logico deve essere identificato e contato come cambiato.

Un singolo cambiamento in un trattamento logico non sempre interessa tutte le transazioni collegate.

Per esempio, se viene apportata una modifica di *edit* o di validazione al trattamento logico di input e esistono le transazioni di Inserimento, Cancellazione, Aggiornamento e Interrogazione implicita, solo quelle di Inserimento e Aggiornamento sono contate per la manutenzione evolutiva. Le transazioni di Cancellazione e di Interrogazione implicita non cambiano, a meno che non ci sia uno specifico cambiamento nella logica di cancellazione (per es. edit dell'integrità referenziale) o di interrogazione (per es. selezione o reperimento).



Nel diagramma sopra riportato:

1. La Transazione 1 è modificata perché è stato aggiunto un DET che attraversa il confine.
2. In modo analogo, la Transazione 2 è contata come modificata in quanto un DET è aggiunto ed è inviato attraverso il confine.
3. Nel caso della Transazione 3, una routine interna di validazione è stata modificata in una applicazione. Dato che ciò rappresenta una modifica nel trattamento logico di un processo elementare, la Transazione 3 associata è contata come modificata.
4. In modo simile, la Transazione 4 che prevede una modifica ai precedenti criteri di selezione o di filtro, viene contata come funzionalità modificata.

In alcuni casi, una specifica modifica può impattare su come sono eseguite diverse funzioni di tipo transazionale. Nonostante le modifiche logiche siano implementate fisicamente in una routine usata da diverse transazioni, le funzioni modificate dovrebbero essere identificate in base al processo elementare che racchiude quel trattamento logico. Se sono coinvolti diversi processi elementari, allora si contano diverse funzioni di tipo transazionale. Se è coinvolto solo un singolo processo elementare, si considera come modificata una sola transazione. In ogni caso, i requisiti utente e il punto di vista dell'utente rappresentano il fattore determinante che guida nella scelta.

Ad esempio, il requisito è modificare l'*edit* dei nuovi ordini per includere la validazione dell'ultimo saldo del cliente. Se ci sono varie transazioni di ordini (Individuale, Commerciale, Governativo, etc.) ma viene modificata soltanto la transazione dell'ordine Commerciale, allora dovrebbe essere contata come cambiata solo la transazione dell'ordine Commerciale. Diversamente, se il requisito è modificare tutti i tipi di ordini, allora dovrebbe essere contata come modificata ogni singola transazione.

Un'altra indicazione per contare le funzioni di tipo transazionale è rappresentata dal livello di copertura dei casi di test. Un solo insieme di casi di test dovrebbe indicare che è stato modificato un solo processo elementare.

Spesso la natura del cambiamento è connessa alla modifica di un unico modulo che è usato nella produzione di un grande numero di stampe o estratti, o nell'elaborazione di molte transazioni di input. Tutte le funzioni possono utilizzare la routine comune, ma soltanto un sottoinsieme di quelle funzioni usa l'edit che è stato modificato in quella routine. Dovrebbero essere contate come cambiate solo le funzioni che incorporano il trattamento logico cambiato. La sfida è pertanto quella di accertare correttamente l'appropriato livello di *modifica funzionale*. Va posto l'accento sui requisiti di business, in modo tale che la dimensione funzionale del progetto di manutenzione evolutiva rifletta l'intento della richiesta dell'utente.

## Trattamento logico

Il *trattamento logico* è definito come ognuno dei requisiti specificatamente richiesti dall'utente per completare un processo elementare come validazioni, algoritmi o calcoli, lettura o manutenzione di funzioni di Tipo Dato. Tali requisiti possono includere le seguenti azioni:

1. Eseguire validazioni.

Per esempio, quando si inserisce un impiegato in una nuova organizzazione, il processo valida il DET della tipologia dell'impiegato.

- Se esiste un requisito per elaborare una validazione diversa o per modificare la validazione in una funzione di tipo transazionale esistente, si conterebbe la transazione interessata come modificata nella dimensione funzionale del progetto di manutenzione evolutiva.

2. Eseguire formule matematiche e calcoli.

Per esempio, il processo che presenta una lista di tutti gli impiegati di un'organizzazione include il calcolo del numero totale degli impiegati salariati, quello degli impiegati a ore e di tutti gli impiegati.

- Se esiste un requisito di business per modificare un calcolo esistente (per es. prima la formula era  $A + B = C$  e ora  $C = A \times B$ ), si conterebbe la transazione che include quel calcolo come modificata nella dimensione funzionale del progetto di manutenzione evolutiva.
- Nella situazione attuale esiste una lista di impiegati contata come un EQ. I requisiti del progetto di manutenzione evolutiva stabiliscono di visualizzare il numero totale degli impiegati. Si identificherebbe la transazione interessata come modificata e il suo tipo passerebbe da EQ a EO, nella dimensione funzionale del progetto di manutenzione evolutiva.

3. Convertire valori equivalenti.

Per esempio, l'età di un impiegato è convertita in un raggruppamento di fasce di età utilizzando una tabella.

- Se esiste un requisito di business di modifica della funzionalità per includere la capacità di convertire lo stipendio di un impiegato in un raggruppamento di fasce di stipendi, la transazione deve essere contata come cambiata nella dimensione funzionale del progetto di manutenzione evolutiva.

4. Filtrare e selezionare dati utilizzando criteri specifici per confrontare insiemi multipli di dati.

Per esempio, per generare una lista di impiegati in base al loro incarico, un processo elementare utilizza il codice di un incarico di lavoro per selezionare e ottenere una lista di tutti gli impiegati con quel particolare incarico.

- Se esiste un requisito che modifica i criteri di selezione o ne aggiunge di ulteriori, escludendo le modifiche o l'aggiunta di valori, per una transazione esistente (un elenco di impiegati deve ora mostrare quelli assegnati al loro incarico da meno di un anno), la transazione deve essere contata come cambiata nella dimensione funzionale del progetto di manutenzione evolutiva.
- Se esiste un requisito che modifica solo il/i valore/i di un criterio di selezione esistente, come selezionare un diverso dipartimento o aggiungendo un ulteriore dipartimento ad un elenco di dipartimenti, allora non si conta nessun cambiamento per la transazione.
- Se il requisito è di modificare il criterio di selezione dalla selezione di un singolo dipartimento ad un elenco di dipartimenti, la transazione deve essere considerata come modificata.
- Se esiste un requisito per modificare la schermata di ricerca di un impiegato aggiungendo un ulteriore filtro sulla località, questo filtro non crea un nuovo processo elementare. È contato come una modifica alla funzione nella dimensione funzionale del progetto di manutenzione evolutiva.

5. Analizzare condizioni per determinare quali siano applicabili.

Per esempio, il trattamento logico utilizzato dal processo elementare che aggiunge un nuovo impiegato sarà influenzato dalla condizione che esso sia salariato o pagato a ore. L'ingresso di DET (e il risultante trattamento logico) basati su una scelta differente (salariato o pagato a ore) in questo esempio è parte di un solo processo elementare.

- Se esiste un requisito per modificare la condizione o aggiungere condizioni addizionali ad una transazione esistente, si conterebbe la transazione interessata come modificata nella dimensione funzionale del progetto di manutenzione evolutiva.



6. Aggiornare uno o più ILF.

Per esempio, il processo elementare che aggiunge un nuovo impiegato aggiorna l'ILF Impiegato per mantenere i relativi dati.

- Se un requisito di business richiede l'aggiornamento di un ulteriore ILF o di DET differenti da parte di una transazione esistente, si conterebbe la transazione interessata come modificata nella dimensione funzionale del progetto di manutenzione evolutiva.

7. Referenziare uno o più ILF o EIF.

Per esempio, quando si aggiunge un nuovo impiegato, l'EIF Valuta viene referenziata per usare il corretto tasso di conversione relativo a dollari US al fine di determinare la paga oraria dell'impiegato.

- Se esiste un requisito di business nel referenziare nuovi ILF, EIF o DET in una transazione esistente, si conterebbe la transazione interessata come modificata nella dimensione funzionale del progetto di manutenzione evolutiva.

8. Reperire dati o informazioni di controllo.

Per esempio, per visualizzare un elenco di impiegati, l'informazione dell'impiegato è reperita da una funzione di Tipo Dati.

- Se esiste un requisito di business nel reperire nuove informazioni in una transazione esistente, si conterebbe la transazione interessata come modificata nella dimensione funzionale relativa del progetto di manutenzione evolutiva.

9. Creare dati derivati trasformando dati esistenti per ottenere ulteriori dati.

Per esempio, per determinare (derivare) un numero di registrazione di un paziente (per es. ROSCA01), si concatenano i seguenti dati:

- le prime tre lettere del cognome (per es. ROS per Rossi)
- le prime due lettere del nome (per es. CA per Carlo)
- un progressivo univoco a due cifre (a partire da 01)
- Se esiste un requisito di business che modifica il modo in cui la transazione concatena i dati, si conterebbe la transazione interessata come modificata nella dimensione funzionale del progetto di manutenzione evolutiva.

10. Alterare il comportamento dell'applicazione.

Per esempio, il comportamento del processo elementare relativo al pagamento degli impiegati risulta alterato quando si apporta una modifica alla data di pagamento, da “ogni quindicesimo e ogni ultimo giorno del mese” a “ogni venerdì”; risultando 26 periodi di pagamento per anno rispetto ai 24.

- Se esiste un requisito di business nell'alterare il comportamento del sistema (per es. la transazione viene cambiata in modo tale che la data di pagamento influenzi solo gli impiegati a ore e non tutti gli impiegati), si conterebbe la transazione interessata come modificata nella dimensione funzionale del progetto di manutenzione evolutiva.

11. Preparare e presentare informazioni all'esterno del confine.

Per esempio, un elenco di impiegati è formattato e mostrato all'utente.

- Quando esiste un requisito di business nel presentare nuovi DET al di fuori del confine, si conterebbe la transazione interessata come modificata nella dimensione funzionale del progetto di manutenzione evolutiva.
- Modifiche apportate a intestazioni, formati, colori o altri elementi di presentazione non sono considerati modifiche del trattamento logico e di conseguenza non fanno parte nella dimensione funzionale del progetto di manutenzione evolutiva.
- Quando il requisito di business è di inviare un file di output già esistente verso un'applicazione diversa, o verso altre applicazioni, senza che questo comporti nessun altro cambiamento al trattamento logico (per es. criteri di selezione, calcoli), allora non c'è modifica della dimensione funzionale del progetto di manutenzione evolutiva.
- Il riordinamento di dati sullo schermo, su un report o su file per mostrare dei dati già esistenti in una nuova posizione non è considerato una modifica al trattamento logico e non è contata nell'ambito del progetto di manutenzione evolutiva.
- La modifica delle caratteristiche (per es. lunghezza, tipo, precisione, ecc.) di un attributo che attraversa il confine per essere contata deve comportare la modifica di un'altra forma di trattamento logico (per es. validazione, calcoli).
- Quando per una funzione una nuova schermata, o una esistente, richiede aggiunte o modifiche all'help, il cambiamento alla funzione Help esistente non è contato perché c'è solo l'aggiunta o la modifica del testo.

12. Accettare dati o informazioni di controllo che entrano nel confine dell'applicazione.

Per esempio, un utente immette informazioni per inserire l'ordinativo di un cliente in un'applicazione.

- Quando il requisito di business ha come risultato DET diversi che entrano nel confine, la transazione interessata si conterebbe come modificata nella dimensione funzionale del progetto di manutenzione evolutiva.

- Quando il requisito di business è di accettare un file di input già esistente da un'applicazione diversa, o da un'ulteriore applicazione, senza che questo comporti nessun altro cambiamento al trattamento logico (per es. validazione, calcoli), non c'è nessun impatto sulla dimensione funzionale del progetto di manutenzione evolutiva.
- La modifica delle caratteristiche (per es. lunghezza, tipo, precisione) di un attributo che attraversa il confine deve comportare dei cambiamenti a un'altra forma di trattamento logico (per es. validazione, calcoli) per essere contata.

### 13. Ordinare e organizzare un insieme di dati.

Questa forma di trattamento logico non ha impatto sull'identificazione del tipo o il contributo all'univocità di un processo elementare; cioè l'orientamento di un dato non costituisce univocità.

Per esempio, un elenco di impiegati è ordinato in ordine alfabetico o per località.

Per esempio, sulla schermata di un ordinativo, l'intestazione dell'ordinativo è posizionata in alto sullo schermo, e il dettaglio dell'ordinativo è posizionato al di sotto.

**Nota:** I cambiamenti nella sequenza di ordinamento sono tipicamente contati. I soli cambiamenti nella disposizione non sono in genere contati.

- Quando il requisito di business comporta la modifica della sequenza di ordinamento esistente (per es. l'utente richiede la lista degli impiegati ordinata secondo la sede piuttosto che in ordine alfabetico), si conterebbe la transazione interessata come modificata nella dimensione funzionale del progetto di manutenzione evolutiva.
- Sulla schermata di immissione dei dati di un ordinativo, l'utente richiede che le informazioni sull'intestazione dell'ordine siano posizionati sulla sinistra dei dettagli dell'ordine invece che di sopra. Non c'è nessun conteggio per questa modifica.
- L'utente richiede che nella schermata 'Assegnazioni Impiegati' il cognome sia posizionato a sinistra delle iniziali e del nome. Nessuna modifica è contata per il riposizionamento. Se la richiesta è anche di pre-popolare i dati basandosi sul cognome quando viene inserito, allora c'è la modifica di un'altra forma di trattamento logico e la transazione 'Assegnazione Impiegati' è contata.
- L'utente richiede un report aggiuntivo degli stessi dati (lista di impiegati), ordinati secondo l'allocazione. Non si conterebbe alcuna transazione nuova, ma si includerebbe la modifica della transazione esistente nella dimensione funzionale del progetto di manutenzione evolutiva.

Un processo elementare può includere varie alternative o occorrenze delle suddette azioni. Per esempio: validazioni, filtri, riordinamenti, ecc.

## Considerazioni e suggerimenti

La maggior parte dei progetti di manutenzione evolutiva richiedono cambiamenti solo nei trattamenti logici e non negli input, output o nei file (per esempio aggiunta o cancellazione di attributi). Si riportano di seguito elementi o domande da discutere con gli sviluppatori durante la misurazione delle manutenzioni evolutive per applicazioni già in esercizio.

### Domande da considerare per un progetto di manutenzione evolutiva

Le domande proposte nel seguito possono essere usate durante l'intervista con gli sviluppatori o con l'esperto del dominio (SME – Subject Matter Expert) durante la fase di misurazione funzionale. Risposte affermative a tali domande indicano possibili modifiche alle funzionalità utente. Si richiede pertanto un'ulteriore analisi per determinare se e queste come potrebbero avere un impatto sulla dimensione funzionale.

#### **Quali NUOVE funzionalità sono state create per gli utenti dell'applicazione?**

- A questo progetto sono stati aggiunti nuovi file permanenti, database, tabelle, entità o oggetti?
- L'applicazione riceve e processa nuovi input o file transazione che non erano presenti prima in produzione?
- Sono state costruite nuove schermate per gli utenti?
- Vi sono nuove interazioni (interfacce) con altre applicazioni?
- L'applicazione genera nuovi output, nuovi report o nuovi file per altri sistemi?
- È stato aggiunto un nuovo tipo record ad un file transazione o procedura di popolamento?
- Sono stati inseriti nuovi processi di interrogazione per l'utente?
- Sono stati aggiunti nuovi processi batch?

#### **Quali funzionalità utente sono state CAMBIATE/MODIFICATE a seguito dei nuovi requisiti utente?**

- Sono stati aggiunti o cancellati attributi o colonne da file permanenti, database o tabelle? Sono state modificate le caratteristiche di attributi o colonne già esistenti? L'aggiunta di nuovi valori ad un attributo esistente o nuove righe ad una tabella non influenza il conteggio
- Sono stati modificati trattamenti logici relativi a schermate di input o a input ricevuti tramite file inviati da parte degli utenti o di altre applicazioni?
- Vi sono stati cambiamenti alle attuali schermate di input o a file transazione da parte di altre applicazioni (per esempio, nuovi attributi o modifica delle caratteristiche di attributi già esistenti)?

- Sono stati modificati report utente, file di output o schermate esistenti aggiungendo o rimuovendo degli attributi?
- Per alcuni output esistenti ci sono stati cambiamenti ai trattamenti logici per produrre questi file di output, report o schermate?
- Sono state modificate delle schermate a seguito di nuove interrogazioni?
- Vi sono stati cambiamenti nella selezione o nella digitazione di criteri o filtri nelle schermate delle interrogazioni?
- Sono stati modificati dei processi batch?

**Quali funzionalità utente sono state CANCELLATE/RIMOSSE a seguito dei nuovi requisiti utente?**

- Quale funzionalità utente è stata cancellata o rimossa dalla produzione?
- Esistono file permanenti che non vengono più richiesti?
- L'applicazione ha smesso di ricevere un file di input da un altro sistema o ha rimosso una schermata precedentemente accessibile ad un utente?
- Sono stati rimossi dei report perchè l'utente non li richiede più?
- Sono stati eliminati file di output forniti ad un altro sistema?
- Sono state rimosse interrogazioni per l'utente?
- Sono stati rimossi dei processi batch?

**Quali funzionalità di CONVERSIONE sono state fornite?**

- È avvenuta qualche elaborazione di dati *una tantum* per ripulire, organizzare o popolare nuovi attributi come risultato di modifiche strutturali di un file permanente?
- C'è stata una richiesta di popolamento dati per nuovi file permanenti?
- Vi sono dei report di conversione su richiesta dell'utente?

**Quali ulteriori modifiche sono stati apportate all'applicazione a seguito del presente progetto di manutenzione evolutiva?**

- Questa domanda conclusiva può sollecitare l'esperto di dominio o lo sviluppatore a fornire ulteriori indizi che possano avere un impatto sul conteggio.

## Sommario delle considerazioni e dei suggerimenti

Queste linee guida possono aiutare nel determinare le funzionalità rilasciate dal progetto e il loro impatto sulla baseline dell'applicazione che si sta misurando. La combinazione delle nuove funzionalità aggiunte, gli effetti delle modifiche apportate alle funzionalità esistenti e le funzionalità utente cancellate saranno usate per dimensionare il progetto di manutenzione evolutiva così come per calcolare il conteggio aggiornato della baseline dell'applicazione.

## Elementi per la misurazione dei progetti di manutenzione

In aggiunta a quella identificata nella Parte 2, Capitolo 3 (Raccogliere la Documentazione Disponibile), la seguente documentazione dovrebbe anche essere fornita per un progetto di manutenzione evolutiva:

- Documentazione esistente della misura della Dimensione Funzionale
- Requisiti utente per ciascun cambiamento/modifica apportato all'applicazione esistente.
- Layout delle schermate esistenti, layout dei report e/o diagrammi di flusso delle transazioni batch per tutte le funzioni interessate, come erano PRIMA del progetto di manutenzione evolutiva.
- Layout delle schermate modificate, layout dei report e/o diagrammi di flusso delle transazioni batch per tutte le funzioni interessate, come saranno DOPO il progetto di manutenzione evolutiva.
- Documentazione di tutte le nuove funzioni (nuovi report, output, funzioni di data entry o entità) che devono essere aggiunte all'applicazione come parte del progetto di manutenzione evolutiva.
- Layout delle schermate, layout dei report e/o diagrammi di flusso delle transazioni batch per le funzionalità che devono essere RIMOSSE dall'applicazione come risultato del progetto di manutenzione evolutiva

È vero che, in molti casi, non tutta la documentazione citata può essere disponibile o applicabile. La fonte di informazioni più importante è rappresentata da un esperto che conosca a fondo l'applicazione e dalla documentazione dei requisiti.

Se non si dispone della baseline, è necessario stabilire il confine e identificare i dati logici e le funzioni di tipo transazionale.

## Procedura

Di seguito sono elencati i passi suggeriti per misurare la dimensione funzionale di un progetto di manutenzione evolutiva; comunque, l'esecuzione può essere svolta in qualunque ordine

Passo	Azione
1	Raccogliere e riesaminare la documentazione disponibile.
2	Incontrarsi con l'esperto del dominio per discutere i cambiamenti pianificati/fatti.
3	Identificare e valutare le funzionalità aggiunte (ADD).
4	Identificare e valutare le funzionalità modificate (CHG): <ul style="list-style-type: none"><li>• Determinare la complessità delle funzionalità <i>prima</i> del cambiamento (dalla documentazione della misura della dimensione funzionale precedente o misurarla come era prima del cambiamento) [CHGB – Change Before].</li><li>• Determinare la complessità delle funzionalità <i>dopo</i> i cambiamenti [CHGA – Change After].</li></ul>
5	Identificare e valutare le funzionalità cancellate (DEL).
6	Identificare e valutare le conversioni (CFP, Conversion FP) o le funzionalità richieste <i>una tantum</i> per implementare il progetto di manutenzione evolutiva.

## Esempio di un progetto di manutenzione evolutiva

### Esempio

Questa sezione mostra un esempio relativo ad un semplice progetto di manutenzione evolutiva. Si assuma che i requisiti per il progetto di manutenzione evolutiva includono le seguenti modifiche:

- L'utente ha bisogno di ricevere un report aggiuntivo relativo ai lavori, contenente dei totali.
- Si richiedono DET addizionali per aggiungere i lavori in modalità batch e correggere le transazioni sospese. Per la transazione 'Inserisci Lavoro' si aggiunge anche un riferimento alla sicurezza.
- La funzione 'Assegna Lavoro' deve ora verificare che il tipo di lavoro assegnato a un impiegato sia conforme alla categoria dell'impiegato, mantenuta dal Sistema Relazioni Impiegati.
- Il report 'Impiegati' per la funzione 'Durata Assegnazione' deve essere modificato. Invece di mostrare un conteggio degli impiegati tra i 12 e i 24 mesi, il report dovrà ora mostrare solo gli impiegati con salario.
- L'utente non ha più bisogno di aggiungere un lavoro in modalità interattiva; tale funzionalità quindi deve essere rimossa

### Funzionalità applicative

Il seguente paragrafo illustra le funzionalità dell'applicazione misurate per il progetto di manutenzione evolutiva dell'esempio. Le funzionalità sono descritte come aggiunte, modificate o cancellate.

### Funzionalità aggiunte (ADD)

La seguente tabella mostra la complessità funzionale delle funzionalità aggiunte, misurata quando il progetto è completato.

**Nota 1:** Costruire un nuovo report era un Output Esterno aggiunto.

**Nota 2:** I dati sulle Relazioni Impiegati, referenziati dall'applicazione Risorse Umane (RU), sono identificati come un EIF.

Funzioni di Tipo Dati	RET	DET	Complessità Funzionale
<b>File Esterno di Interfaccia (EIF)</b>			
Dati di Relazione Impiegati	1	2	Bassa

Funzioni di Tipo Transazionale	FTR	DET	Complessità Funzionale
<b>Output Esterno</b>			
Stampa Lavoro	1	15	Bassa



**Funzionalità  
modificate  
(CHGA)**

La seguente tabella mostra la complessità funzionale per le funzionalità modificate, come le funzioni diventeranno *dopo* che il progetto di manutenzione evolutiva sarà completato.

**Nota 1:** La complessità per aggiungere un lavoro era aumentata a causa del tipo di file referenziato (FTR) aggiunto. La complessità per la Correzione delle Transazioni Sospese rimane bassa.

**Nota 2:** Sebbene si conti un ulteriore FTR, la complessità di ‘Aggiungi Assegnazione Lavoro’ non cambia poiché era già ad alta complessità.

**Nota 3:** Sebbene ci sia una modifica nel criterio di selezione e nella logica di totalizzazione, la complessità del report non cambia.

Funzioni di Tipo Transazionale	FTR	DET	Complessità Funzionale
<b>Input Esterni (EI)</b>			
Aggiungi informazioni lavoro (input batch)	3	8	Alta
Correggi transazioni sospese	1	8	Bassa
Aggiungi Assegnazione Lavoro	4	7	Alta
<b>Output Esterni (EO)</b>			
Impiegati per Durata assegnazione	3	7	Media

**Funzionalità  
cancellate  
(DEL)**

La seguente tabella mostra la complessità funzionale per le funzionalità cancellate identificate alla fine del progetto.

Funzioni di Tipo Transazionale	FTR	DET	Complessità Funzionale
<b>Input Esterno (EI)</b>			
Aggiungi informazioni lavoro (schermata di input)	1	7	Bassa

## Contributo dell'applicazione alla dimensione funzionale

Il seguente paragrafo spiega il contributo delle funzionalità applicative al totale della dimensione funzionale.

### Funzionalità aggiunte (ADD)

La tabella seguente illustra il contributo alla dimensione funzionale delle funzionalità aggiunte identificate alla fine del progetto.

Tipo Funzione		Complessità Funzionale		Totali Complessità	Totali Tipi Funzione
EIF	1	Bassa	X 5 =	5	
		Media	X 7 =		
		Alta	X 10 =		
					5
EO	1	Bassa	X 4 =	4	
		Media	X 5 =		
		Alta	X 7 =		
					4

### Funzionalità modificate (CHGA)

La tabella seguente illustra il contributo alla dimensione funzionale delle funzionalità modificate per come saranno *dopo* che il progetto di manutenzione evolutiva sarà completato.

Tipo Funzione		Complessità Funzionale		Totali Complessità	Totali Tipi Funzione
EI	1	Bassa	X 3 =	3	
		Media	X 4 =		
	2	Alta	X 6 =	12	
					15
EO		Bassa	X 4 =		
	1	Media	X 5 =	5	
		Alta	X 7 =		
					5

## Funzionalità cancellate (DEL)

La tabella seguente illustra il contributo alla dimensione funzionale delle funzionalità cancellate

Tipo Funzione	Complessità Funzionale			Totali Complessità	Totali Tipi Funzione
EI	1	Bassa	X 3 =	3	
		Media	X 4 =		
		Alta	X 6 =		
					3

## Calcolo finale

Di seguito è mostrata la dimensione funzionale del progetto di manutenzione evolutiva, utilizzando la complessità ed il contributo per questo esempio.

$$EFP = ADD + CHGA + CFP + DEL$$

dove

- EFP è il numero di function point del progetto di manutenzione evolutiva (Enhancement Function Point)
- ADD è la dimensione delle funzioni che sono state aggiunte dal progetto di manutenzione evolutiva (ADDED Function Point)
- CHGA è la dimensione delle funzioni che sono state cambiate dal progetto di manutenzione evolutiva – come sono/saranno dopo l'implementazione (CHANGED Function Point After)
- CFP è la dimensione delle funzionalità di conversione (Conversion Function Point)
- DEL è la dimensione delle funzioni che sono state cancellate dal progetto di manutenzione evolutiva (DELETED Function Point)

$$EFP = 9 + 20 + 0 + 3$$

$$EFP = 32$$

## Dimensione funzionale iniziale dell'applicazione

La dimensione funzionale iniziale dell'applicazione è mostrata di seguito.

$$AFP = ADD$$

dove

- AFP è il numero di function point dell'applicazione (Application Function Point)
- ADD è la dimensione delle funzioni che saranno rilasciate all'utente con il progetto di sviluppo (escludere la dimensione di ogni funzionalità di conversione) o le funzionalità che esistono quando l'applicazione è misurata (ADDED Function Point)

$$AFP = 115$$

**Nota:** Solo la dimensione delle funzionalità applicative installate per l'utente vengono incluse nella dimensione funzionale iniziale dell'applicazione.

## Dimensione funzionale dell'applicazione dopo un progetto di manutenzione evolutiva

La misura della dimensione funzionale dell'applicazione che riflette un progetto di manutenzione evolutiva è mostrata di seguito.

$$AFPA = (AFPB + ADD + CHGA) - (CHGB + DEL)$$

dove

- AFPA è il numero di function point dell'applicazione dopo il progetto di manutenzione evolutiva (Application Function Point After)
- AFPB è il numero di function point dell'applicazione prima del progetto di manutenzione evolutiva (Application Function Point Before)
- ADD è la dimensione delle funzioni che sono state aggiunte dal progetto di manutenzione evolutiva (ADDED Function Point)
- CHGA è la dimensione delle funzioni che sono state cambiate dal progetto di manutenzione evolutiva – come sono/saranno dopo l'implementazione (CHANGED Function Point After)
- CHGB è la dimensione delle funzioni che sono state cambiate dal progetto di manutenzione evolutiva – come sono/erano prima dell'inizio del progetto (CHANGED Function Point Before)
- DEL è la dimensione delle funzioni che sono state cancellate dal progetto di manutenzione evolutiva (DELETED Function Point)

$$AFPA = (115 + 9 + 20) - (18 + 3)$$

$$AFPA = 123$$

Alcune persone possono voler applicare un fattore di aggiustamento del valore (VAF, Value Adjustment Factor), che considera 14 caratteristiche generali del sistema (GSC, General System Characteristics). Per indicazioni sull'uso del VAF e delle GSC, si faccia riferimento all'Appendice C.

## Considerazioni su manutenzione evolutiva e manutenzione ordinaria

Una volta che un'applicazione è stata sviluppata e installata, deve essere poi mantenuta (modificata) affinché possa continuare a soddisfare le necessità di un contesto di business e tecnico in continua trasformazione. Questa manutenzione comprende un ampio spettro di attività che vanno eseguite durante questa fase del ciclo di vita dell'applicazione, alcune delle quali includono cambiamenti funzionali che sono applicabili all'analisi dei function point (FPA – Function Point Analysis).

### Categorie di manutenzione

L'IEEE (Institute of Electrical and Electronics Engineers) definisce tre categorie di manutenzione:

**Manutenzione Adeguativa:** manutenzione del software svolta per rendere un programma utilizzabile in un ambiente mutato.

**Manutenzione Correttiva:** manutenzione del software svolta per correggere i difetti del software o dell'hardware.

**Manutenzione Perfettiva:** manutenzione del software svolta per migliorare le prestazioni, la manutenibilità oppure altre caratteristiche di un programma.

Mentre la parte principale di questo capitolo ha fornito dei suggerimenti sul calcolo dei function point e delle linee guida per il miglioramento di applicazioni esistenti, non esiste uno standard generalmente riconosciuto dalle industrie per una classificazione coerente delle attività che rientrano nelle categorie citate. Questa sezione fornisce un quadro di riferimento, basato su esperienze comuni nell'industria, in cui valutare l'applicabilità della FPA nell'ambito del supporto alle applicazioni installate.

L'ISO (Organizzazione Internazionale per la Standardizzazione) e la IEC (Commissione Elettronica Internazionale) definiscono tre categorie di manutenzione:

#### Manutenzione adeguativa

Le modifiche di un prodotto software, svolte dopo la consegna, per mantenerlo usabile in un ambiente modificato o che sta cambiando. La manutenzione adeguativa fornisce i miglioramenti necessari per accogliere le modifiche all'ambiente nel quale un prodotto software deve operare. Questi cambiamenti sono quelli che devono essere fatti per mantenere l'ordine con i cambiamenti ambientali. Per esempio, il sistema operativo deve essere aggiornato e devono essere fatti alcuni cambiamenti per accogliere il nuovo sistema operativo. (ISO/IEC 14764:2006)

#### Manutenzione correttiva

La modifica reattiva di un prodotto software consegnato per correggere dei problemi scoperti. La modifica corregge il prodotto software per soddisfare dei requisiti. (ISO/IEC 14764:2006)

## **Manutenzione perfettiva**

La modifica di un prodotto software consegnato per scoprire e correggere dei vizi occulti nel prodotto software prima che essi si manifestino come funzionamenti difettosi. La manutenzione migliorativa fornisce una evoluzione per l'utente, miglioramenti della documentazione del programma, ricodifica per migliorare le prestazioni del software, manutenibilità o per altri attributi del software. Da confrontare con: manutenzione adeguativa; manutenzione correttiva. (ISO/IEC 14764:2006)

La misurazione della dimensione funzionale quantifica la dimensione dei requisiti di business. Nel contesto della manutenzione evolutiva, essa misura gli effetti delle modifiche su tali requisiti. Perciò, la misurazione della dimensione funzionale è applicabile ad un sottoinsieme della manutenzione adeguativa. Ciò include le funzionalità software aggiunte, modificate o cancellate, così come le funzionalità software fornite per convertire i dati e soddisfare altri requisiti di realizzazione (per esempio, prospetti sulla conversione).

Non è agevole fornire una lista completa ed esaustiva di tutte le attività di sviluppo e supporto. Piuttosto, è possibile individuare le seguenti aree che riguardano l'applicabilità della FPA. Si raccomanda vivamente che ogni compagnia/azienda sviluppi una propria guida relativa alle attività, alle definizioni e alla terminologia specifica per quella data organizzazione.

## **Manutenzione ordinaria dell'applicazione e attività di supporto**

Poiché le attività di manutenzione ordinaria e supporto sono soggette a una reportistica non sempre consistente, è opportuno sviluppare su base locale linee che affrontino tali aspetti. Quelle che seguono sono alcune delle attività più comunemente incontrate, con suggerimenti relativi alla FPA.

### **Richieste di manutenzio- ne ordinaria**

A prescindere dal tempo o dal livello di impegno lavorativo richiesto, è il tipo di attività che determina come classificare il lavoro. La FPA non dovrebbe essere usata per dimensionare lavori di manutenzione correttiva e perfettiva. La manutenzione correttiva dovrebbe essere attribuita al progetto di manutenzione evolutiva o di sviluppo che ha introdotto i difetti. La manutenzione perfettiva non dovrebbe essere attribuita ad alcun progetto di manutenzione evolutiva o di sviluppo.

Ci può essere la tendenza a classificare alcune funzionalità evolutive come lavoro di manutenzione ordinaria, ma tale tipo di attività dovrebbe essere monitorata e riportata separatamente. La motivazione usuale per includerle è legata alla velocità o alla convenienza. Le organizzazioni spesso prevedono un percorso rapido per richieste evolutive di piccola entità, solitamente di 40 ore o meno, per ridurre il sovraccarico sul progetto. Quando sono impattati i requisiti di business, si dovrebbe comunque applicare l'analisi dei function point, almeno per misurare i risultati.

Se un rilascio di progetto contiene una miscela di requisiti di manutenzione adeguativa, correttiva e/o perfettiva, occorre fare attenzione a come viene ripartito l'impegno tra queste attività, poiché le ultime due categorie contribuiscono con zero (0) function point al business. Mentre tale ripartizione dell'impegno lavorativo può essere relativamente facile durante la fase di realizzazione, in funzione del livello di granularità di classificazione degli impegni, essa è generalmente più complicata durante la maggior parte delle fasi finali del test. Un possibile approccio potrebbe essere una ripartizione dell'intero rilascio in base alla proporzionalità dei suoi contenuti.

Un progetto che comporta solo aggiornamenti della piattaforma, linguaggio o da un ambiente tecnico ad un altro, e che non cambia la funzionalità utente, non dovrebbe portare ad una evoluzione della misura della dimensione funzionale.

Attività	Inclusa nell'ambito di conteggio della manutenzione evolutiva
Correzione di errori in produzione ("break/fix")	No
Manutenzione perfettiva o preventiva	No
Aggiornamenti di piattaforma, rilasci di nuovo software di sistema	No
Progetto con contemporanee riparazioni (fix) ed evoluzioni	Parzialmente

**Richieste on-demand (ad hoc)**

Funzionalità fornite all'utente finale sotto forma di report e estrazioni di dati *una tantum* / a richiesta sono sicuramente conteggiabili. La decisione se contarle o meno dovrebbe essere presa valutando se le funzioni saranno mantenute in seguito e a quale necessità del business risponderà la relativa dimensione funzionale. Si deve notare che questa discussione è circoscritta a report / estrazioni di dati prodotti dal settore di Sviluppo S.I. e non include report o interrogazioni ad hoc generate dall'utente. Si dovrebbe notare che la metodologia per produrre report ad hoc non possiede generalmente un rigore analogo a quella di un intero progetto di manutenzione evolutiva. Per tale motivo, si dovrebbe usare particolare cautela nel paragonare i costi di una simile attività con quelli dell'attività di manutenzione evolutiva in generale.

Attività	Inclusa nell'ambito di conteggio della manutenzione evolutiva
Report <i>una tantum</i>	Convenzioni locali
Aggiornamenti di tabelle	No
Setup di operazioni (job) speciali	No
Correzione di dati	No
Modifiche di massa dei dati	Sì – come conversione, se associata a un progetto.



## Supporto all'utente finale

Qualsiasi impegno lavorativo non progettuale, legato a attività classificate come 'non conteggiabili', dovrebbe essere attribuito a una categoria di lavoro differente da *nuovo sviluppo* o *manutenzione evolutiva*. Per Stime Preliminari o Studi di Fattibilità il problema è che i requisiti utente non sono ancora ben definiti. Inoltre, in tali fasi solitamente non è ancora stato sovvenzionato alcun progetto (e potrebbe non esserlo mai). Al massimo, si può determinare una stima di ordine di grandezza approssimato (ROM – Rough Order of Magnitude) dei function point, ma in tale fase non si dovrebbe applicare alcuna misura quantitativa. Qualsiasi risultato numerico è dato solo a scopo di budget e pianificazione.

Generiche attività non progettuali di supporto all'utente, come rispondere a domande del tipo '*what-if*' e fornire assistenza all'utente, non dovrebbero essere oggetto dell'analisi dei function point.

Attività	Inclusa nell'ambito di conteggio della manutenzione evolutiva
Stima preliminare o analisi di fattibilità	Al massimo, ordine di grandezza approssimato (ROM – Rough Order of Magnitude)
Risposte a domande di tipo ' <i>what-if</i> '	No
Generico supporto non progettuale al cliente	No
Supporto di tipo <i>help desk</i>	Parzialmente

## Conclusioni

Questo capitolo ha solo l'intento di fornire una guida e non deve essere interpretato come un insieme di regole assolute. Si possono sempre trovare delle eccezioni agli scenari proposti. Ogni esempio deve essere valutato in base ai requisiti utente e alla situazione specifica di business. Inoltre, ogni organizzazione IT ha proprie attività e propri processi di sviluppo unici. Questi dovrebbero essere affrontati in una specifica Linea Guida interna ai Function Point, che consideri tutti gli aspetti della FPA, incluse le assunzioni e qualsiasi interpretazione particolare del Manuale delle Regole di Conteggio (CPM).

## Sommario

Mentre la maggior parte delle risorse di sviluppo applicativo IT sono di solito dedicate al supporto o alla manutenzione evolutiva di applicazioni esistenti, la maggior parte degli esempi e delle illustrazioni presenti nella Parte 4 sono rivolti al caso di nuovi sviluppi. Il presente capitolo fornisce una guida su cosa contare e un approfondimento sul processo di identificazione nell'ambito della manutenzione evolutiva.

Il primo passo è identificare correttamente le funzioni di tipo dati e di tipo transazionale che sono state aggiunte, modificate o cancellate in base ai requisiti utente. Per le modifiche, la parte che più spesso viene trascurata è quella relativa alle modifiche che coinvolgono il solo trattamento logico interno. La Parte 1 contiene le definizioni dei trattamenti logici, utili per identificare le modifiche apportate alle funzioni di tipo transazionale che si dovrebbero contare. Infine, ogni Funzione di tipo dati e di tipo transazionale deve essere contata in accordo con le regole della Parte 1.

È importante anche separare le differenti manutenzioni e le attività di supporto, e l'impegno ad esse associato, come illustrato nella sezione "Considerazioni su manutenzione evolutiva e manutenzione ordinaria".

---

### Attività di conversione dei dati

---

**Introduzione** Questa sezione affronta le funzionalità da valutare quando vi siano dei requisiti di migrazione o di conversione dei dati in concomitanza con un progetto di nuovo sviluppo o di manutenzione evolutiva o di spostare un'applicazione su una nuova piattaforma. La Parte 4 del CPM fornisce ulteriori esempi di funzioni di tipo dati e di tipo transazionale per la conversione di dati.

**Contenuti** Questo capitolo discute quanto segue per illustrare vari scenari di conversione:

Argomento	Pagina
Funzionalità di Conversione	5-2
Scenario 1: Conversione di Dati per un Progetto di Manutenzione Evolutiva	5-3
Scenario 2: Conversione di Dati con EIF Referenziati	5-3
Scenario 3: Assegnazione di Valori di Default	5-3
Cosa Non È una Funzionalità di Conversione	5-4
Riepilogo	5-4

## Funzionalità di conversione

La conversione dei dati applicativi si basa sul punto di vista dell'utente dei dati. Gli utenti identificano i requisiti sui dati in base a esigenze distinte, come per esempio dati relativi al Personale, alla Contabilità, ai Clienti o all'Inventario. Il punto di vista dell'utente di questi dati comprende tutti gli attributi associati al gruppo di dati così come è definito nell'applicazione. Questo gruppo di dati riconoscibile dall'utente e gli attributi dei dati ad esso associati diventano la base per un gruppo logico di dati che risponde ad uno specifico requisito dell'utente. È questo file logico che richiede che tutti i suoi attributi dei dati siano mantenuti come parti di un unico insieme (collegate e non indipendenti).

Ulteriori attributi potrebbero essere richiesti a causa di requisiti applicativi nuovi o modificati. Come parte di una manutenzione evolutiva, potrebbe essere necessario convertire e popolare gli attributi dei dati aggiunti come parte del progetto di manutenzione evolutiva. La percezione del processo di conversione si basa sull'applicazione originale, sui file logici oggetto di conversione e sui requisiti relativi ai dati della nuova applicazione.

Il processo di conversione si svolge rispetto a tutti i dati così come sono visti dall'utente per creare un file logico aggiornato che risponde a specifici requisiti utente relativi ai dati applicativi nuovi/convertiti.

Si applichino le regole di identificazione standard dei processi elementari (EP, Elementary Process) per identificare le funzionalità di conversione. Il processo elementare include qualsiasi report delle eccezioni, degli errori, di conversione o di controllo, richiesto per assicurare l'integrità dei dati oggetto di conversione. L'ILF o gli ILF nuovi o modificati dell'applicazione sono popolati dai dati convertiti e i requisiti utente relativi ad essi dettano ciò che si richiede da parte della o delle vecchie applicazioni per soddisfare i requisiti utente funzionali del progetto.

### Scenario 1: Conversione di dati per un progetto di manutenzione evolutiva

Il progetto comporta l'integrazione di una funzione di Competenze RU a livello globale, che un reparto aveva in precedenza implementato come un'applicazione a sé stante. C'è un requisito *una tantum* per raccogliere tutti i dati sulle competenze esistenti e popolare degli attributi di dati esistenti in un ILF di un'applicazione RU esistente. I dati sulle competenze esistenti che devono essere importati saranno contati come un EI. C'è un report di controllo ed errori che viene prodotto per assicurare l'integrità della migrazione. Questo processo sarà eseguito come parte dell'installazione delle nuove funzionalità. C'è un solo processo elementare per popolare inizialmente i nuovi attributi dei dati in un ILF del sistema RU, comprendente il report di controllo ed errori. Il processo di conversione sarà contato come un EI, che sarà incluso nella Dimensione Funzionale del Progetto di Manutenzione Evolutiva, ma che non sarà sommato alla Dimensione Funzionale dell'Applicazione, poiché è un processo *una tantum*.

### Scenario 2: Conversione di dati con EIF referenziati

L'utente ha richiesto che un ILF (o parte di un ILF) sia popolato a partire da un ILF di un'altra applicazione. In questo esempio, si richiede anche la validazione rispetto a un ulteriore ILF di una terza applicazione. Questo processo è specificato come processo *una tantum* e i dati referenziati nella terza applicazione non saranno utilizzati in seguito.

Gli attributi che devono essere caricati fungono da transazione di input per popolare l'ILF di destinazione e saranno contati come un EI. I dati referenziati nella terza applicazione per la validazione saranno contati come un EIF e come un ulteriore FTR. Anche l'ILF di destinazione sarà considerato come un FTR. Sia l'EI sia l'EIF dovrebbero essere inclusi nella misurazione del progetto, ma non sommati alla dimensione funzionale dell'applicazione.

### Scenario 3: Assegnazione di valori di default

Un progetto di manutenzione evolutiva richiede l'aggiunta di un DET a un ILF esistente. Il nuovo DET sarà popolato con uno specifico valore di default. Sebbene l'ILF e qualsiasi transazione variata saranno contati come modificati, non si conta alcuna funzionalità di conversione. Nessun dato attraversa il confine per fissare il valore di default.

## Cosa non è una funzionalità di conversione

Questa sezione descrive vari casi che non sono considerati di Conversione.

- Non contare aggiornamenti (*upgrade*) software dovuti all'installazione di una versione aggiornata di pacchetti commerciali (*vendor packages*) come funzionalità di conversione.
- Non contare la migrazione di un'applicazione su una nuova piattaforma come funzionalità di conversione.
- Non contare la conversione di dati svolta tramite un'utilità di caricamento esistente. Nessuna funzionalità è stata specificatamente sviluppata per svolgere tale conversione.
- Anche se un EIF dell'applicazione oggetto di misurazione è stato modificato, non può esserci alcuna funzionalità di conversione. Solo l'applicazione che conta quella funzione di tipo dati come ILF può contare una funzionalità di conversione.

## Riepilogo

Quando si aggiunge o si modifica un ILF, è possibile che sia richiesto un processo di conversione per popolare il nuovo ILF o uno o più DET di un ILF esistente. Parte dell'analisi consiste nell'identificare cosa attraversa il confine. Nel caso di un nuovo sviluppo, si ritiene che l'archivio o gli archivi di dati o l'ILF o gli ILF esistenti dei sistemi che vengono rimpiazzati attraversino il confine. Quando una manutenzione evolutiva implica un ILF modificato e si richiede un trattamento logico per popolare un nuovo attributo (per es. validazioni, confronti logici, ecc.), si può ritenere che l'ILF esistente attraversi il confine come un EI. Se un nuovo attributo di un ILF è popolato soltanto con un valore di default o un valore nullo, non si dovrebbe contare alcuna conversione, poiché non c'è nulla che attraversi il confine.

---

## Indice analitico – Parte 3

---

**A**

Aggiornamento

definizione, 3-3

Attributi non-chiave, 2-10

Attributi tecnici, 2-9, 2-24

Approccio, 3-4

Attributi chiave, 2-23

Chiave Esterna (FK), 2-23

Chiave Primaria (PK), 2-23

Chiave Secondaria (SK), 2-23

Attributi, 2-6, 2-21

Attribuzione, 2-23

**B****C**

Caricamento (Load)

definizione, 3-3

Convenzioni terminologiche degli scenari, 3-5

Campi ripetuti, 2-27

Concetti

(in-)dipendenza dell'entità, 2-12, 2-15

relazione opzionale/obbligatoria, 2-12, 2-15

Copia

definizione, 3-3

**D**

Database relazionali, 2-4

Diagramma della soluzione

Scenario 1, 3-8

Scenario 2, 3-10

Scenario 3, 3-12

Scenario 4, 3-14

Scenario 5, 3-16

Scenario 6, 3-17

Scenario 7, 3-19

Scenario 8, 3-21

Dati di business

definizione, 1-4

definizioni, 1-4

caratteristiche logiche, 1-4

esempi, 1-5

caratteristiche fisiche, 1-5

Dipendenza dell'entità, 2-12

Dati di decodifica

quali dati non sono, 1-13

definizione, 1-6

esempi, 1-7

caratteristiche logiche, 1-7

caratteristiche fisiche, 1-7

origine, 1-8

identificazione, 1-10

quali dati sono, 1-10

Dati di riferimento

definizione, 1-5

esempi, 1-6

caratteristiche logiche, 1-5

caratteristiche fisiche, 1-6

Data Element Type (DET)

regole, 2-24

Definizioni

manutenzione adeguativa, 4-20

Entità associativa, 2-3

- Tipo entità associativa, 2-5
- Attributi, 2-6, 2-21
- Tipo entità attributiva, 2-6
- dati di decodifica, 1-6
- copia, 3-3
- manutenzione correttiva, 4-20
- Elemento dati, 2-4, 2-21
- Data Element Type (DET), 2-24
- Oggetto dati, 2-4
- Entità, 2-3
- Sottotipo di entità, 2-3, 2-6
- Tipo di entità, 2-3, 2-5
- file, 3-3
- File, 2-4
- File System, 2-4
- immagine, 3-3
- caricamento (load), 3-3
- unione (merge), 3-3
- Normalizzazione, 2-6
- manutenzione perfetta, 4-20
- intento primario, 3-2
- Record, 2-4
- dati di riferimento, 1-5
- aggiornamento, 3-3
- Relazioni, 2-6
- E**
- Entità intersezione, 2-10
- Entità chiave-a-chiave, 2-10
- Entità Attributiva, 2-37
  - obbligatoria, 2-37
  - opzionale, 2-37
- Entità Associativa, 2-3, 2-10, 2-34
  - non contata, 2-34
  - contata come RET, 2-35
  - contata come file logico, 2-36
- Elemento dati, 2-4, 2-21
  - esempio di attributi singoli/multipli, 2-25
  - riconoscibile dall'utente, 2-25
  - esempio di indirizzi, 2-26
  - esempio di nomi, 2-26
  - esempio di campi ripetuti, 2-27
  - esempio di campi di stato, 2-28
  - esempio di chiavi esterne, 2-29
  - esempio di date di sistema, 2-29
  - misurare le funzioni di tipo dati, 4-2
- Entità, 2-3
  - non mantenuta, 2-9
- Sottotipo di entità, 2-3, 2-6, 2-32, 2-38,
  - contato come RET, 2-38
  - non contato, 2-39
- Esempi
  - funzionalità aggiunte, 4-14
  - funzionalità modificate, 4-15
  - funzionalità cancellate, 4-15
- F**
- Funzionalità aggiunte
  - esempio, 4-14
  - identificazione, 4-10
- Funzionalità modificate
  - esempio, 4-15
  - identificazione, 4-10
- Funzionalità di conversione
  - identificazione, 4-11
- Funzionalità di conversione, 4-2
- Funzionalità cancellate
  - esempio, 4-15
  - identificazione, 4-11
- File, 2-4
  - definizione, 3-3
- Files System, 2-4
- Funzionale, 3-5
- File indice, 2-9
- G**
- Gruppi/dati ripetuti
  - contati come RET, 2-40
  - non contati come RET, 2-40
- I**
- Immagine
  - definizione, 3-3
- Indipendenza dell'entità, 2-12
- Intento primario
  - definizione, 3-2
- K**
- L**
- File logici
  - processo per stabilire l'insieme dei, 2-2
- M**
- Manutenzione adeguativa
  - definizione, 4-20
- Manutenzione perfetta
  - definizione, 4-20
- Modello fisico dei dati, 2-9
- Manutenzione correttiva
  - definizione, 4-20
- N**
- Nome dell'attributo, 2-22
- Normalizzazione, 2-6
- O**
- Oggetto dati, 2-4
- P**
- Progetto di manutenzione evolutiva
  - misurare le funzioni di tipo transazionale, 4-3
  - trattamento logico, 4-5
- Progetto di manutenzione evolutiva
  - Conversione di dati, 4-3
- Punto di vista dell'utente, 2-9
- Punto di vista del business. *Vedi Punto di vista dell'utente*
- Processo di classificazione
  - File logici, 2-2
- Processo di identificazione
  - Data Element Types (DET), 2-2



- File logici, 2-2
  - tramite il metodo dei processi elementari, 2-9
  - tramite il metodo della (In-)Dipendenza, 2-9
- Record Element Types (RET), 2-2
- R**
- Richieste On-demand (ad-hoc), 4-22
- Richieste di manutenzione ordinaria, 4-21
- Record, 2-4
- Relazione, 2-6
  - Obbligatoria 1-1, 1-N, 2-6, 2-14
  - natura della, 2-14
  - Opzionale 1-(N), (1)-(N), (1)-N, 2-6, 2-14
- Riga, 2-4
- S**
- Sintesi degli scenari, 3-5
- Supporto all'utente finale, 4-23
- Scenario 4: Copia Immagine / Carica unica tabella
  - fisica - senza ulteriore elaborazione, , 3-13
- Scenari
  - Copia e Unione, 3-15
  - Copia/Carica immagine - senza ulteriore elaborazione, 3-11
  - Lettura, 3-7
  - Scenario 4, 3-13
  - Cattura da schermo, 3-17
  - Dati di transazione standard, 3-20
  - Copia Immagine Statica, 3-9
  - Aggiornamento del medesimo archivio dati, 3-18
  - Significatività per il business, 2-13
  - Sottogruppo, 2-31
  - Simboli usati nella soluzione del diagramma, 3-4
- T**
- Tipi di dati
  - di business, 1-4
  - di decodifica, 1-6
  - di riferimento, 1-5
- Tipo di Entità Associativa, 2-5, 2-32
- Tipi di dati di decodifica
  - statici o costanti, 1-11
  - sostitutivi, 1-11
  - valori validi, 1-12
- Tabella, 2-4
- Tecnico, 3-6
- Tupla, 2-4
- Tipo di entità, 2-3, 2-5
- Tipo di Entità Attributiva, 2-6, 2-32
- U**
- Unione (Merge)
  - definizione, 3-3

Questa pagina è lasciata intenzionalmente vuota.

---

## **Parte 4 – Esempi**

---

---

Questa pagina è lasciata intenzionalmente vuota.

---

---

## Parte 4 Esempi

---

**Introduzione** La Parte 4 fornisce esempi di funzioni di tipo dati e funzioni di tipo transazionale per illustrare le Regole di conteggio della Parte 1. Gli esempi includono:

- una descrizione delle funzioni di tipo dati o di tipo transazionale,
- le basi per la misura,
- le tabelle da usare durante l'applicazione delle regole di conteggio,
- la determinazione della complessità funzionale,
- il loro contributo alla dimensione funzionale.

**Contenuti** La Parte 4 include le seguenti sezioni:

<b>Argomento</b>	<b>Pagina</b>
<b>Esempi di conteggio delle funzioni di tipo dati</b>	<b>1-3</b>
ILF: Esempi di conteggio	1-7
EIF: Esempi di conteggio	1-37
<b>Esempi di conteggio delle funzioni di tipo transazionale</b>	<b>2-1</b>
Processi elementari: esempi di identificazione	2-7
EI: Esempi di conteggio	2-61
EO: Esempi di conteggio	2-103
EQ: Esempi di conteggio	2-127

---

Questa pagina è lasciata intenzionalmente vuota.

---

## Esempi di conteggio delle funzioni di tipo dati

---

**Introduzione** Questa sezione usa molti esempi per illustrare le procedure di misurazione delle funzioni di tipo dati, ciascuno dei quali è a sé stante.

**Nota:** Ogni esempio mostra solo i requisiti specifici della situazione che si sta illustrando, anche se in pratica noi dobbiamo valutare tutti i requisiti ed il loro impatto funzionale.

Questa sezione utilizza un'applicazione Risorse Umane (RU) insieme a un'applicazione Sicurezza e a un'applicazione Distribuzione Posta per illustrare le procedure di identificazione e misurazione delle funzioni di tipo dati. In aggiunta a questa sezione, vi sono esempi nei Casi di Studio inclusi nella documentazione supplementare IFPUG.

**Nota:** Gli esempi in questa sezione e in tutto il manuale hanno due obiettivi:

1. Illustrare come si applicano le Regole di conteggio dei function point per un dato insieme di requisiti utente
2. Permettere di esercitarsi sulle procedure di conteggio.

Ciascun conteggiatore deve:

- analizzare gli specifici requisiti utente che si applicano a ciascun progetto o applicazione oggetto di misurazione, e
- effettuare il conteggio sulla base di tali requisiti.

**Contenuti** Questa sezione spiega come sono organizzati gli esempi e comprende esempi dettagliati per il conteggio degli ILF e degli EIF.

Argomento	Pagina
<b>Esempi di conteggio delle funzioni di tipo dati</b>	<b>1-3</b>
ILF: Esempi di conteggio	1-7
EIF: Esempi di conteggio	1-37

## Organizzazione degli esempi di conteggio

Questa sezione spiega come sono presentati gli esempi.

### Schema dell'organizzazione

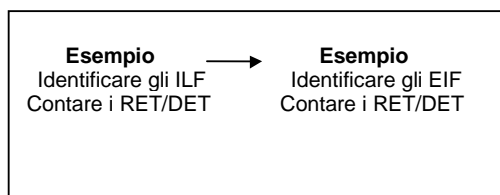
La seguente lista illustra le informazioni di dettaglio contenute negli esempi.

Per ogni esempio:

1. Sono identificate le funzioni di tipo dati
2. Le funzioni di tipo dati sono classificate come ILF o come EIF
3. Sono identificati e contati i RET e i DET che contribuiscono alla complessità funzionale.

### Diagramma dell'organizzazione

Il seguente diagramma illustra l'organizzazione degli esempi.



### Conteggio per ogni esempio

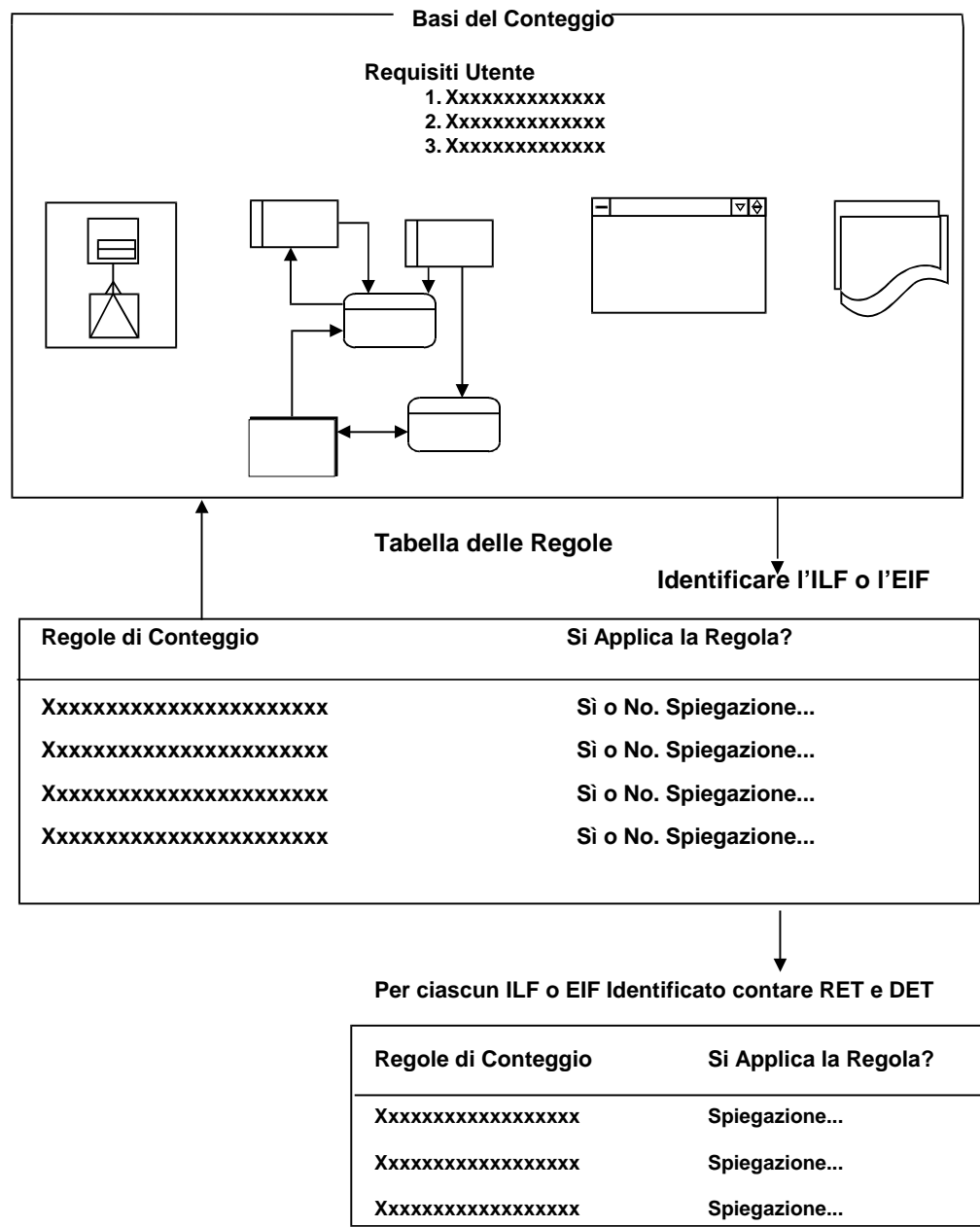
Ogni esempio include le seguenti componenti:

1. Basi per la misurazione.
2. Tabelle per l'applicazione delle regole di conteggio.



Diagramma delle componenti

Il seguente diagramma illustra le componenti per ogni esempio e il flusso delle informazioni.



**Basi per la misurazione**

Ogni esempio inizia con le basi per la misurazione. Come mostrato nel diagramma dei componenti, la misurazione può basarsi sui seguenti elementi:

- requisiti utente,
- modelli dei dati e dei processi,
- finestre, schermate o report.

**Nota:** Non tutti i componenti del diagramma sono inclusi in tutti gli esempi. In alcuni esempi, i requisiti costituiscono l'unica base per la misurazione. Altri esempi includono il modello dei dati o dei processi, finestre, schermate e report.

**Tabella delle regole di conteggio**

L'analisi per identificare le funzioni è presentata in una tabella che elenca le regole per il conteggio relative al tipo di funzione considerato. Le regole sono applicate ai componenti che costituiscono la base per la misurazione. L'analisi è spiegata nella tabella tramite la colonna "Si applica la regola?".

**Nota:** Se tutte le regole si applicano, l'esempio è contato come un ILF o EIF.

Le tabelle successive mostrano le regole e le spiegazioni per la complessità per ciascun tipo di funzione identificata.

---

## ILF: Esempi di conteggio

---

**Introduzione** Questa sezione utilizza un'applicazione Risorse Umane (RU) per illustrare le procedure di identificazione e misurazione delle funzioni di tipo dati. In aggiunta a questa sezione, vi sono esempi nei Casi di Studio inclusi nella documentazione supplementare IFPUG.

**Contenuti** Questa sezione include i seguenti esempi:

Argomento	Pagina
Descrizione sintetica degli esempi di conteggio degli ILF	1-8
Esempio: Dati di controllo per interrogazioni e report	1-9
Esempio: Definizione prospetto	1-15
Esempio: Indice secondario	1-20
Esempio: Dati condivisi dell'applicazione	1-21
Esempio: Utenti differenti / Viste dati differenti	1-29

## Descrizione sintetica degli esempi di conteggio di ILF

Gli esempi per gli ILF sono descritti nella seguente tabella.

<b>Esempio</b>	<b>Descrizione Sintetica</b>	<b>Pagina</b>
Dati di controllo per interrogazioni e prospetti	Questo esempio mostra l'analisi e la misurazione dei dati che sono mantenuti per motivi di controllo.	1-9
Definizione prospetto	Questo esempio mostra come contare le definizioni di report fornite dall'utente e mantenute all'interno del confine dell'applicazione.	1-15
Indice secondario	Questo esempio illustra l'analisi dei requisiti utente per l'esempio di definizione dei report che si focalizza sui requisiti per la realizzazione fisica.	1-20
Dati condivisi dell'applicazione	Questo esempio mostra come contare i dati mantenuti da più di una applicazione.	1-21
Utenti differenti / Viste dati differenti	Questo esempio mostra che due applicazioni possono conteggiare lo stesso file con DET differenti.	1-29

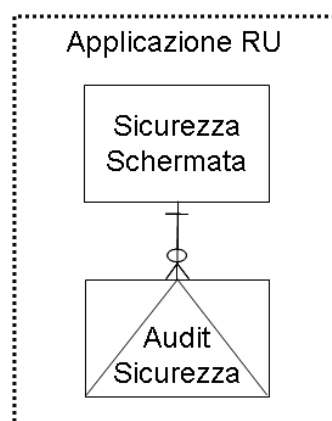
## Esempio: Dati di controllo per interrogazioni e report

### Requisiti utente

L'analisi dei seguenti requisiti utente sulla sicurezza ha evidenziato la necessità di dati di controllo (*audit*):

1. Permettere o impedire l'accesso utente a livello di ciascuna schermata dell'applicazione.
2. Cambiare l'accesso utente per ciascuna schermata.
3. Produrre report sugli inserimenti o variazioni dei livelli di sicurezza associati ad ogni schermata, riportando i seguenti dati:
  - Identificativo dell'utente che effettua le operazioni di inserimento o variazione delle informazioni di sicurezza
  - L'utente e la schermata, i cui livelli di sicurezza sono stati aggiunti o variati
  - L'utente e la schermata (i cui livelli sono stati aggiunti o variati) prima o dopo l'effettuazione del cambiamento
  - Data e ora dell'operazione di inserimento o variazione.
4. Catturare i dati di controllo per monitorare e produrre a livello giornaliero report sulle attività di sicurezza. Questo requisito è stato determinato durante la fase di disegno per soddisfare i requisiti utente sui livelli di sicurezza delle schermate.

### ERD



#### Legenda:



Entità

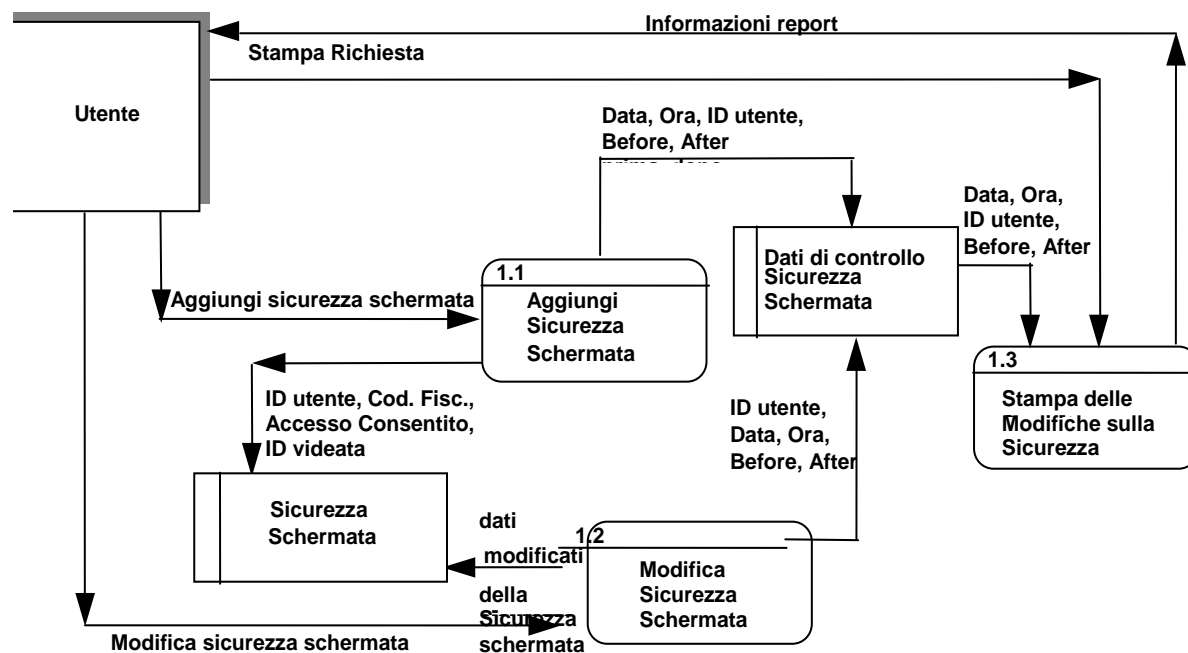


Entità Attributiva

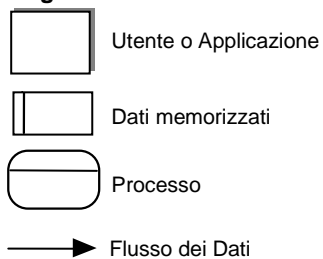


Relazione Opzionale 1 : N

## Diagramma di flusso dei dati



### Legenda:



## Passo 1 Identificare le funzioni di tipo dati

Usare le Regole di identificazione delle funzioni di tipo dati per determinare se la rappresentazione dei Dati di Controllo sulla Sicurezza è una funzione di tipo dati. La seguente tabella mostra l'analisi per la rappresentazione dei Dati di Controllo sulla Sicurezza.

Regole di identificazione delle funzioni di tipo dati	Si applica la regola?
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	Sicurezza Schermata e Dati di Controllo Sicurezza Schermata.
2. Escludere entità non mantenute da alcuna applicazione.	Non ci sono entità di questo tipo.
3. Raggruppare entità collegate dipendenti.	Sicurezza Schermata e Dati di Controllo Sicurezza Schermata sono collegati. Dati di Controllo Sicurezza Schermata è dipendente da Sicurezza Schermata. Essi sono raggruppati insieme in una sola funzione di tipo dati.
4. Escludere entità referenziate come dati di decodifica.	Non ci sono entità di questo tipo.
5. Escludere entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non ci sono entità di questo tipo.

Dati di Controllo Sicurezza Schermata non è contata come una funzione di tipo dati a sé stante perché è dipendente da Sicurezza Schermata. Dati di Controllo Sicurezza Schermata è parte della funzione di tipo dati Sicurezza Schermata.

## Passo 2 Classificare le funzioni di tipo dati

La seguente tabella mostra l'analisi per determinare se le informazioni Sicurezza Schermata sono classificate come un ILF.

Regole di classificazione delle funzioni di tipo dati	Si applica la regola?
1. Classificare come ILF, se il gruppo di dati è mantenuto dall'applicazione che si sta misurando.	La funzione di tipo dati Sicurezza Schermata è mantenuta all'interno dell'applicazione.
2. Classificare come un EIF, se il gruppo di dati: <ul style="list-style-type: none"> <li>• è referenziato, ma non mantenuto, dall'applicazione che si sta misurando, e</li> <li>• è identificata come ILF in una o più ulteriori applicazioni.</li> </ul>	È classificata come ILF; di conseguenza, non sono identificati EIF.

In base all'analisi, Sicurezza Schermata è classificata come un ILF.

## Passo 3 Contare i DET

**Per i DET**, esaminare ciascun attributo associato all'ILF Sicurezza Schermata e determinare se si applicano le Regole di conteggio dei DET.

L'ILF Sicurezza Schermata comprende:

- ID Utente
- Cod. Fisc.
- ID Videata
- Accesso Consentito
- Data modifica
- Ora Modifica
- *Before Image*
  - ID Utente Before
  - ID Videata Before
  - Accesso Consentito Before
- *After Image*
  - ID Utente After
  - ID Videata After
  - Accesso Consentito After



Di seguito l'analisi per la definizione dei DET dell'ILF Sicurezza Schermata.

Regole di conteggio dei DET	Si applica la regola?
1. Contare un DET per ciascun campo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di un qualsiasi processo elementare nell'ambito del conteggio.	ID Utente, Cod. Fisc., ID Videata, Accesso Consentito, Data Modifica e Ora Modifica.
2. Quando due applicazioni mantengono e/o referenziano la stessa funzione di tipo dati, contare solo i DET effettivamente utilizzati dall'applicazione misurata.	Non ci sono attributi di questo tipo.
3. Contare un DET per ogni singolo dato richiesto dall'utente per stabilire una relazione con un'altra funzione di tipo dati	Non ci sono attributi di questo tipo.
4. Riesaminare gli attributi correlati per determinare se questi sono raggruppati e conteggiati come un singolo DET oppure se sono conteggiati come DET multipli; il raggruppamento dipenderà da come i processi elementari utilizzano gli attributi nell'ambito dell'applicazione.	ID Utente Before, ID Videata Before e Accesso Consentito Before sono raggruppati e contati come un solo DET per la <i>Before Image</i> . Si fa lo stesso anche per gli attributi della <i>After Image</i> .

#### Passo 4

#### Contare i RET

Per i RET, identificare i sottogruppi in base alle regole di conteggio dei RET.

Regole di conteggio dei RET	Si applica la regola?
1. Conta un RET per ogni funzione di tipo dati (cioè, di default ogni funzione di tipo dati ha un sottogruppo di DET che deve essere conteggiato come un RET).	Conta un RET per l'ILF Sicurezza Schermata.
2. Conta un RET aggiuntivo per ognuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contiene più di un DET: <ul style="list-style-type: none"> <li>entità associativa con attributi non-chiave,</li> <li>entità sotto-tipo (ulteriore rispetto al primo dei sotto-tipi), e</li> <li>entità attributiva, in una relazione 1-1 non obbligatoria.</li> </ul>	<p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p> <p>Dati di Controllo Sicurezza Schermata è una entità attributiva in una relazione opzionale 1 : M. Conta un RET aggiuntivo per Dati di Controllo Sicurezza Schermata.</p>

**I totali dei RET e dei DET** per Sicurezza Schermata sono riportati nella seguente tabella.

<b>RET</b>	<b>DET</b>
<ul style="list-style-type: none"> <li>• Sicurezza Schermata</li> <li>• Dati di Controllo Sicurezza Schermata</li> </ul>	<ul style="list-style-type: none"> <li>• ID Utente</li> <li>• Cod. Fisc.</li> <li>• ID Videata</li> <li>• Accesso Consentito</li> <li>• Data Modifica</li> <li>• Ora Modifica</li> <li>• Before Image</li> <li>• After Image</li> </ul>
<b>Totale</b> 2 RET	<b>Totale</b> 8 DET

**Passo 5**      **Determinare la complessità funzionale**

2 RET e 8 DET	La complessità è Bassa
---------------	------------------------

**Passo 6**      **Determinare la dimensione funzionale**

Dimensione Funzionale di 1 ILF di bassa complessità	7 FP
---	------

## Esempio: Definizione report

**Requisiti  
utente**

L'utente richiede la possibilità di eseguire le seguenti attività:

1. Inserire la definizione di un report che includa:
  - un identificativo unico per il report
  - un nome del report
  - gli attributi utilizzati nel report
  - i calcoli per generare il report.
2. Riusare il report definito in qualsiasi momento, anche per cambiare la definizione, se necessario.
3. Visualizzare e stampare un report usando la sua definizione.
4. Interrogare le definizioni dei report esistenti utilizzando il nome del report o l'identificativo.

**Passo 1****Identificare le funzioni di tipo dati**

In base ai requisiti utente, l'identificativo report, il nome report, gli attributi sul report e i calcoli costituiscono insieme un raggruppamento logico di dati per la definizione di un report, poiché sono mantenuti come un gruppo.

La seguente tabella illustra l'analisi per determinare se le informazioni sulla definizione di un report sono una funzione di tipo dati. Consultare i Casi di Studio per vedere come possono essere contati i requisiti rimanenti.

<b>Regole di identificazione delle funzioni di tipo dati</b>	<b>Si applica la regola?</b>
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	Definizione Report.
2. Escludere entità non mantenute da alcuna applicazione.	Non ci sono entità di questo tipo.
3. Raggruppare entità collegate dipendenti.	Non ci sono entità di questo tipo.
4. Escludere entità referenziate come dati di decodifica.	Definizione Report non è una istanza di dati di decodifica. Questa entità è usata come riferimento nella generazione dei report, consiste in più di un semplice codice e descrizione ed è cambiata in base alle esigenze dell'utente.
5. Escludere entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non ci sono entità di questo tipo.

**Passo 2**      **Classificare le funzioni di tipo dati**

La seguente tabella illustra l'analisi per determinare se le informazioni sulla Definizione Report sono classificate come un ILF. Consultare i Casi di Studio per vedere come possono essere contati i requisiti rimanenti.

Regole di classificazione delle funzioni di tipo dati	Si applica la regola?
1. Classificare come ILF, se il gruppo di dati è mantenuto dall'applicazione che si sta misurando.	La funzione di tipo dati Definizione Report è mantenuta all'interno dell'applicazione.
2. Classificare come un EIF, se il gruppo di dati: <ul style="list-style-type: none"><li>• è referenziato, ma non mantenuto, dall'applicazione che si sta misurando, e</li><li>• è identificata come ILF in una o più ulteriori applicazioni.</li></ul>	È classificata come ILF; di conseguenza, non sono identificati EIF.

In base all'analisi, le informazioni Definizione Report sono classificate come un ILF.

**Passo 3**      **Contare i DET**

**Per i DET**, esaminare ciascun attributo associato all'ILF Definizione Report e determinare se si applicano le Regole di conteggio dei DET.

L'ILF Definizione Report comprende:

- ID Report
- Nome Report
- Attributi
- Calcoli

Di seguito è illustrata l'analisi per la definizione dei DET per l'ILF Definizione Report.

Regole di conteggio dei DET	Si applica la regola?
1. Contare un DET per ciascun campo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di un qualsiasi processo elementare nell'ambito del conteggio.	ID Report, Nome Report, Attributi, Calcoli. Anche se ci sono occorrenze multiple di Attributi e Calcoli, ognuno di questi è contato come un solo DET.
2. Quando due applicazioni mantengono e/o referenziano la stessa funzione di tipo dati, contare solo i DET effettivamente utilizzati dall'applicazione misurata.	Non ci sono attributi di questo tipo.
3. Contare un DET per ogni singolo dato richiesto dall'utente per stabilire una relazione con un'altra funzione di tipo dati	Non ci sono attributi di questo tipo.
4. Riesaminare gli attributi correlati per determinare se questi sono raggruppati e conteggiati come un singolo DET oppure se sono conteggiati come DET multipli; il raggruppamento dipenderà da come i processi elementari utilizzano gli attributi nell'ambito dell'applicazione.	Non ci sono attributi di questo tipo.

#### Passo 4

#### Contare i RET

Per i RET, identificare i sottogruppi in base alle regole di conteggio dei RET.

Regole di conteggio dei RET	Si applica la regola?
1. Conta un RET per ogni funzione di tipo dati (cioè, di default ogni funzione di tipo dati ha un sottogruppo di DET che deve essere conteggiato come un RET).	Conta un RET per l'ILF Definizione Report.
2. Conta un RET aggiuntivo per ognuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contiene più di un DET: <ul style="list-style-type: none"> <li>entità associativa con attributi non-chiave,</li> <li>entità sotto-tipo (ulteriore rispetto al primo dei sotto-tipi), e</li> <li>entità attributiva, in una relazione 1-1 non obbligatoria.</li> </ul>	<p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p>

I totali dei **RET** e dei **DET** per l'ILF Definizione Report sono riportati nella seguente tabella.

<b>RET</b>	<b>DET</b>
<ul style="list-style-type: none"> <li>Definizione Report</li> </ul>	<ul style="list-style-type: none"> <li>ID Report</li> <li>Nome Report</li> <li>Attributi</li> <li>Calcoli</li> </ul>
<b>Totale</b> 1 RET	<b>Totale</b> 4 DET

**Passo 5****Determinare la complessità funzionale**

1 RET e 4 DET	La complessità è Bassa
---------------	------------------------

**Passo 6****Determinare la dimensione funzionale**

Dimensione Funzionale di 1 ILF di bassa complessità	7 FP
---	------

## Esempio: Indice secondario

### Requisiti utente

L'utente vuole interrogare la definizione dei report usando il nome del report come chiave di ricerca per trovare la definizione desiderata. Per soddisfare il requisito utente è stato creato un indice secondario usando il nome del report come chiave.

### Passo 1

#### Identificare le funzioni di tipo dati

La seguente tabella illustra l'analisi per determinare se l'indice secondario è una funzione di tipo dati.

Regole di identificazione delle funzioni di tipo dati	Si applica la regola?
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	No. Dal punto di vista dell'utente questa funzione di filtro fornisce all'utente specifici attributi delle definizioni report create che fanno riferimento all'ILF Definizione Report. Questo filtro tecnico, necessario per creare la lista di interrogazione, di per sé non costituisce una funzione applicativa.
2. Escludere entità non mantenute da alcuna applicazione.	Non applicabile.
3. Raggruppare entità collegate dipendenti.	Non applicabile.
4. Escludere entità referenziate come dati di decodifica.	Non applicabile.
5. Escludere entità che non contengono attributi richiesti dall'utente.	Non applicabile.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non applicabile.

In base all'analisi in tabella, l'indice secondario non è un gruppo logico, perciò non è una funzione di tipo dati.



## Esempio: Dati condivisi dell'applicazione

### Requisiti utente

L'utente RU richiede la possibilità di mantenere informazioni su ogni nuovo impiegato.

Le informazioni che devono essere mantenute dall'utente RU comprendono:

- ID Impiegato
- Nome Impiegato
- Indirizzo Postale Impiegato
- Livello Retributivo Impiegato
- Qualifica Impiegato.

(\*) Alla creazione di un record per un nuovo impiegato, dovrebbe essere calcolata automaticamente e salvata, insieme alle altre informazioni dell'impiegato, la data minima di pensionamento.

L'utente della Sicurezza richiede che sia assegnato ad ogni nuovo impiegato un livello di sicurezza. Il dipartimento della Sicurezza effettua un'indagine quando un impiegato è assunto e gli assegna l'appropriato livello di autorizzazione di sicurezza organizzativa. Non si tratta della sicurezza applicativa che determina l'accesso dei singoli utenti all'applicazione.

Le informazioni che devono essere mantenute dall'utente della Sicurezza comprendono:

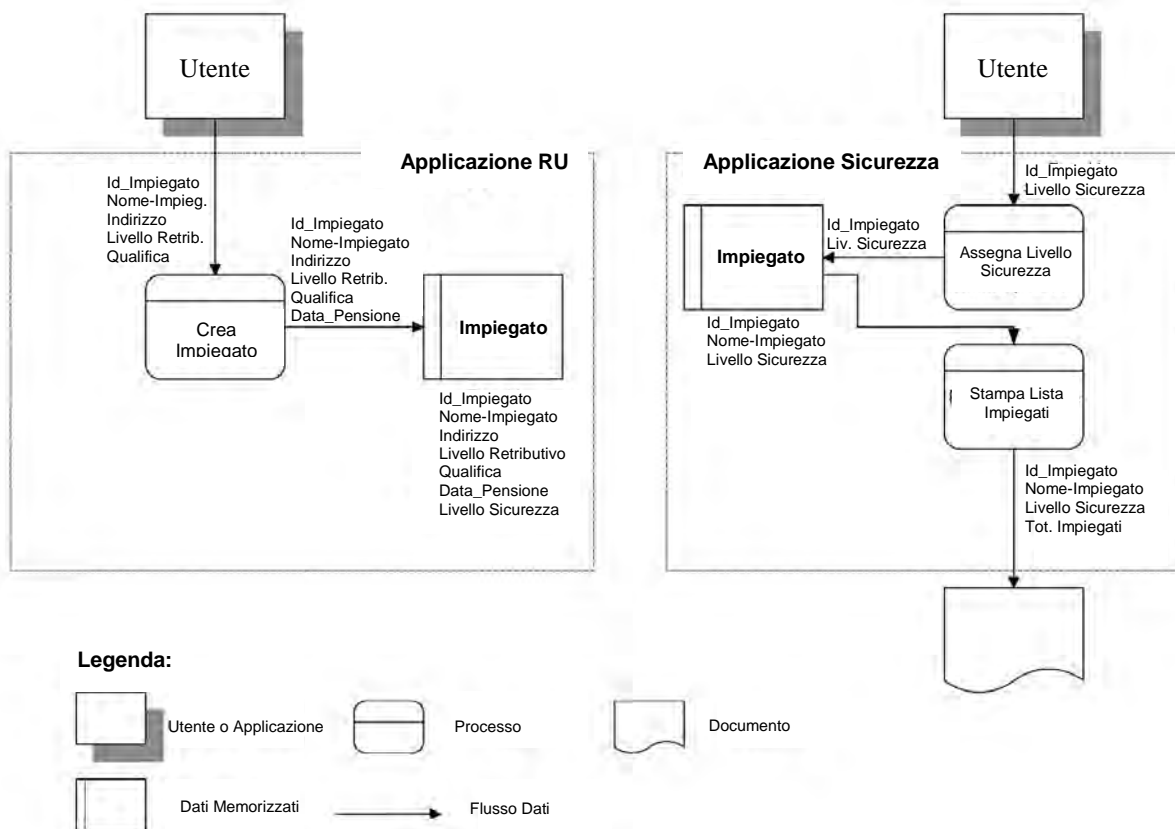
- ID Impiegato
- Livello di Autorizzazione di Sicurezza Organizzativa.

L'utente della Sicurezza richiede anche un report che riporti le seguenti informazioni:

- Totale degli ID Impiegato
- Nome Impiegato
- ID Impiegato
- Livello di Autorizzazione di Sicurezza Organizzativa

## Diagramma di flusso dei dati

Il diagramma seguente illustra il flusso dei dati per questo esempio.



**Passo 1****Identificare le funzioni di tipo dati (Applicazione RU)**

Determinare se le informazioni dell'Impiegato sono una funzione di tipo dati per l'applicazione RU. La seguente tabella illustra l'analisi sintetica.

Regole di identificazione delle funzioni di tipo dati	Si applica la regola?
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	Impiegato.
2. Escludere entità non mantenute da alcuna applicazione.	Non ci sono entità di questo tipo.
3. Raggruppare entità collegate dipendenti.	Non ci sono entità di questo tipo.
4. Escludere entità referenziate come dati di decodifica.	Non ci sono entità di questo tipo.
5. Escludere entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non ci sono entità di questo tipo.

**Passo 2****Classificare le funzioni di tipo dati (Applicazione RU)**

Determinare se Impiegato è classificato come un ILF per l'applicazione RU.

Regole di classificazione delle funzioni di tipo dati	Si applica la regola?
1. Classificare come ILF, se il gruppo di dati è mantenuto dall'applicazione che si sta misurando.	La funzione di tipo dati Impiegato è mantenuta all'interno dell'applicazione RU.
2. Classificare come un EIF, se il gruppo di dati: <ul style="list-style-type: none"> <li>• è referenziato, ma non mantenuto, dall'applicazione che si sta misurando, e</li> <li>• è identificata come ILF in una o più ulteriori applicazioni.</li> </ul>	È classificata come ILF; di conseguenza, non sono identificati EIF.

L'analisi indica che le informazioni sull'Impiegato sono classificate come un ILF per l'applicazione RU.

### Passo 3 Contare i DET (Applicazione RU)

**Per i DET**, esaminare ciascun attributo associato all'ILF Impiegato nell'applicazione RU e determinare se si applicano le Regole di conteggio dei DET.

La seguente lista comprende gli attributi per l'ILF Impiegato:

- ID Impiegato
- Nome Impiegato
- Indirizzo Postale Impiegato
- Livello Retributivo Impiegato
- Qualifica Impiegato
- Data Minima Pensionamento
- Livello di Autorizzazione di Sicurezza Organizzativa

Di seguito è illustrata l'analisi per la definizione dei DET per l'ILF Impiegato nell'applicazione RU.

Regole di conteggio dei DET	Si applica la regola?
1. Contare un DET per ciascun campo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di un qualsiasi processo elementare nell'ambito del conteggio.	I seguenti attributi soddisfano questa regola: <ul style="list-style-type: none"> <li>• ID Impiegato</li> <li>• Nome Impiegato</li> <li>• Indirizzo Postale Impiegato</li> <li>• Livello Retributivo Impiegato</li> <li>• Qualifica Impiegato</li> <li>• Data Minima Pensionamento</li> </ul>
2. Quando due applicazioni mantengono e/o referenziano la stessa funzione di tipo dati, contare solo i DET effettivamente utilizzati dall'applicazione misurata.	Tutti gli attributi sono utilizzati all'interno dell'applicazione RU <i>eccetto</i> Livello di Autorizzazione di Sicurezza Organizzativa.
3. Contare un DET per ogni singolo dato richiesto dall'utente per stabilire una relazione con un'altra funzione di tipo dati	Non ci sono attributi di questo tipo.
4. Riesaminare gli attributi correlati per determinare se questi sono raggruppati e conteggiati come un singolo DET oppure se sono conteggiati come DET multipli; il raggruppamento dipenderà da come i processi elementari utilizzano gli attributi nell'ambito dell'applicazione.	Non ci sono attributi di questo tipo.

**Passo 4 Contare i RET (Applicazione RU)**

**Per i RET**, identificare i sottogruppi in base alle regole di conteggio dei RET.

<b>Regole di conteggio dei RET</b>	<b>Si applica la regola?</b>
1. Conta un RET per ogni funzione di tipo dati (cioè, di default ogni funzione di tipo dati ha un sottogruppo di DET che deve essere conteggiato come un RET).	Contare un RET per l'ILF Impiegato.
2. Conta un RET aggiuntivo per ognuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contiene più di un DET: <ul style="list-style-type: none"> <li>entità associativa con attributi non-chiave,</li> <li>entità sotto-tipo (ulteriore rispetto al primo dei sotto-tipi), e</li> <li>entità attributiva, in una relazione 1-1 non obbligatoria.</li> </ul>	Non ci sono entità di questo tipo.  Non ci sono entità di questo tipo.  Non ci sono entità di questo tipo.

**I totali dei RET e dei DET** per l'ILF Impiegato nell'applicazione RU sono riportati nella seguente tabella.

<b>RET</b>	<b>DET</b>
<ul style="list-style-type: none"> <li>Impiegato</li> </ul>	<ul style="list-style-type: none"> <li>ID Impiegato</li> <li>Nome Impiegato</li> <li>Indirizzo Postale Impiegato</li> <li>Livello Retributivo Impiegato</li> <li>Qualifica Impiegato</li> <li>Data Minima Pensionamento</li> </ul>
<b>Totale</b> 1 RET	<b>Totale</b> 6 DET

**Passo 5 Determinare la complessità funzionale (Applicazione RU)**

1 RET e 6 DET	La complessità è Bassa
---------------	------------------------

**Passo 6 Determinare la dimensione funzionale (Applicazione RU)**

Dimensione Funzionale di 1 ILF di bassa complessità	7 FP
---	------

**Passo 1****Identificare le funzioni di tipo dati (Applicazione Sicurezza)**

Determinare se le informazioni dell'impiegato sono una funzione di tipo dati per l'applicazione Sicurezza. La seguente tabella illustra l'analisi sintetica.

<b>Regole di identificazione delle funzioni di tipo dati</b>	<b>Si applica la regola?</b>
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente all'interno dell'ambito di conteggio.	Impiegato.
2. Considerare le entità indipendenti come gruppi di dati logici separati.	Non ci sono entità di questo tipo.
3. Raggruppare in una funzione di tipo dati le entità collegate che sono entità dipendenti.	Non ci sono entità di questo tipo.
4. Escludere quelle entità che sono denominate come dati di decodifica.	Non ci sono entità di questo tipo.
5. Escludere le entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Eliminare le entità associative che contengono attributi addizionali non richiesti dall'utente e le entità associative che contengono soltanto le chiavi esterne; gruppi di chiavi esterne con le entità primarie.	Non ci sono entità di questo tipo.

**Passo 2****Classificare le funzioni di tipo dati (Applicazione Sicurezza)**

Determinare se Impiegato è classificato come un ILF per l'applicazione Sicurezza.

<b>Regole di classificazione delle funzioni di tipo dati</b>	<b>Si applica la regola?</b>
1. Classificare come un ILF, se i dati sono mantenuti dall'applicazione che si sta misurando.	La funzione di tipo dati Impiegato è mantenuta all'interno dell'applicazione Sicurezza.
2. Classificare come un EIF, se essa : <ul style="list-style-type: none"> <li>• è referenziata, ma non mantenuta, dall'applicazione che si sta misurando e</li> <li>• è identificata come un ILF in una o più altre applicazioni.</li> </ul>	È classificata come un ILF; di conseguenza, non sono identificati EIF.

L'analisi indica che Impiegato è classificato come un ILF anche per l'applicazione Sicurezza.

### Passo 3 Contare i DET (Applicazione Sicurezza)

**Per i DET**, esaminare ciascun attributo associato all'ILF Impiegato nell'applicazione Sicurezza e determinare se si applicano le Regole di conteggio dei DET.

L'ILF Impiegato comprende:

- ID Impiegato
- Nome Impiegato
- Indirizzo Postale Impiegato
- Livello Retributivo Impiegato
- Qualifica Impiegato
- Data Minima Pensionamento
- Livello di Autorizzazione di Sicurezza Organizzativa

Di seguito è illustrata l'analisi per la definizione dei DET per l'ILF Impiegato nell'applicazione Sicurezza.

Regole di conteggio dei DET	Si applica la regola?
1. Contare un DET per ciascun campo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di un qualsiasi processo elementare nell'ambito del conteggio.	I seguenti attributi soddisfano questa regola: <ul style="list-style-type: none"> <li>• ID Impiegato</li> <li>• Nome Impiegato</li> <li>• Livello di Autorizzazione di Sicurezza Organizzativa</li> </ul>
2. Quando due applicazioni mantengono e/o referenziano la stessa funzione di tipo dati, contare solo i DET effettivamente utilizzati dall'applicazione misurata.	Soltanto ID Impiegato, Nome e Livello di Autorizzazione di Sicurezza Organizzativa sono utilizzati all'interno dell'applicazione Sicurezza.
3. Contare un DET per ogni singolo dato richiesto dall'utente per stabilire una relazione con un'altra funzione di tipo dati.	Non ci sono attributi di questo tipo.
4. Riesaminare gli attributi correlati per determinare se questi sono raggruppati e conteggiati come un singolo DET oppure se sono conteggiati come DET multipli; il raggruppamento dipenderà da come i processi elementari utilizzano gli attributi nell'ambito dell'applicazione.	Non ci sono attributi di questo tipo.

**Passo 4 Contare i RET (Applicazione Sicurezza)**

**Per i RET**, identificare i sottogruppi in base alle regole di conteggio dei RET.

Regole di conteggio dei RET	Si applica la regola?
1. Conta un RET per ogni funzione di tipo dati (cioè, di default ogni funzione di tipo dati ha un sottogruppo di DET che deve essere conteggiato come un RET).	Contare un RET per l'ILF Impiegato.
2. Conta un RET aggiuntivo per ognuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contiene più di un DET: <ul style="list-style-type: none"> <li>entità associativa con attributi non-chiave,</li> <li>entità sotto-tipo (ulteriore rispetto al primo dei sotto-tipi), e</li> <li>entità attributiva, in una relazione 1-1 non obbligatoria.</li> </ul>	<p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p>

**I totali dei RET e dei DET** per l'ILF Impiegato nell'applicazione Sicurezza sono riportati nella seguente tabella.

RET	DET
<ul style="list-style-type: none"> <li>Impiegato</li> </ul>	<ul style="list-style-type: none"> <li>ID Impiegato</li> <li>Nome Impiegato</li> <li>Livello di Autorizzazione di Sicurezza Organizzativa</li> </ul>
<b>Totale</b> 1 RET	<b>Totale</b> 3 DET

**Passo 5 Determinare la complessità funzionale (Applicazione Sicurezza)**

1 RET e 3 DET	La complessità è Bassa
---------------	------------------------

**Passo 6 Determinare la dimensione funzionale (Applicazione Sicurezza)**

Dimensione Funzionale di 1 ILF di bassa complessità	7 FP
---	------



## Esempio: Utenti differenti / Viste dati differenti

### Requisiti utente

Le informazioni che devono essere mantenute da parte dell'utente dell'applicazione RU comprendono:

- ID Impiegato
- Nome Impiegato
- Indirizzo Postale Impiegato

L'Indirizzo Postale Impiegato mantenuto nell'archivio Impiegati comprende Piano, Codice Edificio, Via, Città, Stato e CAP; comunque, nell'applicazione RU, Indirizzo Postale è utilizzato come un singolo attributo.

- Livello Retributivo Impiegato
- Qualifica Impiegato
- Data Minima Pensionamento (\*)

(\*) Alla creazione di un record per un nuovo impiegato, dovrebbe essere calcolata automaticamente e salvata, insieme alle altre informazioni dell'impiegato, la Data Minima Pensionamento.

L'utente RU richiede la possibilità di produrre etichette postali per ogni impiegato.

L'utente dell'applicazione Distribuzione Posta richiede la possibilità di aggiornare per ogni impiegato il Codice Edificio, al fine di riportare le modifiche per i codici ammessi.

L'utente dell'applicazione Distribuzione Posta richiede inoltre la possibilità di conoscere per ogni ubicazione il numero di persone, al fine di determinare il processo più efficiente per la distribuzione della posta interna. È prodotto un report contenente il numero di impiegati presenti in ogni piano per ogni edificio.

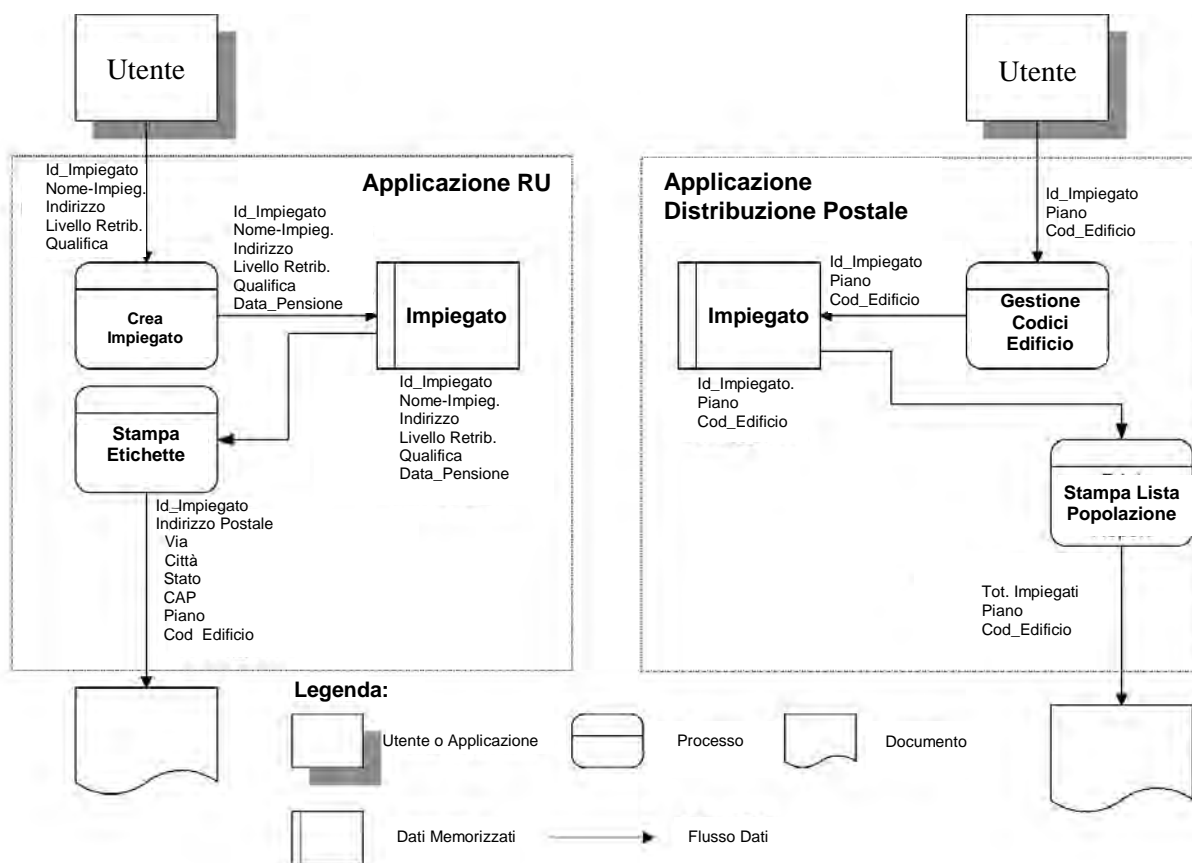
Le informazioni che devono essere mantenute o referenziate dall'utente di Distribuzione Posta comprendono:

- ID Impiegato;
- Piano;
- Codice Edificio.

Esistono altri attributi (per esempio, Livello di Autorizzazione di Sicurezza Organizzativa) nell'entità Impiegato, ma questi non sono referenziati o mantenuti né dall'applicazione RU, né dall'applicazione Distribuzione Posta.

**Diagramma  
flusso dati**

Il seguente diagramma mostra il flusso dei dati per questo esempio.

**Passo 1****Identificare le funzioni di tipo dati (Applicazione RU)**

Determinare se le informazioni dell'impiegato sono una funzione di tipo dati per l'applicazione RU. La seguente tabella illustra l'analisi sintetica.

Regole di identificazione delle funzioni di tipo dati	Si applica la regola?
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	Impiegato.
2. Escludere entità non mantenute da alcuna applicazione.	Non ci sono entità di questo tipo.
3. Raggruppare entità collegate dipendenti.	Non ci sono entità di questo tipo.
4. Escludere entità referenziate come dati di decodifica.	Non ci sono entità di questo tipo.

5. Escludere entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non ci sono entità di questo tipo.

**Passo 2****Classificare le funzioni di tipo dati (Applicazione RU)**

Determinare se Impiegato è classificato come un ILF per l'applicazione RU.

Regole di classificazione delle funzioni di tipo dati	Si applica la regola?
1. Classificare come ILF, se il gruppo di dati è mantenuto dall'applicazione che si sta misurando.	La funzione di tipo dati Impiegato è mantenuta all'interno dell'applicazione RU.
2. Classificare come un EIF, se il gruppo di dati: <ul style="list-style-type: none"> <li>• è referenziato, ma non mantenuto, dall'applicazione che si sta misurando, e</li> <li>• è identificata come ILF in una o più ulteriori applicazioni.</li> </ul>	È classificata come ILF; di conseguenza, non sono identificati EIF.

L'analisi indica che le informazioni dell'Impiegato costituiscono un ILF per l'applicazione RU.

**Passo 3****Contare i DET (Applicazione RU)**

**Per i DET**, esaminare ciascun attributo associato all'ILF Impiegato nell'applicazione RU e determinare se si applicano le Regole di conteggio dei DET.

L'ILF Impiegato comprende:

- ID Impiegato
- Nome Impiegato
- Indirizzo Postale Impiegato (Piano, Codice Edificio, Via, Città, Stato e CAP)
- Livello Retributivo Impiegato
- Qualifica Impiegato
- Data Minima Pensionamento
- Livello di Autorizzazione di Sicurezza Organizzativa

Di seguito è illustrata l'analisi per la definizione dei DET per l'ILF Impiegato nell'applicazione RU.

Regole di conteggio dei DET	Si applica la regola?
1. Contare un DET per ciascun campo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di un qualsiasi processo elementare nell'ambito del conteggio.	I seguenti attributi soddisfano questa regola: <ul style="list-style-type: none"> <li>• ID Impiegato</li> <li>• Nome Impiegato</li> <li>• Indirizzo Postale Impiegato</li> <li>• Livello Retributivo Impiegato</li> <li>• Qualifica Impiegato</li> <li>• Data Minima Pensionamento</li> <li>• Livello di Autorizzazione di Sicurezza Organizzativa</li> </ul>
2. Quando due applicazioni mantengono e/o referenziano la stessa funzione di tipo dati, contare solo i DET effettivamente utilizzati dall'applicazione misurata.	Soltanto ID Impiegato, Nome Impiegato, Indirizzo Postale Impiegato, Livello Retributivo Impiegato, Qualifica Impiegato e Data Minima Pensionamento sono utilizzati all'interno dell'applicazione RU. L'attributo Livello di Autorizzazione di Sicurezza Organizzativa non è contato come un DET perchè non è utilizzato all'interno dell'applicazione RU.
3. Contare un DET per ogni singolo dato richiesto dall'utente per stabilire una relazione con un'altra funzione di tipo dati.	Non ci sono attributi di questo tipo.
4. Riesaminare gli attributi correlati per determinare se questi sono raggruppati e conteggiati come un singolo DET oppure se sono conteggiati come DET multipli; il raggruppamento dipenderà da come i processi elementari utilizzano gli attributi nell'ambito dell'applicazione.	Indirizzo Postale Impiegato è contato come un singolo DET.

**Passo 4 Contare i RET (Applicazione RU)**

Per i RET, identificare i sottogruppi in base alle regole di conteggio dei RET.

Regole di conteggio dei RET	Si applica la regola?
1. Conta un RET per ogni funzione di tipo dati (cioè, di default ogni funzione di tipo dati ha un sottogruppo di DET che deve essere conteggiato come un RET).	Contare un RET per l'ILF Impiegato.
2. Conta un RET aggiuntivo per ognuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contiene più di un DET: <ul style="list-style-type: none"> <li>entità associativa con attributi non-chiave,</li> <li>entità sotto-tipo (ulteriore rispetto al primo dei sotto-tipi), e</li> <li>entità attributiva, in una relazione 1-1 non obbligatoria.</li> </ul>	<p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p>

I totali dei RET e dei DET per l'ILF Impiegato nell'applicazione RU sono riportati nella seguente tabella.

RET	DET
<ul style="list-style-type: none"> <li>Impiegato</li> </ul>	<ul style="list-style-type: none"> <li>ID Impiegato</li> <li>Nome Impiegato</li> <li>Indirizzo Postale Impiegato</li> <li>Livello Retributivo Impiegato</li> <li>Qualifica Impiegato</li> <li>Data Minima Pensionamento</li> </ul>
<b>Totale</b> 1 RET	<b>Totale</b> 6 DET

**Passo 5 Determinare la complessità funzionale (Applicazione RU)**

1 RET e 6 DET	La complessità è Bassa
---------------	------------------------

**Passo 6 Determinare la dimensione funzionale (Applicazione RU)**

Dimensione Funzionale di 1 ILF di complessità bassa	7 FP
---	------

**Passo 1**      **Identificare le funzioni di tipo dati (Applicazione Distribuzione Posta)**

Determinare se le informazioni dell'Impiegato sono una funzione di tipo dati per l'applicazione Distribuzione Posta. La seguente tabella illustra l'analisi.

Regole di identificazione delle funzioni di tipo dati	Si applica la regola?
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente all'interno dell'ambito di conteggio.	Impiegato.
2. Considerare le entità indipendenti come gruppi di dati logici separati.	Non ci sono entità di questo tipo.
3. Raggruppare in una funzione di tipo dati le entità collegate che sono entità dipendenti.	Non ci sono entità di questo tipo.
4. Escludere quelle entità che sono denominate come dati di decodifica.	Non ci sono entità di questo tipo.
5. Escludere le entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Eliminare le entità associative che contengono attributi aggiuntivi non richiesti dall'utente e le entità associative che contengono soltanto le chiavi esterne; gruppi di chiavi esterne con le entità primarie.	Non ci sono entità di questo tipo.

**Passo 2**      **Classificare le funzioni di tipo dati (Applicazione Distribuzione Posta)**

Determinare se Impiegato è classificato come un ILF per l'applicazione Distribuzione Posta.

Regole di classificazione delle funzioni di tipo dati	Si applica la regola?
1. Classificare come un ILF, se i dati sono mantenuti dall'applicazione che si sta misurando.	La funzione di tipo dati Impiegato è mantenuta all'interno dell'applicazione Distribuzione Posta.
2. Classificare come un EIF, se essa: <ul style="list-style-type: none"> <li>• è referenziata, ma non mantenuta, dall'applicazione che si sta misurando e</li> <li>• è identificata come un ILF in una o più altre applicazioni.</li> </ul>	È classificata come un ILF; di conseguenza, non sono identificati EIF.

L'analisi indica che le informazioni dell'Impiegato costituiscono un ILF per l'applicazione Distribuzione Posta.

**Passo 3****Contare i DET (Applicazione Distribuzione Posta)**

**Per i DET**, esaminare ciascun attributo associato all'ILF Impiegato nell'applicazione Distribuzione Posta e determinare se si applicano le Regole di conteggio dei DET.

L'ILF Impiegato comprende:

- ID Impiegato
- Nome Impiegato
- Indirizzo Postale Impiegato (Piano, Codice Edificio, Via, Città, Stato e CAP; nell'applicazione Distribuzione Posta, Piano e Codice Edificio sono utilizzati come attributi separati)
- Livello Retributivo Impiegato
- Qualifica Impiegato
- Data Minima Pensionamento
- Livello di Autorizzazione di Sicurezza Organizzativa

Di seguito è illustrata l'analisi per la definizione dei DET per l'ILF Impiegato nell'applicazione RU.

Regole di conteggio dei DET	Si applica la regola?
1. Contare un DET per ciascun campo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di un qualsiasi processo elementare nell'ambito del conteggio.	I seguenti attributi soddisfano questa regola: <ul style="list-style-type: none"> <li>• ID Impiegato</li> <li>• Piano</li> <li>• Codice Edificio</li> </ul>
2. Quando due applicazioni mantengono e/o referenziano la stessa funzione di tipo dati, contare solo i DET effettivamente utilizzati dall'applicazione misurata.	Soltanto ID Impiegato, Piano e Codice Edificio sono utilizzati dall'applicazione Distribuzione Posta.
3. Contare un DET per ogni singolo dato richiesto dall'utente per stabilire una relazione con un'altra funzione di tipo dati.	Non ci sono attributi di questo tipo.
4. Riesaminare gli attributi correlati per determinare se questi sono raggruppati e conteggiati come un singolo DET oppure se sono conteggiati come DET multipli; il raggruppamento dipenderà da come i processi elementari utilizzano gli attributi nell'ambito dell'applicazione.	Non ci sono attributi di questo tipo. Anche se Indirizzo Postale Impiegato è stato considerato un singolo attributo nell'applicazione RU, i due attributi separati (Piano e Codice Edificio) sono contati come DET separati nell'applicazione Distribuzione Posta.

**Passo 4****Contare i RET (Applicazione Distribuzione Posta)**

Per i RET, identificare i sottogruppi in base alle regole di conteggio dei RET.

Regole di conteggio dei RET	Si applica la regola?
1. Conta un RET per ogni funzione di tipo dati (cioè, di default ogni funzione di tipo dati ha un sottogruppo di DET che deve essere conteggiato come un RET).	Contare un RET per l'ILF Impiegato.
2. Conta un RET aggiuntivo per ognuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contiene più di un DET: <ul style="list-style-type: none"> <li>entità associativa con attributi non-chiave,</li> <li>entità sotto-tipo (ulteriore rispetto al primo dei sotto-tipi), e</li> <li>entità attributiva, in una relazione 1-1 non obbligatoria.</li> </ul>	<p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p>

I totali dei RET e dei DET per l'ILF Impiegato nell'applicazione Distribuzione Posta sono riportati nella seguente tabella.

RET	DET
<ul style="list-style-type: none"> <li>Impiegato</li> </ul>	<ul style="list-style-type: none"> <li>ID Impiegato</li> <li>Piano</li> <li>Codice Edificio</li> </ul>
<b>Totale</b> 1 RET	<b>Totale</b> 3 DET

**Passo 5****Determinare la complessità funzionale (Applicazione Distribuzione Posta)**

1 RET e 3 DET	La complessità è Bassa
---------------	------------------------

**Passo 6****Determinare la dimensione funzionale (Applicazione Distribuzione Posta)**

Dimensione Funzionale di 1 ILF di bassa complessità	7 FP
---	------



---

## EIF: Esempi di conteggio

**Introduzione** Questa sezione utilizza un'applicazione Risorse Umane (RU) insieme con un'applicazione Sicurezza e un'applicazione Pensioni, per illustrare le procedure usate per misurare le funzioni di tipo dati. In aggiunta a questa sezione, vi sono esempi nel Casi di Studio inclusi nella supplementare documentazione IFPUG.

**Contenuti** Questa sezione include i seguenti esempi:

<b>Argomento</b>	<b>Pagina</b>
Descrizione sintetica degli esempi di conteggio degli EIF	1-38
Esempio: Riferimento a dati di altre applicazioni	1-39
Esempio: Riferimento a dati di un'altra applicazione	1-43
Esempio: Fornire dati a altre applicazioni	1-48
Esempio: Applicazione di help	1-50
Esempio: Conversione di dati	1-58
Esempio: File di input di transazione	1-60
Esempio: Utenti differenti / Viste dati differenti	1-62
Esempio: Utilizzi multipli di dati	1-66

## Descrizione sintetica degli esempi di conteggio di EIF

Gli esempi per gli EIF sono descritti nella seguente tabella.

<b>Esempio</b>	<b>Descrizione Sintetica</b>	<b>Pagina</b>
Riferimento a dati di altre applicazioni (per produrre degli output)	Questo esempio identifica gli EIF per una applicazione che fa riferimento ai dati mantenuti da un'altra applicazione. I dati sono usati per produrre un output esterno.	1-39
Riferimento a dati di un'altra applicazione (da usare come parte di un processo di input)	Anche questo esempio considera il riferimento a dati di un'altra applicazione. Questo esempio identifica gli EIF per una applicazione che fa riferimento ai dati mantenuti da un'altra applicazione per usarli come input esterno.	1-43
Fornire dati a altre applicazioni	Questo è un altro esempio di conteggio dei dati che sono referenziati da una diversa applicazione.	1-48
Applicazione di Help	Questo è un esempio di conteggio della funzionalità di help all'interno dell'applicazione RU.	1-50
Conversione di dati	Questo è un esempio di conteggio quando si esegue una conversione verso una nuova applicazione.	1-58
File di input di transazione	Questo esempio applica le regole per il conteggio degli EIF ad un file di input di transazione che viene elaborato per aggiungere lavori all'interno dell'applicazione RU.	1-60
Utenti differenti / Viste dati differenti	Questo esempio mostra come il punto di vista è diverso quando un EIF è usato da molteplici applicazioni.	1-62
Utilizzi multipli di dati	Questo esempio illustra utilizzi multipli per gli stessi dati.	1-66

## Esempio: Riferimento a dati di altre applicazioni

### Requisiti utente

L'utente vuole che l'applicazione Risorse Umane fornisca la possibilità di:

1. Inserire, interrogare e produrre report sulle informazioni dell'impiegato
2. Interfacciare l'applicazione Beni Patrimoniali per reperire le informazioni sulla sede dell'impiegato, da utilizzare per i report. Le informazioni relative alla sede includono nome e descrizione.

### Passo 1

#### Identificare le funzioni di tipo dati

In base ai requisiti utente si hanno due gruppi di informazioni:

- Informazioni sull'Impiegato
- Informazioni sulla Sede

La seguente tabella illustra l'analisi sintetica per valutare se le informazioni sull'Impiegato sono una funzione di tipo dati.

Regole di identificazione delle funzioni di tipo dati	Si applica la regola?
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	Impiegato e Sede.
2. Escludere entità non mantenute da alcuna applicazione.	Non ci sono entità di questo tipo.
3. Raggruppare entità collegate dipendenti.	Non ci sono entità di questo tipo. Impiegato e Sede sono entità indipendenti tra loro.
4. Escludere entità referenziate come dati di decodifica.	Non ci sono entità di questo tipo.
5. Escludere entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non ci sono entità di questo tipo.

Sulla base dell'analisi, Impiegato e Sede sono identificate ciascuna come una funzione di tipo dati.

**Passo 2 Classificare le funzioni di tipo dati (Impiegato)**

Determinare se Impiegato è classificato come un EIF per l'applicazione RU.

Regole di classificazione delle funzioni di tipo dati	Si applica la regola?
1. Classificare come ILF, se il gruppo di dati è mantenuto dall'applicazione che si sta misurando.	Impiegato è mantenuto all'interno dell'applicazione.
2. Classificare come un EIF, se il gruppo di dati: <ul style="list-style-type: none"> <li>• è referenziato, ma non mantenuto, dall'applicazione che si sta misurando, e</li> <li>• è identificata come ILF in una o più ulteriori applicazioni.</li> </ul>	È classificato come ILF; di conseguenza, non sono identificati EIF.

Sulla base dell'analisi, le informazioni sull'Impiegato non sono esterne all'applicazione RU. Esse sono mantenute all'interno, perciò non sono un EIF.

**Passo 2 Classificare le funzioni di tipo dati (Sede)**

Determinare se Sede è classificato come un EIF per l'applicazione RU.

Regole di classificazione delle funzioni di tipo dati	Si applica la regola?
1. Classificare come un ILF, se i dati sono mantenuti dall'applicazione che si sta misurando.	La funzione di tipo dati Sede non è mantenuta all'interno dell'applicazione RU.
2. Classificare come un EIF, se essa : <ul style="list-style-type: none"> <li>• è referenziata, ma non mantenuta, dall'applicazione che si sta misurando e</li> <li>• è identificata come un ILF in una o più altre applicazioni.</li> </ul>	<p>La funzione di tipo dati Sede è referenziata, ma non mantenuta, dall'applicazione RU per produrre i report sugli impiegati.</p> <p>Inizialmente non è chiaro se le informazioni sulla Sede sono mantenute da un'altra applicazione. Dopo aver chiesto agli utenti, si apprende che essi immettono le informazioni usando una schermata dell'applicazione Beni Patrimoniali. Perciò, Sede è un ILF per l'applicazione Beni Patrimoniali e un EIF per l'applicazione RU.</p>

Sulla base dell'analisi, le informazioni sulla Sede sono classificate come un EIF per l'applicazione RU.

### Passo 3 Contare i DET (Sede)

**Per i DET**, esaminare ciascun attributo associato all'EIF Sede e determinare se si applicano le Regole di conteggio dei DET.

L'EIF Sede comprende i seguenti attributi:

- Codice Edificio
- Nome Edificio
- Descrizione Edificio
  - Linea 1
  - Linea 2
  - Linea 3
- Città
- Stato
- Nazione

La seguente tabella illustra l'analisi sintetica per il conteggio dei DET.

Regole di conteggio dei DET	Si applica la regola?
1. Contare un DET per ciascun campo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di un qualsiasi processo elementare nell'ambito del conteggio.	Codice Edificio, Nome Edificio, Descrizione Edificio, Città, Stato e Nazione. Le linee ripetute della Descrizione Edificio sono contate come un solo DET.
2. Quando due applicazioni mantengono e/o referenziano la stessa funzione di tipo dati, contare solo i DET effettivamente utilizzati dall'applicazione misurata.	Non ci sono attributi di questo tipo.
3. Contare un DET per ogni singolo dato richiesto dall'utente per stabilire una relazione con un'altra funzione di tipo dati.	Non ci sono attributi di questo tipo.
4. Riesaminare gli attributi correlati per determinare se questi sono raggruppati e conteggiati come un singolo DET oppure se sono conteggiati come DET multipli; il raggruppamento dipenderà da come i processi elementari utilizzano gli attributi nell'ambito dell'applicazione.	Non ci sono attributi di questo tipo.

**Passo 4****Contare i RET (Sede)**

Per i RET, identificare i sottogruppi in base alle regole di conteggio dei RET.

Regole di conteggio dei RET	Si applica la regola?
1. Conta un RET per ogni funzione di tipo dati (cioè, di default ogni funzione di tipo dati ha un sottogruppo di DET che deve essere conteggiato come un RET).	Conta un RET per l'EIF Sede.
2. Conta un RET aggiuntivo per ognuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contiene più di un DET: <ul style="list-style-type: none"> <li>entità associativa con attributi non-chiave,</li> <li>entità sotto-tipo (ulteriore rispetto al primo dei sotto-tipi), e</li> <li>entità attributiva, in una relazione 1-1 non obbligatoria.</li> </ul>	<p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p>

I totali dei RET e dei DET per l'EIF Sede sono riportati nella seguente tabella .

RET	DET
<ul style="list-style-type: none"> <li>Sede</li> </ul>	<ul style="list-style-type: none"> <li>Codice Edificio</li> <li>Nome Edificio</li> <li>Descrizione Edificio (linee ripetute)</li> <li>Città</li> <li>Stato</li> <li>Nazione</li> </ul>
<b>Totale</b> 1 RET	<b>Totale</b> 6 DET

**Passo 5****Determinare la complessità funzionale**

1 RET e 6 DET	La complessità è Bassa
---------------	------------------------

**Passo 6****Determinare la dimensione funzionale**

Dimensione Funzionale di 1 EIF di bassa complessità	5 FP
---	------

## Esempio: Riferimento a dati di un'altra applicazione

### Requisiti utente

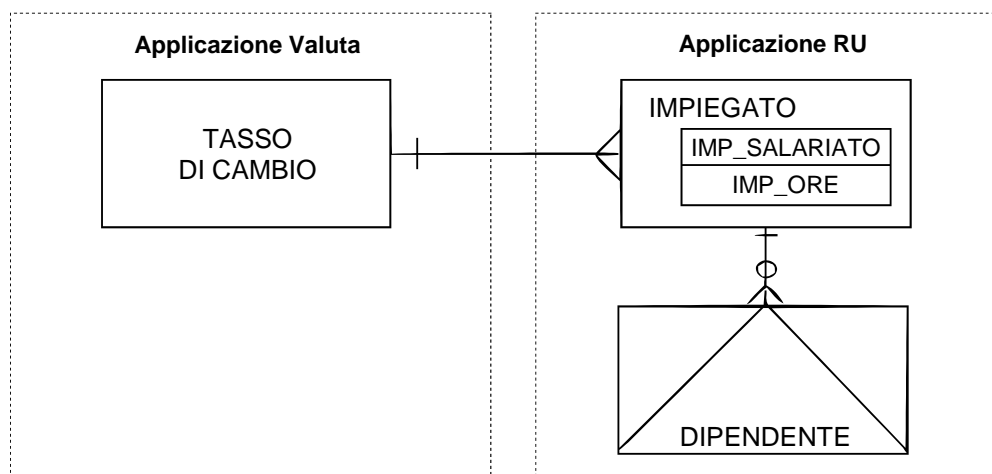
L'utente richiede che l'applicazione Risorse Umane fornisca le seguenti possibilità:

- Tutti gli impiegati ad ore devono essere pagati in dollari statunitensi;
- Quando l'utente aggiunge o modifica le informazioni sull'impiegato, l'applicazione Risorse Umane deve accedere all'applicazione Valuta per reperire il tasso di cambio. Dopo aver reperito il tasso di cambio, l'applicazione RU converte la paga oraria standard locale dell'impiegato in paga oraria in dollari U.S. con il seguente calcolo:






$$\frac{\text{Paga Oraria Standard}}{\text{Tasso di Cambio}} = \text{Paga Oraria in Dollari U.S.}$$

### Modello dei Dati

Il seguente diagramma illustra le relazioni per questo esempio.



#### Legenda:

-  Entità
-  Entità Attributiva
-  Sottotipo di Entità
-  Relazione Obbligatoria 1 : N
-  Relazione Opzionale 1 : N

Le informazioni sul cambio di valuta includono:

VALUTA

Tasso\_di\_Cambio\_alla\_Valuta\_Base

Valuta

**Passo 1**      **Identificare le funzioni di tipo dati**

In base ai requisiti utente, ci sono due gruppi di informazioni:

- Informazioni sul cambio di valuta
- Informazioni sull'impiegato

La seguente tabella illustra l'analisi sintetica per valutare se le informazioni sul cambio di valuta sono una funzione di tipo dati.

<b>Regole di identificazione delle funzioni di tipo dati</b>	<b>Si applica la regola?</b>
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	Cambio di Valuta, Impiegato e Dipendente.
2. Escludere entità non mantenute da alcuna applicazione.	Non ci sono entità di questo tipo.
3. Raggruppare entità collegate dipendenti.	Cambio di Valuta è una entità indipendente dalle altre. Dipendente è un'entità dipendente da Impiegato.
4. Escludere entità referenziate come dati di decodifica.	Anche se Cambio di Valuta sembra essere una istanza di dati di decodifica, Valuta e Tasso di Conversione non sono sostituibili con altri (cioè, non sono intercambiabili). Inoltre le informazioni sul Cambio di Valuta cambiano regolarmente, così non soddisfano il criterio di essere essenzialmente statiche.
5. Escludere entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non ci sono entità di questo tipo.

Sulla base dell'analisi, Cambio di Valuta e Impiegato sono identificate ciascuna come una funzione di tipo dati. Dipendente non è una funzione di tipo dati da sé stante, ma è parte della funzione di tipo dati Impiegato.



**Passo 2 Classificare le funzioni di tipo dati (Impiegato)**

La seguente tabella mostra l'analisi per determinare se le informazioni di Impiegato sono classificate come un EIF.

Regole di classificazione delle funzioni di tipo dati	Si applica la regola?
1. Classificare come ILF, se il gruppo di dati è mantenuto dall'applicazione che si sta misurando.	Le informazioni su Impiegato sono mantenute all'interno dell'applicazione RU.
2. Classificare come un EIF, se il gruppo di dati: <ul style="list-style-type: none"> <li>• è referenziato, ma non mantenuto, dall'applicazione che si sta misurando, e</li> <li>• è identificata come ILF in una o più ulteriori applicazioni.</li> </ul>	È classificata come ILF; di conseguenza, non sono identificati EIF.

Sulla base dell'analisi, le informazioni sull'Impiegato non sono esterne all'applicazione RU. Esse sono mantenute all'interno, perciò non sono un EIF.

**Passo 2 Classificare le funzioni di tipo dati (Cambio di Valuta)**

La seguente tabella mostra l'analisi per determinare se le informazioni di Cambio di Valuta sono classificate come un EIF.

Regole di classificazione delle funzioni di tipo dati	Si applica la regola?
1. Classificare come ILF, se il gruppo di dati è mantenuto dall'applicazione che si sta misurando.	Cambio di Valuta non è mantenuta all'interno dell'applicazione RU.
2. Classificare come un EIF, se il gruppo di dati: <ul style="list-style-type: none"> <li>• è referenziato, ma non mantenuto, dall'applicazione che si sta misurando, e</li> <li>• è identificata come ILF in una o più ulteriori applicazioni.</li> </ul>	<p>La funzione di tipo dati Cambio di Valuta è referenziata dall'applicazione RU per effettuare i calcoli della retribuzione di un dipendente.</p> <p>Anche se Cambio di Valuta sembra essere una istanza di dati di decodifica, Valuta e Tasso di Conversione non sono sostituibili con altri (cioè, non sono intercambiabili). Inoltre le informazioni sul Cambio di Valuta cambiano regolarmente, così non soddisfano il criterio di essere essenzialmente statiche.</p>

Poichè l'applicazione Valuta fornisce il tasso di conversione per l'applicazione RU, il gruppo di dati Cambio di Valuta è un EIF per l'applicazione RU.

**Passo 3****Contare i RET (Cambio di Valuta)**

**Per i DET**, esaminare ciascun attributo associato all'EIF Cambio di Valuta e determinare se si applicano le Regole di conteggio dei DET. La seguente tabella illustra l'analisi sintetica per il conteggio dei DET.

Regole di conteggio dei DET	Si applica la regola?
1. Contare un DET per ciascun campo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di un qualsiasi processo elementare nell'ambito del conteggio.	Tasso di Conversione, Valuta.
2. Quando due applicazioni mantengono e/o referenziano la stessa funzione di tipo dati, contare solo i DET effettivamente utilizzati dall'applicazione misurata.	Tutti gli attributi sono referenziati dall'applicazione RU.
3. Contare un DET per ogni singolo dato richiesto dall'utente per stabilire una relazione con un'altra funzione di tipo dati.	Non ci sono attributi di questo tipo.
4. Riesaminare gli attributi correlati per determinare se questi sono raggruppati e conteggiati come un singolo DET oppure se sono conteggiati come DET multipli; il raggruppamento dipenderà da come i processi elementari utilizzano gli attributi nell'ambito dell'applicazione.	Non ci sono attributi di questo tipo.

**Passo 4 Contare i RET (Cambio di Valuta)**

Per i RET, identificare i sottogruppi in base alle regole di conteggio dei RET.

Regole di conteggio dei RET	Si applica la regola?
1. Conta un RET per ogni funzione di tipo dati (cioè, di default ogni funzione di tipo dati ha un sottogruppo di DET che deve essere conteggiato come un RET).	Contare un RET per l'EIF Cambio di Valuta.
2. Conta un RET aggiuntivo per ognuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contiene più di un DET: <ul style="list-style-type: none"> <li>entità associativa con attributi non-chiave,</li> <li>entità sotto-tipo (ulteriore rispetto al primo dei sotto-tipi), e</li> <li>entità attributiva, in una relazione 1-1 non obbligatoria.</li> </ul>	<p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo</p>

I totali dei RET e dei DET per l'EIF Cambio di Valuta sono riportati nella seguente tabella.

RET	DET
• Cambio di Valuta	• Tasso di Conversione • Valuta
<b>Totale</b> 1 RET	<b>Totale</b> 2 DET

**Passo 5 Determinare la complessità funzionale (Cambio di Valuta)**

1 RET e 2 DET	La complessità è Bassa
---------------	------------------------

**Passo 6 Determinare la dimensione funzionale**

Dimensione Funzionale di 1 EIF di bassa complessità	5 FP
---	------

## Esempio: Fornire dati ad altre applicazioni

### Requisiti utente

L'utente ha fissato i seguenti requisiti per l'applicazione Valuta:

- Gestire il tasso di cambio in dollari U.S. per altre valute
- Fornire una interfaccia che permetta ad altre applicazioni, come Risorse Umane, di reperire informazioni sui tassi di cambio.

### Passo 1

#### Identificare le funzioni di tipo dati

Per questo esempio, determinare se le informazioni sul Cambio di Valuta sono una funzione di tipo dati. La seguente tabella illustra l'analisi sintetica.

Regole di identificazione delle funzioni di tipo dati	Si applica la regola?
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	Cambio di Valuta.
2. Escludere entità non mantenute da alcuna applicazione.	Non ci sono entità di questo tipo.
3. Raggruppare entità collegate dipendenti.	Non ci sono entità di questo tipo.
4. Escludere entità referenziate come dati di decodifica.	Anche se Cambio di Valuta sembra essere una istanza di dati di decodifica, Valuta e Tasso di Conversione non sono sostituibili con altri (cioè, non sono intercambiabili). Inoltre le informazioni sul Cambio di Valuta cambiano regolarmente, così non soddisfano il criterio di essere essenzialmente statiche.
5. Escludere entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non ci sono entità di questo tipo.

**Passo 2****Classificare le funzioni di tipo dati (Cambio di Valuta)**

La seguente tabella mostra l'analisi per determinare se le informazioni di Cambio di Valuta sono classificate come un EIF.

<b>Regole di classificazione delle funzioni di tipo dati</b>	<b>Si applica la regola?</b>
1. Classificare come ILF, se il gruppo di dati è mantenuto dall'applicazione che si sta misurando.	L'applicazione Valuta mantiene i dati di Cambio di Valuta con una transazione via Internet.
2. Classificare come un EIF, se il gruppo di dati: <ul style="list-style-type: none"><li>• è referenziato, ma non mantenuto, dall'applicazione che si sta misurando, e</li><li>• è identificata come ILF in una o più ulteriori applicazioni.</li></ul>	È classificata come un ILF; di conseguenza, non sono identificati EIF.

Le informazioni di Cambio di Valuta non sono esterne all'applicazione Valuta, perciò non sono contate come un EIF per tale applicazione; quindi esso è contato come un ILF piuttosto che un EIF per l'applicazione Valuta. Consultare l'esempio precedente in questo capitolo per riesaminare come il riferimento al Cambio di Valuta può essere contato come un EIF.

## Esempio: Applicazione di Help

### Requisiti utente

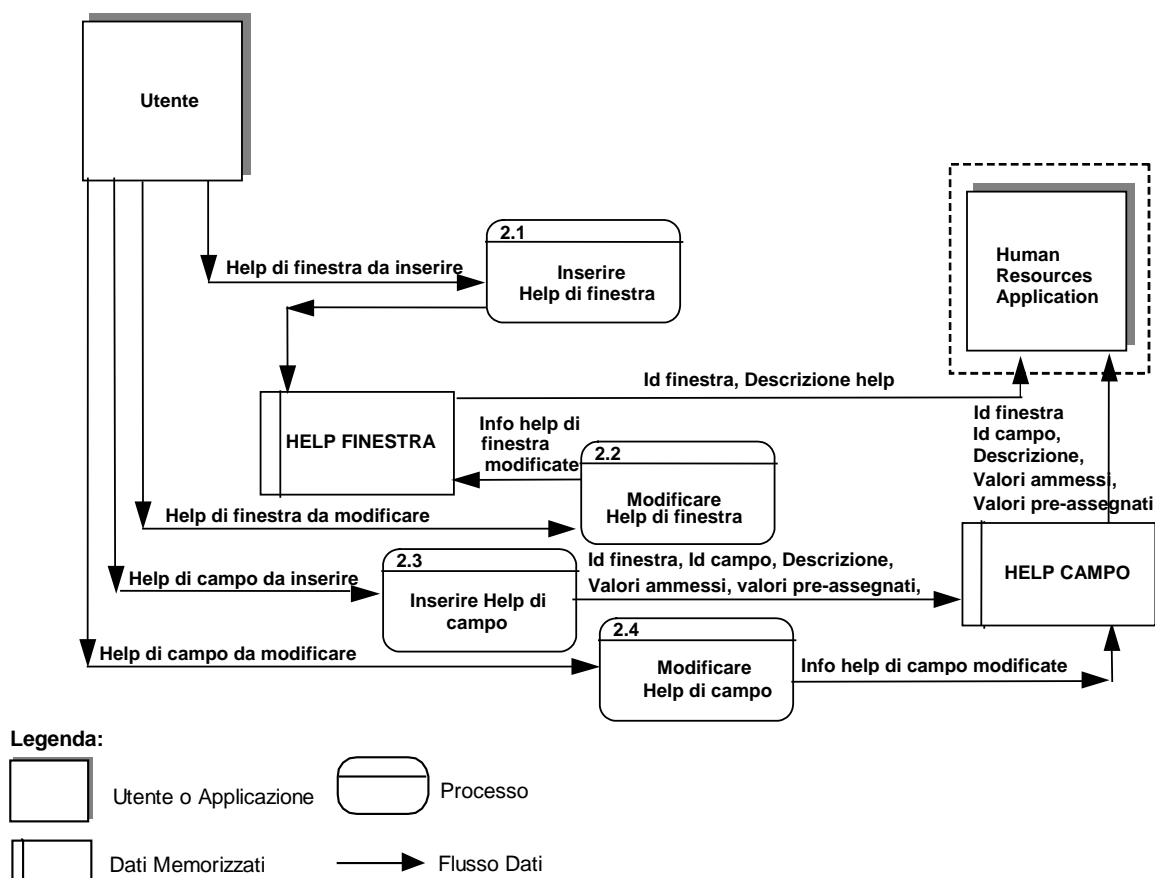
L'utente richiede che l'applicazione di Help fornisca:

1. La possibilità per un utente di descrivere come usare ciascuna finestra per completare ogni funzione applicativa
2. La possibilità di modificare l'help di finestra
3. La possibilità di impostare una definizione, i valori pre-assegnati (default) e i valori ammessi per ciascun attributo nell'applicazione RU
4. La possibilità di modificare l'help di campo
5. La possibilità di reperire e visualizzare l'help di finestra e di campo per l'applicazione RU.

L'help di finestra e l'help di campo sono mantenuti indipendentemente. Una occorrenza può esistere per un tipo di help senza che esista per l'altro.

### Diagramma di flusso

Il seguente diagramma illustra il flusso dei dati per questo esempio.



**Passo 1**      **Identificare le funzioni di tipo dati**

In base ai requisiti dell'applicazione Risorse Umane, si hanno due gruppi di dati:

- Help di Finestra
- Help di Campo

La seguente tabella illustra l'analisi sintetica per valutare se Help di Finestra e Help di Campo sono funzioni di tipo dati per l'applicazione RU.

<b>Regole di identificazione delle funzioni di tipo dati</b>	<b>Si applica la regola?</b>
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	Help di Finestra, Help di Campo.
2. Escludere entità non mantenute da alcuna applicazione.	Non ci sono entità di questo tipo.
3. Raggruppare entità collegate dipendenti.	Help di Finestra e Help di Campo sono indipendenti l'uno dall'altra. Una occorrenza può esistere per un tipo di help senza che esista per l'altro tipo.
4. Escludere entità referenziate come dati di decodifica.	Help di Finestra e Help di Campo consistono in più di un semplice codice e descrizione, non sono sostituibili e sono memorizzati per supportare le attività applicative degli utenti; di conseguenza non sono considerati dei dati di decodifica.
5. Escludere entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non ci sono entità di questo tipo.

**Passo 2**      **Classificare le funzioni di tipo dati (Help di Finestra)**

La seguente tabella mostra l'analisi per determinare se le informazioni dell'Help di Finestra sono classificate come EIF.

Regole di classificazione delle funzioni di tipo dati	Si applica la regola?
1. Classificare come ILF, se il gruppo di dati è mantenuto dall'applicazione che si sta misurando.	Help di Finestra non è mantenuto all'interno dell'applicazione RU.
2. Classificare come un EIF, se il gruppo di dati: <ul style="list-style-type: none"><li>• è referenziato, ma non mantenuto, dall'applicazione che si sta misurando, e</li><li>• è identificata come ILF in una o più ulteriori applicazioni.</li></ul>	<p>L'applicazione RU referencia, ma non mantiene, Help di Finestra.</p> <p>L'applicazione Help ha identificato Help di Finestra come ILF.</p>

Le informazioni Help di Finestra sono un EIF nell'applicazione RU poichè le informazioni sono referenziate da tale applicazione. Le informazioni Help di Finestra sono mantenute nell'applicazione Help, dove sono contate come un ILF.



**Passo 3 Contare i DET (Help di Finestra)**

**Per i DET**, esaminare ciascun attributo associato all'EIF Help di Finestra e determinare se si applicano le Regole di conteggio dei DET. Gli attributi per Help di Finestra comprendono:

- ID Finestra
- Descrizione Funzione Applicativa.

La seguente tabella mostra l'analisi dei DET per l'Help di Finestra.

Regole di conteggio dei DET	Si applica la regola?
1. Contare un DET per ciascun campo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di un qualsiasi processo elementare nell'ambito del conteggio.	ID Finestra, Descrizione Funzione Applicativa.
2. Quando due applicazioni mantengono e/o referenziano la stessa funzione di tipo dati, contare solo i DET effettivamente utilizzati dall'applicazione misurata.	Tutti gli attributi sono referenziati dall'applicazione RU.
3. Contare un DET per ogni singolo dato richiesto dall'utente per stabilire una relazione con un'altra funzione di tipo dati.	Non ci sono attributi di questo tipo.
4. Riesaminare gli attributi correlati per determinare se questi sono raggruppati e conteggiati come un singolo DET oppure se sono conteggiati come DET multipli; il raggruppamento dipenderà da come i processi elementari utilizzano gli attributi nell'ambito dell'applicazione.	Non ci sono attributi di questo tipo.

**Passo 4 Contare i RET (Help di Finestra)**

Per i RET, identificare i sottogruppi in base alle regole di conteggio dei RET.

Regole di conteggio dei RET	Si applica la regola?
1. Conta un RET per ogni funzione di tipo dati (cioè, di default ogni funzione di tipo dati ha un sottogruppo di DET che deve essere conteggiato come un RET).	Contare un RET per Help di Finestra.
2. Conta un RET aggiuntivo per ognuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contiene più di un DET: <ul style="list-style-type: none"> <li>entità associativa con attributi non-chiave,</li> <li>entità sotto-tipo (ulteriore rispetto al primo dei sotto-tipi), e</li> <li>entità attributiva, in una relazione 1-1 non obbligatoria.</li> </ul>	Non ci sono entità di questo tipo.  Non ci sono entità di questo tipo.  Non ci sono entità di questo tipo.

I totali dei RET e dei DET per l'EIF Help di Finestra sono riportati nella seguente tabella .

RET	DET
• Help di Finestra	• ID Finestra • Descrizione Funzione Applicativa
<b>Totale</b> 1 RET	<b>Totale</b> 2 DET

**Passo 5 Determinare la complessità funzionale (Help di Finestra)**

1 RET e 2 DET	La complessità è Bassa
---------------	------------------------

**Passo 6 Determinare la dimensione funzionale (Help di Finestra)**

Dimensione Funzionale di 1 EIF di bassa complessità	5 FP
---	------

**Passo 2**      **Classificare le funzioni di tipo dati (Help di Campo)**

La seguente tabella mostra l'analisi sintetica per determinare se Help di Campo è classificato come un EIF.

Regole di classificazione delle funzioni di tipo dati	Si applica la regola?
1. Classificare come un ILF, se i dati sono mantenuti dall'applicazione che si sta misurando.	Help di Campo non è mantenuto all'interno dell'applicazione RU.
2. Classificare come un EIF, se essa : <ul style="list-style-type: none"><li>• è referenziata, ma non mantenuta, dall'applicazione che si sta misurando e</li><li>• è identificata come un ILF in una o più altre applicazioni.</li></ul>	<p>L'applicazione RU referencia, ma non mantiene, Help di Campo.</p> <p>L'applicazione Help ha identificato Help di Campo come un ILF.</p>

Le informazioni Help di Campo sono un EIF nell'applicazione RU poichè le informazioni sono reperite da tale applicazione. Le informazioni Help di Campo sono mantenute nell'applicazione Help, dove sono contate come un ILF.

**Step 3****Count DET (for Field Help)**

**Per i DET**, esaminare ciascun attributo associato all'EIF Help di Campo e determinare se si applicano le Regole di conteggio dei DET. La seguente lista mostra gli attributi per Help di Campo:

- ID Finestra
- ID Campo
- Descrizione Campo
- Valori Pre-Assegnati
- Valori Ammessi

La seguente tabella mostra l'analisi dei DET per Help di Campo.

Regole di conteggio dei DET	Si applica la regola?
1. Contare un DET per ciascun campo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di un qualsiasi processo elementare nell'ambito del conteggio.	ID Finestra, ID Campo, Descrizione Campo, Valori pre-Assegnati, Valori Ammessi.
2. Quando due applicazioni mantengono e/o referenziano la stessa funzione di tipo dati, contare solo i DET effettivamente utilizzati dall'applicazione misurata.	Tutti gli attributi sono referenziati dall'applicazione RU.
3. Contare un DET per ogni singolo dato richiesto dall'utente per stabilire una relazione con un'altra funzione di tipo dati.	Non ci sono attributi di questo tipo.
4. Riesaminare gli attributi correlati per determinare se questi sono raggruppati e conteggiati come un singolo DET oppure se sono conteggiati come DET multipli; il raggruppamento dipenderà da come i processi elementari utilizzano gli attributi nell'ambito dell'applicazione.	Non ci sono attributi di questo tipo.

**Passo 4 Contare i RET (Help di Campo)**

Per i RET, identificare i sottogruppi in base alle regole di conteggio dei RET.

Regole di conteggio dei RET	Si applica la regola?
1. Conta un RET per ogni funzione di tipo dati (cioè, di default ogni funzione di tipo dati ha un sottogruppo di DET che deve essere conteggiato come un RET).	Contare un RET per Help di Campo.
2. Conta un RET aggiuntivo per ognuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contiene più di un DET: <ul style="list-style-type: none"> <li>entità associativa con attributi non-chiave,</li> <li>entità sotto-tipo (ulteriore rispetto al primo dei sotto-tipi), e</li> <li>entità attributiva, in una relazione 1-1 non obbligatoria.</li> </ul>	<p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p>

I totali dei RET e dei DET per l'EIF Help di Campo sono riportati nella seguente tabella.

RET	DET
<ul style="list-style-type: none"> <li>Help di Campo</li> </ul>	<ul style="list-style-type: none"> <li>ID Finestra</li> <li>ID Campo</li> <li>Descrizione Campo</li> <li>Valori Pre-Assegnati</li> <li>Valori Ammessi</li> </ul>
<b>Totale</b> 1 RET	<b>Totale</b> 5 DET

**Passo 5 Determinare la complessità funzionale (Help di Campo)**

1 RET e 5 DET	La complessità è Bassa
---------------	------------------------

**Passo 6 Determinare la dimensione funzionale (Help di Campo)**

Dimensione Funzionale di 1 EIF di bassa complessità	5 FP
---	------

## Esempio: Conversione di dati

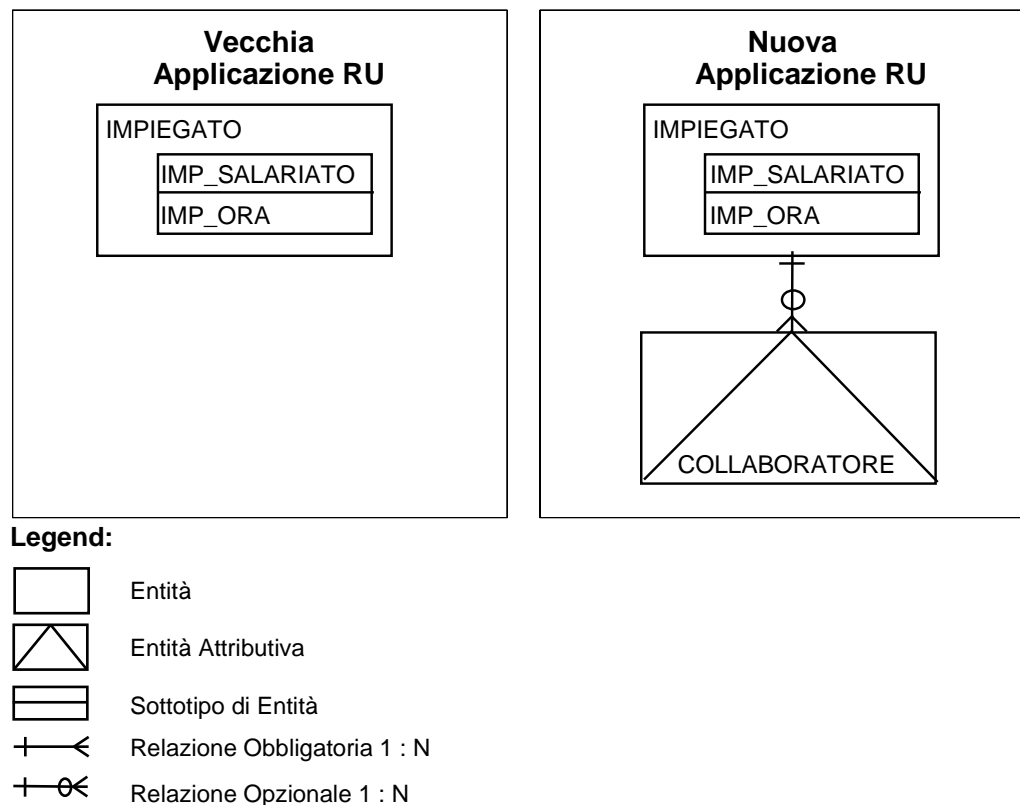
### Requisiti utente

Una organizzazione ha acquistato un pacchetto per una nuova applicazione RU. L'organizzazione ha richiesto la possibilità di convertire il suo file degli impiegati dal Sistema RU pre-esistente al nuovo Sistema RU.

Il precedente sistema non forniva la possibilità di mantenere le informazioni sui collaboratori di un impiegato. Le informazioni sui collaboratori sono inizializzate quando gli impiegati esistenti sono migrati nella nuova applicazione.

### Modello dei Dati

Il seguente diagramma illustra i dati per le due applicazioni.



Il file Impiegato della precedente applicazione RU è usato per aggiungere impiegati alla nuova applicazione RU. Il file Impiegato della precedente applicazione ha l'intento primario di mantenere (per esempio, popolare) il file Impiegato nella nuova applicazione. Il file Impiegato della precedente applicazione non soddisfa l'intento primario di un EIF che è quello di reperire i dati attraverso uno o più processi elementari.

## Passo 1 Identificare le funzioni di tipo dati

In base ai requisiti utente, valutare se il vecchio file Impiegato è una funzione di tipo dati. La seguente tabella illustra l'analisi sintetica.

Regole di identificazione delle funzioni di tipo dati	Si applica la regola?
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	Dal punto di vista dell'utente, il vecchio file Impiegato non è un gruppo logico di dati.
2. Escludere entità non mantenute da alcuna applicazione.	Il vecchio file Impiegato è un output (per esempio, estrazione) della prima applicazione, piuttosto che un file logico mantenuto.
3. Raggruppare entità collegate dipendenti.	Non ci sono entità di questo tipo.
4. Escludere entità referenziate come dati di decodifica.	Non ci sono entità di questo tipo.
5. Escludere entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non ci sono entità di questo tipo.

Il file Impiegato è un file di transazione delle informazioni dell'impiegato che sono migrate nel nuovo sistema. Il processo di conversione usa il file di transazione per mantenere le informazioni sull'impiegato dopo che esso entra nella nuova applicazione.

Il vecchio file Impiegato non è un gruppo logico di dati nella nuova applicazione RU dal punto di vista dell'utente. L'intento primario del vecchio file Impiegato è quello di servire da input per la nuova applicazione RU, non di reperire dati attraverso uno o più processi elementari della nuova applicazione RU; perciò non è un EIF. Fare riferimento agli Esempi sul Conteggio degli EI/EO/EQ per vedere come il vecchio file Impiegato può essere contato come un input esterno.

## Esempio: File di input di transazione

### Requisiti utente

L'utente richiede la possibilità di:

1. Inserire, modificare, cancellare, interrogare e ottenere report sul lavoro in modalità interattiva
2. Inserire e modificare le informazioni sul lavoro in modalità batch.

### Formato Record

Il seguente diagramma riporta per questo esempio il formato del record per inserire e modificare le informazioni sul lavoro in modalità batch.

	123456789101234567890123456789012345678901234567890123456789012345678901234567890
1	ADD 01 SRENG INGEGNERE SENIOR SISTEMI INFORMATIVI  05
2	ADD 02 SRENG 01 PARTE DA LIV.RETRIB.05
3	ADD 02 SRENG 02 ALTRI LIV.RETRIB.:06 E 07
4	CHG 01 STENG   04
5	CHG 02 STENG 02  ALTRI LIV.RETRIB.:05 E 06
6	
7	
8	
9	
0	

### Descrizione dei Record

La seguente tabella include le descrizioni per ciascun tipo record.

Record	Position	Description
01	1-3	Tipo Transazione
	4-5	Tipo Record
	6-10	Codice Lavoro
	11-45	Nome Lavoro
	46-47	Livello Retributivo
02	1-3	Tipo Transazione
	4-5	Tipo Record
	6-10	Codice Lavoro
	11-12	Progressivo Linea Descrizione Lavoro
	13-41	Linee Descrizione Lavoro



**Passo 1**      **Identificare le funzioni di tipo dati**

In base ai requisiti utente, valutare se il file di transazione è una funzione di tipo dati. La seguente tabella illustra l'analisi sintetica.

<b>Regole di identificazione delle funzioni di tipo dati</b>	<b>Si applica la regola?</b>
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	Sì. I dati sono raggruppati in transazioni che entrano nel confine dell'applicazione per mantenere l'ILF Lavoro.
2. Escludere entità non mantenute da alcuna applicazione.	Il File Transazione è escluso. Le transazioni entrando nel confine dell'applicazione per aggiornare l'ILF Lavoro, formano dei processi elementari. Non c'è un processo elementare che aggiorna il file di transazione.
3. Raggruppare entità collegate dipendenti.	Non ci sono entità di questo tipo.
4. Escludere entità referenziate come dati di decodifica.	Non ci sono entità di questo tipo.
5. Escludere entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non ci sono entità di questo tipo.

Non ci sono EIF per questo esempio. Fare riferimento agli Esempi sul Conteggio degli EI/EO/EQ per avere spiegazioni su come un file di input di transazione può essere contato come un input esterno.

## Esempio: Utenti differenti / Viste dati differenti

### Requisiti utente

L'utente dell'applicazione RU richiede la possibilità di mantenere informazioni su ogni nuovo impiegato.

Le informazioni che devono essere mantenute da parte dell'utente dell'applicazione RU comprendono:

- ID Impiegato
- Nome Impiegato
- Indirizzo Postale Impiegato
- Livello Retributivo Impiegato
- Qualifica Impiegato
- Data Minima Pensionamento (\*)

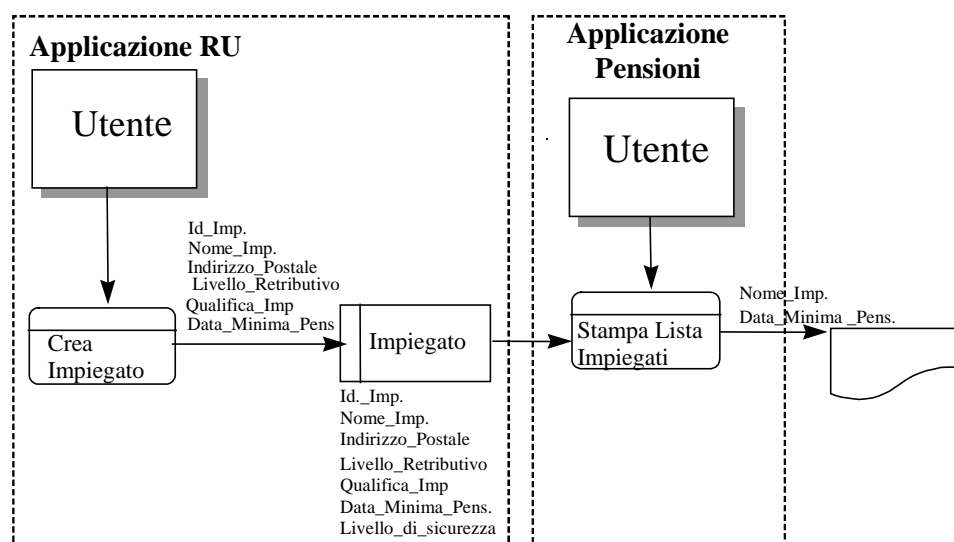
(\*) Alla creazione di un record per un nuovo impiegato, dovrebbe essere calcolata automaticamente e salvata, insieme alle altre informazioni dell'impiegato, la Data Minima di Pensionamento.

L'utente dell'applicazione Pensioni richiede la possibilità di produrre un elenco di tutti gli impiegati con la loro data minima di pensionamento.

Esistono altri attributi (per esempio, Livello di Autorizzazione di Sicurezza Organizzativa) nell'entità Impiegato, ma questi non sono referenziati o mantenuti né dall'applicazione RU né dall'applicazione Distribuzione Posta.

### Diagramma di flusso dei dati

Il seguente diagramma illustra il flusso dei dati per questo esempio.



**Passo 1**      **Identificare le funzioni di tipo dati (Applicazione Pensioni)**

La seguente tabella mostra l'analisi sintetica per determinare se le informazioni sull'impiegato sono una funzione di tipo dati per l'applicazione Pensioni.

<b>Regole di identificazione delle funzioni di tipo dati</b>	<b>Si applica la regola?</b>
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	Impiegato.
2. Escludere entità non mantenute da alcuna applicazione.	Le informazioni sull'Impiegato sono mantenute all'interno dell'applicazione RU. Queste non sono escluse.
3. Raggruppare entità collegate dipendenti.	Non ci sono entità di questo tipo.
4. Escludere entità referenziate come dati di decodifica.	Non ci sono entità di questo tipo.
5. Escludere entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non ci sono entità di questo tipo.

**Passo 2**      **Classificare le funzioni di tipo dati (Applicazione Pensioni)**

La seguente tabella mostra l'analisi sintetica per determinare se Impiegato è classificato come un EIF per l'applicazione Pensioni.

<b>Regole di classificazione delle funzioni di tipo dati</b>	<b>Si applica la regola?</b>
1. Classificare come ILF, se il gruppo di dati è mantenuto dall'applicazione che si sta misurando.	L'applicazione Pensioni non mantiene le informazioni sull'Impiegato.
2. Classificare come un EIF, se il gruppo di dati: <ul style="list-style-type: none"> <li>• è referenziato, ma non mantenuto, dall'applicazione che si sta misurando, e</li> <li>• è identificata come ILF in una o più ulteriori applicazioni.</li> </ul>	<p>L'applicazione Pensioni referencia, ma non mantiene, i dati dell'Impiegato.</p> <p>L'applicazione RU ha identificato Impiegato come ILF.</p>

Le informazioni sull'Impiegato verificano tutti i requisiti di un EIF per l'applicazione Pensioni.

### Passo 3 Contare i DET (Applicazione Pensioni)

**Per i DET**, esaminare ciascun attributo associato all'EIF Impiegato per l'applicazione Pensioni e determinare se si applicano le Regole di conteggio dei DET.

Le informazioni sull'Impiegato comprendono:

- ID Impiegato
- Nome Impiegato
- Indirizzo Postale Impiegato
- Livello Retributivo Impiegato
- Qualifica Impiegato
- Data Minima Pensionamento

La seguente tabella mostra l'analisi dei DET per le informazioni Impiegato dell'applicazione Pensioni.

Regole di conteggio dei DET	Si applica la regola?
1. Contare un DET per ciascun campo unico, riconoscibile dall'utente, non ripetuto, mantenuto nella o reperito dalla funzione di tipo dati attraverso l'esecuzione di un qualsiasi processo elementare nell'ambito del conteggio.	Le informazioni sull'Impiegato comprendono i seguenti attributi: <ul style="list-style-type: none"> <li>• ID Impiegato</li> <li>• Nome Impiegato</li> <li>• Indirizzo Postale Impiegato</li> <li>• Livello Retributivo Impiegato</li> <li>• Qualifica Impiegato</li> <li>• Data minima Pensionamento</li> <li>• Livello di Autorizzazione di Sicurezza Organizzativa</li> </ul>
2. Quando due applicazioni mantengono e/o referenziano la stessa funzione di tipo dati, contare solo i DET effettivamente utilizzati dall'applicazione misurata.	Soltanto Nome Impiegato e Data Minima Pensionamento sono riconosciuti dall'utente dell'applicazione Pensioni. Tutti gli altri non sono contati come DET per l'applicazione Pensioni.
3. Contare un DET per ogni singolo dato richiesto dall'utente per stabilire una relazione con un'altra funzione di tipo dati.	Non ci sono attributi di questo tipo.
4. Riesaminare gli attributi correlati per determinare se questi sono raggruppati e conteggiati come un singolo DET oppure se sono conteggiati come DET multipli; il raggruppamento dipenderà da come i processi elementari utilizzano gli attributi nell'ambito dell'applicazione.	Non ci sono attributi di questo tipo.

**Passo 4 Contare i RET (Applicazione Pensioni)**

Per i RET, identificare i sottogruppi in base alle regole di conteggio dei RET.

Regole di conteggio dei RET	Si applica la regola?
1. Conta un RET per ogni funzione di tipo dati (cioè, di default ogni funzione di tipo dati ha un sottogruppo di DET che deve essere conteggiato come un RET).	Contare un RET per l'EIF Impiegato.
2. Conta un RET aggiuntivo per ognuno dei seguenti ulteriori sottogruppi logici di DET (nella funzione di tipo dati) che contiene più di un DET: <ul style="list-style-type: none"> <li>entità associativa con attributi non-chiave,</li> <li>entità sotto-tipo (ulteriore rispetto al primo dei sotto-tipi), e</li> <li>entità attributiva, in una relazione 1-1 non obbligatoria.</li> </ul>	<p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p> <p>Non ci sono entità di questo tipo.</p>

I totali dei RET e dei DET per l'EIF Impiegato nell'applicazione Pensioni sono riportati nella seguente tabella.

RET	DET
<ul style="list-style-type: none"> <li>Impiegato</li> </ul>	<ul style="list-style-type: none"> <li>Nome Impiegato</li> <li>Data Minima Pensionamento</li> </ul>
<b>Totale</b> 1 RET	<b>Totale</b> 2 DET

**Passo 5 Determinare la complessità funzionale (Applicazione Pensioni)**

1 RET e 2 DET	La complessità è Bassa
---------------	------------------------

**Passo 6 Determinare la dimensione funzionale (Applicazione Pensioni)**

Dimensione Funzionale di 1 EIF di bassa complessità	5 FP
---	------

## Esempio: Utilizzi multipli di dati

### Requisiti utente

Le informazioni sull'Impiegato sono mantenute dall'applicazione RU.

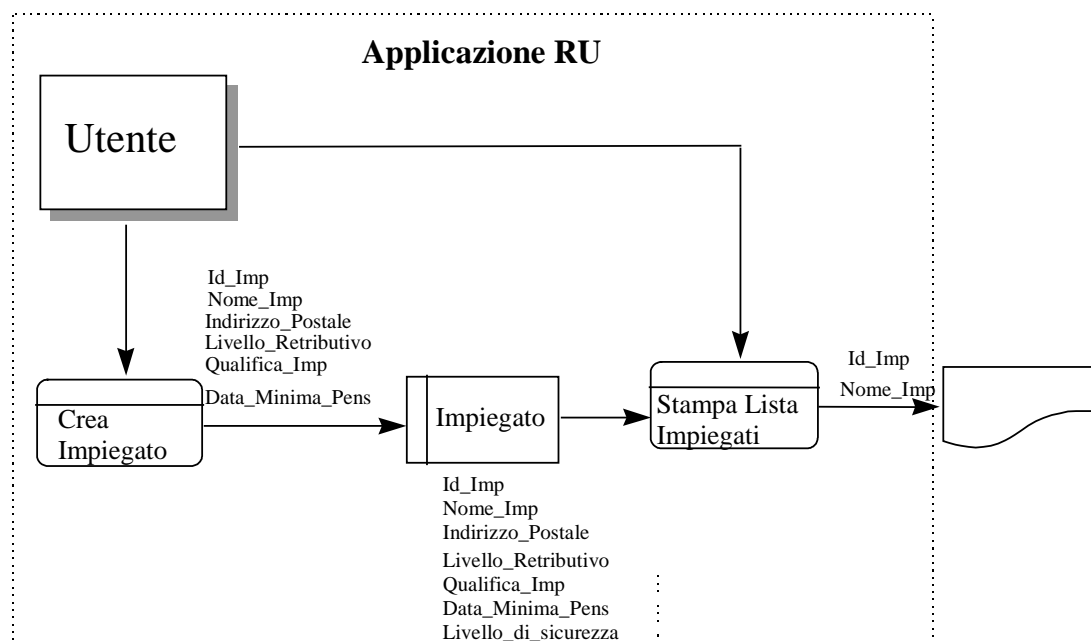
L'utente dell'applicazione RU richiede la possibilità di generare un elenco di tutti gli impiegati.

Le informazioni che devono essere visualizzate per ciascun impiegato comprendono:

- ID Impiegato
- Nome Impiegato

### Diagramma di flusso dei dati

Il seguente diagramma illustra il flusso dei dati per questo esempio.



### Passo 1 Identificare le funzioni di tipo dati

La seguente tabella mostra l'analisi sintetica per determinare se le informazioni sull'impiegato sono una funzione di tipo dati per l'applicazione RU.

Regole di identificazione delle funzioni di tipo dati	Si applica la regola?
1. Identificare tutti i dati o le informazioni di controllo logicamente collegati e riconoscibili dall'utente nell'ambito del conteggio.	Impiegato.
2. Escludere entità non mantenute da alcuna applicazione.	Le informazioni sull'Impiegato sono mantenute all'interno dell'applicazione RU. Queste non sono escluse.
3. Raggruppare entità collegate dipendenti.	Non ci sono entità di questo tipo.
4. Escludere entità referenziate come dati di decodifica.	Non ci sono entità di questo tipo.
5. Escludere entità che non contengono attributi richiesti dall'utente.	Non ci sono entità di questo tipo.
6. Rimuovere entità associative che contengono attributi aggiuntivi non richiesti dall'utente e entità associative che contengono solo chiavi esterne; raggruppare gli attributi di chiave esterna con le entità primarie.	Non ci sono entità di questo tipo.

## Passo 2

### Classificare le funzioni di tipo dati

La seguente tabella mostra l'analisi sintetica per determinare se le informazioni sull'impiegato che sono utilizzate per creare l'elenco degli impiegati sono classificate anche come un EIF per l'applicazione RU.

Regole di classificazione delle funzioni di tipo dati	Si applica la regola?
1. Classificare come ILF, se il gruppo di dati è mantenuto dall'applicazione che si sta misurando.	Impiegato è mantenuto dall'applicazione RU. Classificarlo come ILF.
2. Classificare come un EIF, se il gruppo di dati: <ul style="list-style-type: none"> <li>• è referenziato, ma non mantenuto, dall'applicazione che si sta misurando, e</li> <li>• è identificata come ILF in una o più ulteriori applicazioni.</li> </ul>	<p>È classificato come ILF; di conseguenza, non è identificato come EIF.</p> <p>Non applicabile.</p>

Le informazioni sull'Impiegato utilizzate per creare l'Elenco degli Impiegati non è un EIF per l'applicazione RU.

Questa pagina è lasciata intenzionalmente vuota.



### Esempi di conteggio delle funzioni di tipo transazionale

**Introduzione** Questa sezione utilizza un'applicazione Risorse Umane (RU) per illustrare le procedure usate per contare le funzioni di tipo transazionale. In aggiunta a questa sezione, vi sono esempi nei Casi di Studio inclusi nella corrispondente documentazione IFPUG.

**Attenzione:** Gli esempi in questa sezione e in tutto il manuale hanno due obiettivi:

1. Illustrare come si applicano le Regole di conteggio dei function point per un dato insieme di requisiti utente.
2. Permettere di esercitarsi sulle procedure di conteggio.

Ciascun conteggiatore deve:

- analizzare gli specifici requisiti utente che si applicano a ciascun progetto o applicazione oggetto di conteggio,
- effettuare il conteggio sulla base di tali requisiti.

**Contenuti** Questa sezione spiega come sono organizzati gli esempi e comprende esempi dettagliati per ciascuna funzione di tipo transazionale.

Argomento	Pagina
Organizzazione degli esempi di conteggio	2-2
Panoramica delle procedure relative alle funzioni di tipo transazionale	2-5
Processi elementari: Esempi di identificazione	2-7
EI: Esempi di conteggio	2-61
EO: Esempi di conteggio	2-103
EQ: Esempi di conteggio	2-127

## Organizzazione degli esempi di conteggio

Questa sezione spiega come sono presentati gli esempi.

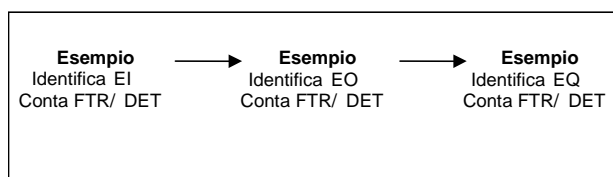
### Schema dell'organizzazione

Il seguente elenco illustra la sequenza di informazioni contenute negli esempi dettagliati. Per ogni esempio:

- Sono identificati gli EI, EO ed EQ.
- Sono contati gli FTR e i DET che contribuiscono alla complessità funzionale.

### Diagramma dell'organizzazione

Il seguente diagramma illustra l'organizzazione degli esempi.



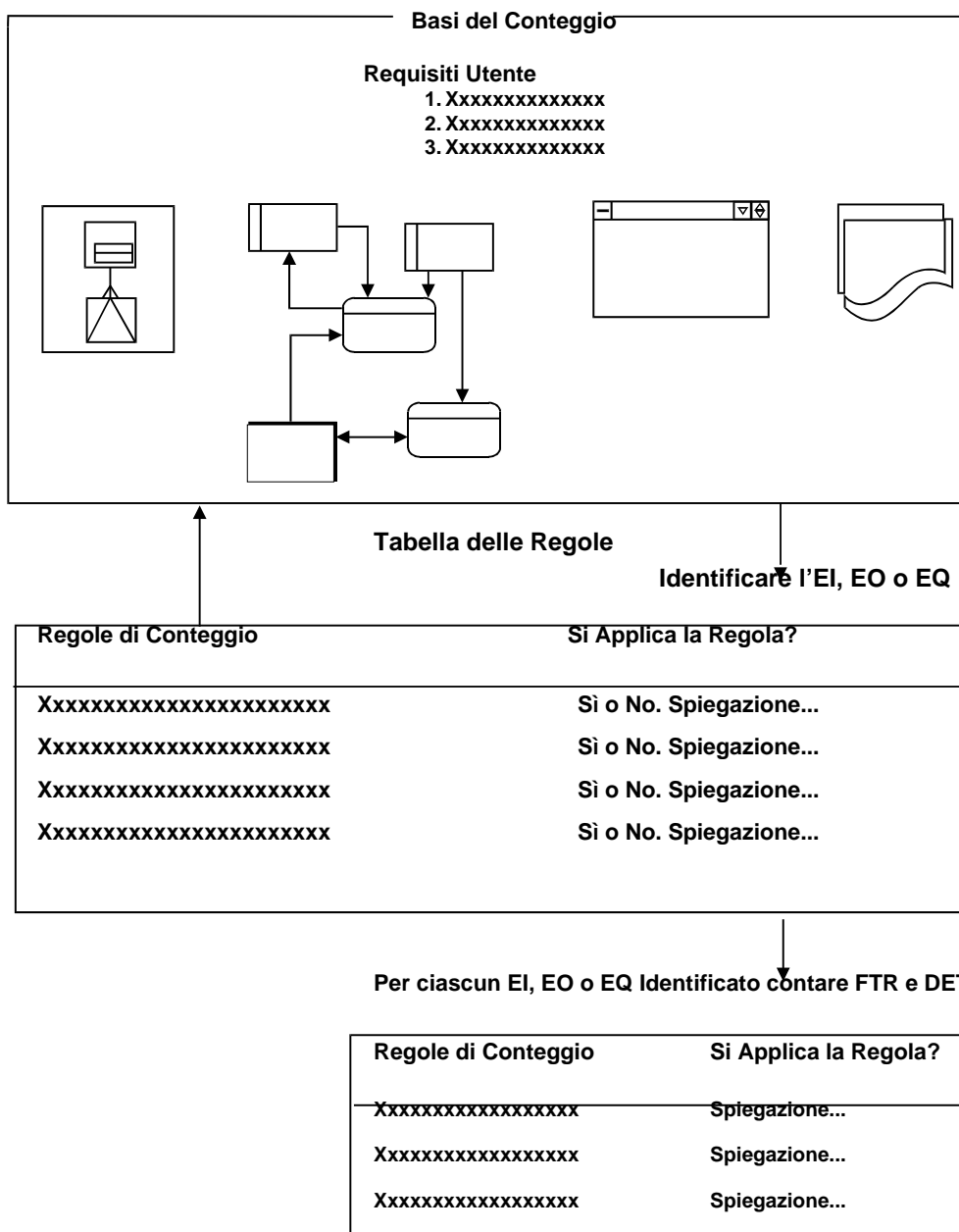
### Conteggio per ogni esempio

Ogni esempio comprende le seguenti componenti:

- basi per il conteggio,
- tabelle per l'applicazione delle regole di conteggio.

## Diagramma delle componenti

Il seguente diagramma illustra le componenti di ogni esempio ed il flusso delle informazioni.



**Basi per il conteggio**

Ogni esempio inizia con le basi per il conteggio. Come mostrato nel diagramma dei componenti, la misurazione della dimensione funzionale può basarsi sui seguenti elementi:

- requisiti utente,
- modelli dei dati e dei processi,
- finestre, schermate o report.

**Nota:** Non tutti i componenti del diagramma sono inclusi in tutti gli esempi. In alcuni esempi, i requisiti costituiscono l'unica base per il conteggio. Altri esempi includono il modello dei dati o dei processi, finestre, schermate e report.

**Tabelle delle regole di conteggio**

L'analisi per identificare le funzioni è presentata in una tabella che elenca le regole per il conteggio relative al tipo di funzione considerato. Le regole sono applicate ai componenti che costituiscono la base per il conteggio. L'analisi è spiegata nella tabella tramite la colonna "Si applica la regola?"

**Note:** Se tutte le regole si applicano, l'esempio è contato come un EI, un EO o un'EQ.

Le tabelle successive mostrano le regole e le spiegazioni per la complessità per ciascun tipo di funzione identificata.

## **Panoramica delle procedure relative alle funzioni di tipo transazionale**

Il processo di analisi di tutti gli esempi ricalca il processo descritto in precedenza in questo capitolo. I passi del processo riguardano l'applicazione delle regole per misurare le funzionalità di tipo transazionale, così come definite nella Parte 1 e sono:

- Identificare ogni Processo Elementare (PE) richiesto dall'utente
- Determinare l'univocità del/dei PE
- Classificare ogni PE in External Input (EI), External Output (EO), o External Inquiry (EQ)
- Conteggiare i Tipi di File Referenziati (FTR) per ogni funzione di tipo transazionale
- Conteggiare i Tipi di Elementi Dati (DET) per ogni funzione di tipo transazionale
- Determinare la complessità funzionale per ogni funzione di tipo transazionale
- Determinare la dimensione funzionale per ogni funzione di tipo transazionale

Questa pagina è lasciata intenzionalmente vuota.

---

## Processi elementari: Esempi di identificazione

---

**Introduzione** Questa sezione utilizza vari esempi per illustrare le procedure di identificazione dei processi elementari, ciascuno dei quali è a sé stante.

**Nota:** Ogni esempio si basa solo sui requisiti specifici relativi alla situazione illustrata, sebbene nella pratica, si debbano valutare tutti i requisiti e il loro impatto funzionale. Occasionalmente possono essere citati riferimenti ad altre funzioni esistenti ma non illustrate nello specifico esempio (ad es. Inserisci Impiegato piuttosto che Modifica Impiegato).

**Contenuti** Questa sezione include i seguenti esempi:

Argomento	Pagina
Descrizione sintetica degli esempi di identificazione dei processi elementari	2-8
Esempi di conteggio delle funzioni di tipo transazionale	2-9
Esempio: Stampa di un assegno / Aggiornamento del conto	16
Esempio: Lista delle assegnazioni lavori	2-21
Esempio: Stampa assegnazioni lavori / Memorizzazione dei criteri	2-26
Esempio: Impiegato / Informazioni sul colloquio d'assunzione	2-31
Esempio: Impiegato / Informazioni sulla patente	2-36
Esempio: Batch di alimentazione dati dell'impiegato	2-41
Esempio: Assegnazione dell'impiegato ad un lavoro	2-47
Esempio: Processi elementari simili	2-56

## Descrizione sintetica degli esempi di identificazione dei processi elementari

Gli esempi per l'identificazione dei processi elementari sono descritti nella tabella seguente.

<b>Esempio</b>	<b>Descrizione Sintetica</b>	<b>Pagina</b>
Nuovo impiegato / Informazioni sui dipendenti	Questo esempio dimostra che processi multipli possono costituire un singolo processo elementare.	2-9
Stampa di un assegno / Aggiornamento del conto	Questo esempio illustra il concetto dell'intento primario di un processo elementare.	2-16
Lista delle assegnazioni lavori	Questo esempio mostra che l'inserimento di un criterio di selezione per la produzione di un report non è un processo elementare.	2-21
Stampa assegnazioni lavori / memorizzazione dei criteri	Questo esempio mostra chiaramente come la memorizzazione dei criteri di selezione per un successivo uso, costituisce un processo elementare separato.	2-26
Impiegato / informazioni sul colloquio d'assunzione	Questo esempio mostra un altro caso di processi multipli che costituiscono un singolo processo elementare.	2-31
Impiegato / informazioni sulla patente di guida	Questo esempio è un terzo caso di processi multipli che costituiscono un singolo processo elementare.	2-36
Batch alimentazione dati dell'impiegato	Questo esempio mostra che la creazione batch di report di errori e la creazione di report statistici non sono processi elementari separati.	2-41
Assegnazione dell'impiegato ad un lavoro	Questo esempio mostra come valutare Processi Elementari simili per determinare se sono univoci.	2-47
Processi elementari simili	Questo esempio mostra due Processi Elementari simili che sono contati come un'unica transazione.	2-56



## Esempio: Nuovo impiegato / Informazioni sui dipendenti

### Requisiti utente

Quando aggiunge un nuovo impiegato, l'utente deve immettere:

1. i dati (basilari) sull'impiegato, e
2. informazioni sui dipendenti, se il loro numero è maggiore di zero.

Durante la variazione delle informazioni dell'impiegato, viene creato un file di transazione. Questo file viene inviato periodicamente (ad es. alla fine della giornata) all'applicazione Benefici.

**Nota:** La variazione dei dati di un impiegato esistente, con i dati del dipendente, non è incluso in questo esempio. Si rimanda al Caso di Studio 1-3 per un'illustrazione del conteggio delle variazioni.

**Inserimento  
Informazioni  
Impiegato  
senza  
Informazioni  
dei Dipendenti**

Determinare se l'inserimento delle informazioni dell'impiegato prive delle informazioni sui dipendenti sia un processo elementare. La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Comporre e/o decomporre i Requisiti Funzionali Utente nella più piccolo unità di attività, che soddisfa tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. Inserire un impiegato fa parte dei requisiti utente funzionali.</p> <p>No. La transazione completa comprende l'inserimento dei dati dell'impiegato con i dati associate del dipendente, se il numero di dipendenti è maggiore di zero. Questo passo non può essere logicamente separato.</p> <p>No. Per completare il processo elementare è necessario lo step susseguente di inserire i dati del dipendente associato.</p> <p>No. L'inserimento dell'impiegato senza l'inserimento del dipendente non lascia l'applicazione in uno stato di coerenza funzionale. Per rimanere in uno stato di coerenza funzionale deve essere soddisfatto il requisito utente senza bisogno di fare altro. Per soddisfare il requisito utente funzionale bisogna inserire anche il dipendente.</p>
<p>2. Per ogni unità di attività indenticata identificare un processo elementare che soddisfa tutti i criteri sopraindicati.</p>	<p>Inserire un impiegato (senza inserire i dipendenti) non soddisfa tutti i criteri.</p>

**Conclusione** L'inserimento delle informazioni dell'impiegato prive delle informazioni sui dipendenti associate non risponde ai requisiti di processo elementare.

**Inserimento delle sole informazioni dei Dipendenti**

Determinare se l'inserimento delle sole informazioni sui dipendenti senza le informazioni base dell'impiegato sia un processo elementare. La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. Inserire un dipendente fa parte dei requisiti utente funzionali.</p> <p>No. Questa attività non è una transazione completa in quanto non può essere eseguita indipendentemente dall'inserimento dell'impiegato.</p> <p>No. Questa attività non è autoconsistente in quanto non può essere eseguita indipendentemente dall'inserimento dell'impiegato.</p> <p>No. Inserire il dipendente senza inserire l'impiegato non lascia l'applicazione in uno stato di coerenza funzionale. Per rimanere in uno stato di coerenza funzionale deve essere soddisfatto il requisito utente senza bisogno di fare altro. Per soddisfare il requisito utente funzionale sia l'impiegato, che il dipendente associato, devono essere inseriti.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>Inserire un dipendente (senza il suo impiegato) non soddisfa tutti i criteri. Non è un processo elementare.</p>

**Conclusione**

Inserire le sole informazioni sul Dipendente non soddisfa i requisiti di un Processo Elementare. In questo esempio inserire le informazioni sul suo dipendente (senza inserire le informazioni sull'impiegato) non soddisfa tutti i criteri. Altri sistemi potrebbero occuparsi della manutenzione dei dati sul dipendente, indipendentemente dall'impiegato.

**Inserimento  
dell'Impiegato  
con le  
informazioni  
dei Dipendenti**

Per un impiegato che ha dei dipendenti, determinare se l'inserimento delle informazioni sull'impiegato, insieme alle associate informazioni sui dipendenti sia un processo elementare. La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti: <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. Inserire un impiegato con le informazioni sui dipendenti fa parte dei requisiti utente funzionali.</p> <p>Si. Le informazioni sull'impiegato e sui suoi dipendenti sono usate insieme per inserire un impiegato nel sistema RU. Questi passi non possono essere logicamente separati.</p> <p>Si. Questa attività è significativa di per sé e tutte le informazioni necessarie sono aggiunte all'applicazione RU</p> <p>Si. L'applicazione rimane in stato di coerenza funzionale all'inserimento dell'impiegato con i suoi dipendenti associati.</p>
2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.	L'inserimento delle informazioni dell'impiegato insieme alle informazioni sui dipendenti risponde a tutti i requisiti di un processo elementare sopra enunciati.

**Conclusione**

Inserire le informazioni dell'impiegato insieme alle informazioni sui dipendenti risponde ai requisiti di un processo elementare.

Ci potrebbero essere diversi modi di implementare il requisito di inserire uno o più dipendenti di un impiegato. Per esempio:

- un campo di input chiamato Numero di Dipendenti sulla maschera delle informazioni dell'impiegato che attiva la visualizzazione della maschera delle informazioni dei dipendenti
- un bottone che permette la visualizzazione della maschera dei dipendenti
- una voce di menu sulla maschera dell'impiegato, che permette la visualizzazione della maschera dei dipendenti
- la possibilità di inserire i dipendenti dalla maschera dell'impiegato

Indipendentemente dalla scelta implementativa del requisito esiste sempre un solo processo elementare: aggiungere un impiegato con i suoi eventuali dipendenti.

Non si possono identificare processi elementari separati a causa della opzionalità di un dato (es. aggiungere un impiegato con dipendenti, aggiungere un impiegato senza dipendenti).

### Determinare l'univocità del Processo Elementare

La seguente tabella illustra l'analisi per determinare se il processo elementare è unico.

Determinare l'univocità del Processo Elementare	Si applica la regola?
<p>1. Dopo il confronto con un altro Processo Elementare (EP) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi:</p> <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>Si. Aggiungere un impiegato richiede lo stesso numero di DET di Modificare un impiegato.</p> <p>Si. Aggiungere un impiegato richiede lo stesso numero di FTR di Modificare un impiegato.</p> <p>No. Aggiungere un impiegato ha un processo logico diverso che Modificare un impiegato.</p>
<p>2. Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.</p>	<p>Il requisito funzionale è da considerarsi come una singola funzione.</p> <p>Aggiungi impiegato non si scompone in due processi elementari (es. Aggiungi solo l'impiegato, Aggiungi solo i dipendenti).</p>

### Conclusione

Il processo elementare Aggiungi un impiegato è unico rispetto a tutti gli altri processi elementari identificati.

**Invio File  
Transazione ad  
Applicazione  
Benefici**

Determinare se l'invio di un file di transazione all'applicazione Benefici sia un ulteriore processo elementare. La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. Inviare il file Transazione alla applicazione Benefici è parte di un requisito funzionale dell'utente.</p> <p>SI. Inviare il file Transazione alla applicazione Benefici è una transazione completa. Logicamente è una funzionalità separata dalla funzionalità di Aggiungi un impiegato.</p> <p>Si. L'attività di inviare una transazione all'applicazione Benefici è autoconsistente. Il file Transazione è inviato indipendentemente (es. alla fine di ogni giornata) dalla funzionalità Aggiungi impiegato.</p> <p>Si. Dopo che le transazioni sono state inviate all'applicazione Benefici, l'applicazione HR è in uno stato consistente. I requisiti utente sono stati pienamente soddisfatti senza nulla da fare in più.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>L'invio del file Transazione all'applicazione Benefici soddisfa tutti i criteri sopra indicati.</p>

**Conclusione** Invia file di transazione ad Applicazione Benefici soddisfa i requisiti di un processo elementare.

### Determinare l'univocità del processo elementare

La seguente tabella illustra le analisi per determinare se il processo elementare sia unico.

Determinare l'univocità del Processo Elementare	Si applica la regola?
<p>1. Dopo il confronto con un altro Processo Elementare (EP) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi:</p> <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>No. Inviare il file transazione all'applicazione Benefici richiede DET differenti rispetto ad altri PE identificati.</p> <p>Sì. Inviare il file transazione all'applicazione Benefici richiede lo stesso insieme di FTR di Aggiungi impiegato.</p> <p>No. Inviare il file transazione all'applicazione Benefici ha una differente differente logica processuale rispetto ad altri PE.</p>
<p>2. Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.</p>	<p>Non vi è nulla da scomporre.</p>

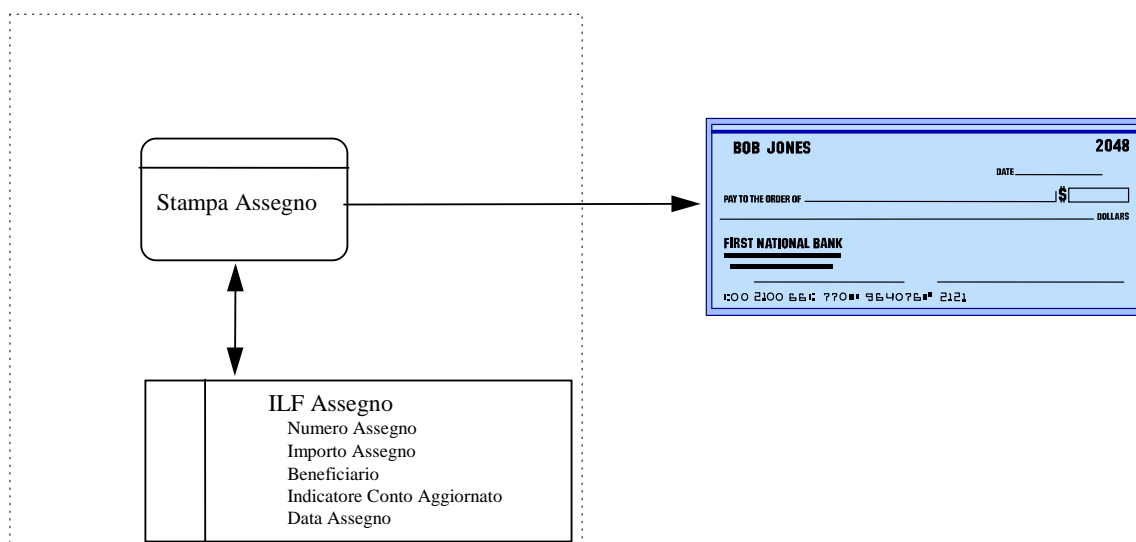
**Conclusione** Inviare un file Transazione all'applicazione Benefici è unico rispetto agli altri processi elementari identificati.

## Esempio: Stampa di un assegno / Aggiornamento del conto

### Requisiti utente

Stampare un assegno e, come risultato, aggiornare il conto per tenere traccia del pagamento dell'assegno. Tutti i dati stampati sull'assegno sono già memorizzati nel file degli assegni.

Il seguente diagramma illustra il flusso dei dati per questo esempio.





**Aggiornamento del conto** Determinare se aggiornare il conto per far risultare il pagamento sia un processo elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. L'Aggiornamento del conto è parte dei requisiti funzionali dell'utente.</p> <p>No. L'Aggiornamento del conto non è una transazione completa, fino a quando l'assegno non sia stampato. Questi passi non possono essere logicamente separati.</p> <p>No. L'Aggiornamento del conto senza anche la stampa dell'assegno non è una funzionalità autoconsistente. Il conto non può essere registrato come pagato indipendentemente dal fatto che l'assegno sia stampato.</p> <p>No. L'Aggiornamento del conto senza la stampa dell'assegno non lascia l'applicazione in uno stato di coerenza funzionale. Affinchè l'applicazione sia in uno stato di coerenza funzionale occorre che sia soddisfatto il requisito utente, nulla più. In questo caso il requisito utente richiede che l'aggiornamento del conto avvenga assieme all'operazione di stampa dell'assegno.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>L'Aggiornamento del conto non soddisfa tutti i criteri. Non è un processo elementare.</p>

**Conclusione** L'aggiornamento del conto non soddisfa i requisiti per essere un processo elementare.

**Stampa dell'assegno**

Determinare se stampare l'assegno senza l'aggiornamento del conto sia un processo elementare. La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. Stampare l'assegno è parte dei requisiti funzionali dell'utente.</p> <p>No. Stampare l'assegno non è una transazione completa., a meno che non sia aggiornato anche il conto. Questi passi non possono essere separati logicamente.</p> <p>No. Stampare l'assegno non è autoconsistente fino a quando il passo successivo di aggiornamento del conto non è stato fatto.</p> <p>No. Stampare l'assegno senza l'aggiornamento del conto non lascia l'applicazione in uno stato di coerenza funzionale. Affinchè l'applicazione sia in uno stato di coerenza funzionale occorre che sia soddisfatto il requisito utente, nulla più Affinchè il requisito utente sia soddisfatto occorre anche che il conto sia aggiornato.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>Stampare l'assegno non soddisfa tutti I criteri. Non è un processo elementare.</p>

**Conclusione**

Stampare l'assegno non soddisfa i requisiti utente per essere un processo elementare.

**Stampa dell'assegno e Aggiornamento del conto**

Determinare se stampare l'assegno e aggiornare il conto per far risultare il pagamento sia un processo elementare.

La seguente tabella illustra l'analisi

Regole conteggio Processo Elementare	Si applica la regola?
1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti: <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. Stampare l'assegno e Aggiornare il conto è parte dei requisiti funzionali.</p> <p>Si. Stampare l'assegno e Aggiornare il conto è una transazione completa. Questi passi non possono essere separati .</p> <p>Si. Stampare l'assegno e aggiornare il conto è autoconsistente. Non ci sono passi precedenti o successivi che necessitano di essere eseguiti.</p> <p>Si. Stampare un assegno e Aggiornare il conto lascia l'applicazione in uno stato di coerenza funzionale. Il requisito funzionale dell'utente è pienamente soddisfatto.</p>
2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.	<p>Stampare l'assegno e Aggiornare il conto soddisfa tutti i criteri. E' un processo elementare.</p>

**Conclusione**

Stampare un assegno e Aggiornare il conto soddisfa i criteri di un processo elementare.

Il requisito dell'utente è stampare un assegno. Il fatto di Aggiornare il conto è parte del processo di stampa. Stampare e aggiornare il conto costituiscono la più piccola unità di attività che è significativa all'utente. L'intero processo è significativo per l'utente, costituisce una transazione completa, è autoconsistente e lascia l'applicazione oggetto del conteggio in una situazione di coerenza funzionale.

**Determinare l'univocità del processo elementare**

La seguente tabella illustra le analisi per determinare se il processo elementare sia unico.

Determinare l'univocità del Processo Elementare	Si applica la regola?
<p>1. Dopo il confronto con un altro Processo Elementare (EP) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi:</p> <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>No. L'insieme dei DET di Stampare l'assegno e Aggiornare il conto è completamente differente da quello degli altri PE.</p> <p>No. L'insieme dei FTR di Stampare l'assegno e Aggiornare il conto è completamente differente da quello degli altri PE.</p> <p>No. Stampare l'assegno e Aggiornare il conto ha una differente logica processuale rispetto ad altri PE.</p>
<p>2. Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.</p>	<p>Aggiornare il conto è parte del processo logico di stampare l'Assegno.</p>

**Conclusione**

Il processo elementare di Stampare l'assegno e Aggiornare il conto è unico da tutti gli altri processi elementari identificati.

## Esempio: Lista delle assegnazioni lavori

### Requisiti utente

Visualizzare una lista delle assegnazioni lavori per un dato intervallo temporale. L'utente sarà in grado di immettere i criteri di selezione. Non è richiesto di memorizzare i criteri di selezione dopo che il report è stato generato. Il seguente diagramma illustra il flusso dati per questo esempio.

Criteri per la Lista Assegnazione Lavori	
ID Impiegato	<input type="text"/>
Data inizio	<input type="text"/>
Data fine	<input type="text"/>
	<input type="button" value="Stampa"/>
	<input type="button" value="Annulla"/>

**Immissione dei criteri di selezione** Determinare se l'immissione dei criteri di selezione (senza visualizzare la lista assegnazione lavori) sia un processo elementare.

La seguente tabella illustra l'analisi

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. L'immissione dei criteri di selezione è parte dei requisiti funzionale dell'utente.</p> <p>No. L'immissione dei criteri di selezione (senza la lista delle assegnazioni lavori) non è una transazione completa, Questi passi non possono essere logicamente separati</p> <p>No. L'immissione dei criteri di selezione (senza la lista delle assegnazioni lavori) non è autoconsistente. La Visualizzazione delle assegnazioni lavoro è un successivo step necessario per completare il processo elementare.</p> <p>No. L'immissione dei criteri di selezione (senza la lista delle assegnazioni lavori) non lascia l'applicazione in uno stato di coerenza funzionale. Affinchè l'applicazione sia in uno stato di coerenza funzionale occorre che sia soddisfatto il requisito utente, nulla più. Per soddisfare il requisito utente , occorre visualizzare la Lista Assegnazione lavori</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>L'immissione dei criteri di selezione (senza la lista delle assegnazioni lavori) non soddisfa i criteri.</p>

**Conclusione** L'Immissione dei criteri di selezione non soddisfa i requisiti per essere un processo elementare.

**Lista assegnazione lavori**

Determinare se la visualizzazione della lista assegnazione lavori (senza l'immissione dei criteri di selezione) sia un processo elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. La Lista Assegnazione Valori è parte dei requisiti funzionali dell'utente .</p> <p>No. La Lista Assegnazioni Lavori (senza l'immissione dei criteri di selezione) non può essere fatta senza l'immissione dei criteri di selezione. Questi passi non possono essere logicamente separati.</p> <p>No. La Lista Assegnazioni Lavori (senza l'immissione dei criteri di selezione) non è autoconsistente. L'Assegnazione Lavori non può essere visualizzata indipendentemente dal precedente passo si immissione dei criteri di selezione.</p> <p>La Lista Assegnazioni Lavori (senza l'immissione dei criteri di selezione) non lascia l'applicazione in uno stato di coerenza funzionale. Affinchè l'applicazione sia in uno stato di coerenza funzionale occorre che sia soddisfatto il requisito utente, nulla più. Per soddisfare il requisito utente, occorre immettere i criteri di selezione.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>La Lista Assegnazioni Lavori (senza l'immissione dei criteri di selezione) non soddisfa i criteri.</p>

**Conclusione** La Lista Assegnazione Lavori (senza l'immissione dei criteri di selezione) non soddisfa i requisiti per questo processo elementare.

**Immissione dei  
criteri di  
selezione e Lista  
assegnazione  
lavori**

Determinare se l'immissione dei criteri di selezione e la lista assegnazione lavori costituiscano un processo elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Sì. L'immissione dei criteri di selezione e la Lista Assegnazione Lavori è parte dei requisiti funzionali dell'utente.</p> <p>Sì. L'immissione dei criteri di selezione e la Lista Assegnazione Lavori costituisce una transazione completa. Questi passi non possono essere logicamente separati.</p> <p>Sì. L'immissione dei criteri di selezione e la Lista Assegnazione Lavori è autoconsistente. Non ci sono passi né precedenti, né successivi che hanno bisogno di essere eseguiti.</p> <p>Sì. L'immissione dei criteri di selezione e la Lista Assegnazione Lavori lascia l'applicazione in uno stato di coerenza funzionale. Affinchè l'applicazione sia in uno stato di coerenza funzionale occorre che sia soddisfatto il requisito utente, nulla più. Per soddisfare il requisito utente, occorre immettere i criteri di selezione e visualizzare la Lista Assegnazione Lavori.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>L'immissione dei criteri di selezione e la Lista Assegnazione Lavori soddisfa tutti i criteri sopracitati.</p>

**Conclusione** L'immissione dei criteri di selezione e la lista assegnazione lavori soddisfano i criteri per un processo elementare.

Le informazioni di controllo costituiscono il lato di input di un EO o EQ. La richiesta che specifica cosa e/o come i dati devono essere reperiti o generati fa parte del processo elementare che fornisce i dati all'utente e non è un processo elementare di per sé.

L'immissione del criterio di selezione non è la più piccola unità di attività significativa per l'utente. Tale processo non è autonomo perché non può essere eseguito indipendentemente dalla produzione del report. La combinazione dei due processi (immissione dei criteri di selezione e generazione del report) costituisce invece la più piccola unità di attività significativa per l'utente, è autonoma e lascia l'applicazione in uno stato di coerenza funzionale.



### Determinare l'univocità del processo elementare

La seguente tabella illustra le analisi per determinare se il processo elementare sia unico

Determinare l'univocità del Processo Elementare	Si applica la regola?
<p>1. Dopo il confronto con un altro Processo Elementare (EP) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi:</p> <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>No. L'insieme dei DET dell'immissione dei criteri di selezione e della Lista Assegnazione Lavori è differente da quello degli altri PE.</p> <p>No. L'insieme dei FTR dell'immissione dei criteri di selezione e della Lista Assegnazione Lavori è differente da quello degli altri PE.</p> <p>No. L'insieme dei trattamenti logici per l'immissione dei criteri di selezione e della Lista Assegnazione Lavori è differente da quello degli altri PE.</p>
<p>2. Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.</p>	<p>È necessario l'immissione dei criteri di selezione al fine di visualizzare la Lista Assegnazione Lavori.</p>

### Conclusione

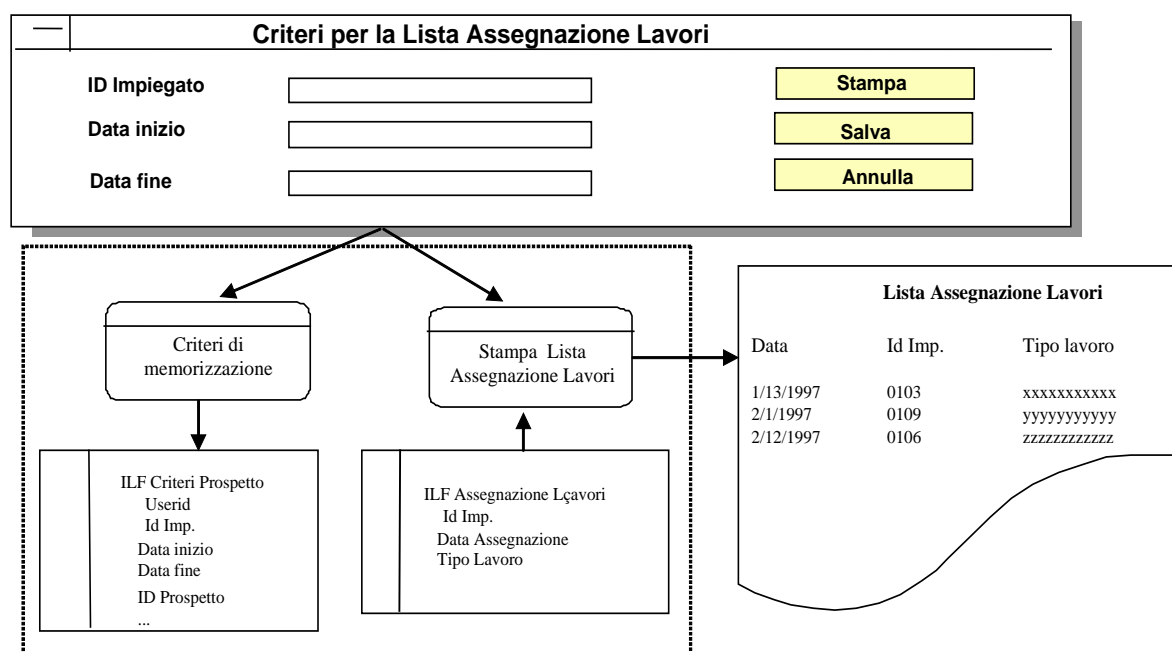
Il processo elementare dell'immissione dei criteri di selezione e della Lista Assegnazione Lavori è unico rispetto a tutti gli altri processi elementari identificati.

## Esempio: Stampa assegnazioni lavori / Memorizzazione dei criteri

### Requisiti utente

Stampare un elenco delle assegnazioni lavori per un dato intervallo temporale. L'utente sarà in grado di immettere i criteri di selezione. È richiesto di consentire all'utente di memorizzare i criteri di selezione per un uso futuro.

Il seguente diagramma illustra il flusso dei dati per questo esempio.



**Immissione e salvataggio dei criteri**

Determinare se la memorizzazione dei criteri di selezione (senza la stampa della lista assegnazione lavori) sia un processo elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. Salvare i criteri di selezione è parte dei requisiti funzionali dell'utente.</p> <p>Si. Salvare i criteri di selezione è una transazione completa. Essa è una funzionalità logicamente separata dalla lista assegnazione lavori.</p> <p>Si. Salvare i criteri di selezione è una transazione autoconsistente. Essa può essere eseguita indipendentemente dalla stampa della lista assegnazione lavori.</p> <p>Si. Salvare i criteri di selezione lascia l'applicazione in uno stato di coerenza funzionale. Il requisito funzionale è completamente soddisfatto senza dover fare nulla più.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>Salvare i criteri di selezione soddisfa tutti i criteri sopracitati .</p>

**Conclusione**

L'immissione e il salvataggio dei criteri di selezione soddisfa i requisiti per un processo elementare.

### Determinare l'univocità del processo elementare

La seguente tabella illustra le analisi per determinare se il processo elementare sia unico.

Determinare l'univocità del Processo Elementare	Si applica la regola?
<p>1. Dopo il confronto con un altro Processo Elementare (EP) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi:</p> <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>No. L'insieme dei DET del salvataggio dei criteri è differente da quello di qualunque altro PE.</p> <p>No. L'insieme dei FTR del salvataggio dei criteri è differente da quello di qualunque altro PE.</p> <p>No. L'insieme dei trattamenti logici per il salvataggio dei criteri è differente da quello di qualunque altro PE.</p>
<p>2. Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.</p>	<p>Non vi è niente da scomporre.</p>

### Conclusione

Il processo elementare Salvataggio dei criteri è unico rispetto a tutti gli altri processi elementari identificati.

**Stampa Lista assegnazione lavori**

Determinare se stampare la lista assegnazione lavoro, indipendentemente dalla memorizzazione dei criteri di selezione, sia un processo elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. La stampa della lista assegnazione lavori è parte dei requisiti funzionali dell'utente.</p> <p>Si. La stampa della lista assegnazione lavori costituisce una transazione completa. Essa è una funzionalità logicamente separata dal salvataggio dei criteri</p> <p>Si. La stampa della lista assegnazione lavori è una transazione autoconsistente. Essa può essere eseguita indipendentemente dal salvataggio dei criteri.</p> <p>Si. La stampa della lista assegnazione lavori lascia l'applicazione in uno stato di coerenza funzionale. Il requisito funzionale è completamente soddisfatto senza dover fare nulla più.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>La stampa della lista assegnazione lavori Soddisfa tutti i criteri sopracitati.</p>

**Conclusione** La stampa della lista assegnazione lavori è un processo elementare.

L'immissione dei criteri di selezione è indubbiamente significativa per l'utente, in quanto egli può memorizzare esplicitamente i criteri di selezione. Sia la stampa del report, sia la memorizzazione dei criteri, possono essere eseguiti indipendentemente, ed entrambe lasciano l'applicazione in uno stato di coerenza funzionale.

Entrambi i processi sono autonomi, sono significativi per l'utente e lasciano l'applicazione in uno stato di coerenza funzionale.

In conformità alle regole di identificazione del Processo Elementare, esistono due Processi Elementari.

**Determinare l'univocità del processo elementare**

La seguente tabella illustra le analisi per determinare se il processo elementare sia unico.

Determinare l'univocità del Processo Elementare	Si applica la regola?
<p>1. Dopo il confronto con un altro Processo Elementare (EP) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi:</p> <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>No. L'insieme dei DET della stampa della lista assegnazioni lavori è differente da quello di qualunque altro PE .</p> <p>No. L'insieme dei FTR della stampa della lista assegnazioni lavori è differente da quello di qualunque altro PE.</p> <p>No. L'insieme dei trattamenti logici della stampa della lista assegnazioni lavori è differente da quello di qualunque altro PE.</p>
<p>2. Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.</p>	<p>Non vi è niente da scomporre.</p>

**Conclusione**

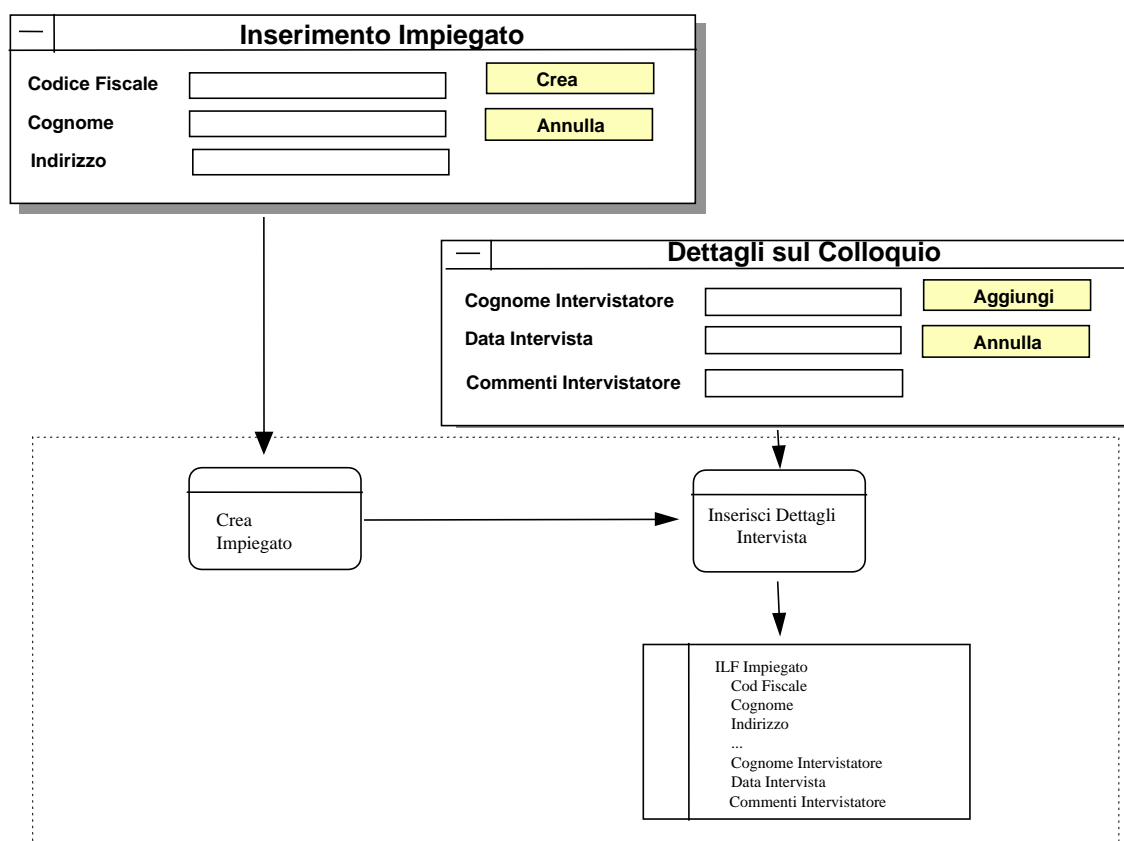
Il processo elementare della stampa della lista assegnazione lavori è unico rispetto a tutti gli altri processi elementari identificati.

## Esempio: Impiegato / Informazioni sul colloquio d'assunzione

### Requisiti utente

Inserendo un nuovo impiegato, oltre ai dati personali dell'impiegato (cioè Codice Fiscale, Nome, Indirizzo, ecc.), occorre immettere i dettagli relativi al colloquio d'assunzione. Le informazioni sul colloquio d'assunzione comprendono: il nome dell'intervistatore, la data del colloquio e i commenti dell'intervistatore.

Il seguente diagramma illustra il flusso dei dati per questo esempio.



**Immissione dei  
Dati Personali  
Impiegato**

Determinare se l'immissione delle sole informazioni personali sull'impiegato costituisca un Processo Elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Sì. L'immissione dei dati personali dell'impiegato è parte dei requisiti funzionali dell'utente.</p> <p>No. La transazione completa include l'immissione dei dati personali dell'impiegato come i dettagli del colloquio d'assunzione. Questi passi non possono essere separati logicamente.</p> <p>No. L'aggiunta delle informazioni dei dettagli del colloquio d'assunzione è un passo successivo necessario a completare il processo elementare.</p> <p>No. L'immissione dei dati personali senza l'immissione dei dettagli del colloquio d'assunzione non lascia l'applicazione in uno stato di coerenza funzionale. Affinchè l'applicazione sia in uno stato di coerenza funzionale occorre che sia soddisfatto il requisito utente, nulla più. Per soddisfare il requisito utente occorre che siano inseriti anche i dettagli del colloquio d'assunzione.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>L'immissione dei soli dati personali dell'impiegato non soddisfa tutti i criteri.</p>

**Conclusione**

L'immissione dei Dati Personali dell'Impiegato (senza l'immissione dei dettagli del colloquio) non soddisfano i requisiti per un processo elementare.



**Immissione dei  
Dettagli sul  
Colloquio di  
Assunzione**

Se l'immissione dei soli dettagli sul colloquio d'assunzione dell'impiegato sia un processo elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Sì. L'immissione dei dettagli del colloquio d'assunzione è parte dei requisiti funzionali.</p> <p>No. La transazione completa include l'immissione dei dati personali dell'impiegato come pure i dettagli del colloquio. Questi passi non possono essere logicamente separati.</p> <p>No. L'immissione dei soli dettagli del colloquio con l'impiegato non può essere fatta indipendentemente dall'inserimento dei dati personali dell'impiegato.</p> <p>No. L'immissione dei soli dettagli sul colloquio senza l'immissione dei dati personali dell'impiegato non lascia l'applicazione in uno stato di coerenza funzionale. Affinchè l'applicazione sia in uno stato di coerenza funzionale occorre che sia soddisfatto il requisito utente, nulla più. Per soddisfare il requisito utente occorre che siano inseriti sia i dettagli del colloquio d'assunzione che i dati personali dell'impiegato.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>L'immissione dei soli dati personali dell'impiegato non soddisfa tutti i criteri.</p>

**Conclusione**

L'immissione dei dettagli sul colloquio d'assunzione senza i dati personali dell'impiegato non soddisfa i requisiti per un processo elementare.

**Immissione dei Dati Personali e dei Dettagli sul Colloquio di Assunzione**

Determinare se l'immissione dei dati personali dell'impiegato, insieme ai dettagli sul colloquio d'assunzione, sia un processo elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. L'immissione dei dettagli del colloquio d'assunzione e dei dati personali dell'impiegato sono entrambi parte dei requisiti funzionali.</p> <p>Si. L'immissione dei dettagli del colloquio d'assunzione e dei dati personali dell'impiegato costituiscono insieme una transazione completa. Questi passi non possono essere logicamente separati.</p> <p>Si. L'immissione dei dettagli del colloquio d'assunzione e dei dati personali dell'impiegato insieme costituiscono una funzionalità autoconsistente.</p> <p>Si. L'immissione dei dettagli del colloquio d'assunzione e dei dati personali dell'impiegato lasciano l'applicazione in uno stato di coerenza funzionale. Per soddisfare il requisito utente entrambi i passi devono essere eseguiti.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>L'immissione dei dettagli del colloquio d'assunzione e dei dati personali dell'impiegato soddisfano tutti i criteri sopracitati.</p>

**Conclusione** L'immissione dei dati personali dell'impiegato e dei dettagli del colloquio di assunzione soddisfano i criteri per un processo elementare.

Se due processi di input sono sempre sequenziali e dipendenti (I passi uno e due sono obbligatori), allora vi è un unico processo elementare e una sola funzionalità.

Un nuovo impiegato non può essere registrato fintantoche non sono stati inseriti anche i dati relativi al suo colloquio d'assunzione. Inserire i dati personale dell'impiegato o i dati di dettaglio del colloquio da soli non costituirebbero la più piccola unità di attività significativa per l'utente.

L'immissione dei dati personali dell'impiegato insieme ai dettagli del colloquio di assunzione costituiscono invece la più piccola unità di attività significativa per l'utente, una transazione completa, autoconsistente e che lascia l'applicazione in uno stato di coerenza funzionale.

### Determinare l'univocità del processo elementare

La seguente tabella illustra le analisi per determinare se il processo elementare sia unico.

Determinare l'univocità del Processo Elementare	Si applica la regola?
<p>1. Dopo il confronto con un altro Processo Elementare (PE) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi:</p> <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>No. L'insieme dei DET dell'immissione dei dati personali dell'impiegato e dei dettagli del colloquio è differente da qualsiasi altro PE.</p> <p>No. L'insieme dei FTR dell'immissione dei dati personali dell'impiegato e dei dettagli del colloquio è differente da qualsiasi altro PE.</p> <p>No. L'insieme dei trattamenti logici dell'immissione dei dati personali dell'impiegato e dei dettagli del colloquio è differente da qualsiasi altro PE.</p>
<p>2. Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.</p>	<p>Un nuovo impiegato non può essere registrato fintanto che siano immessi i dati personali dell'impiegato e i dettagli del colloquio d'assunzione.</p>

### Conclusione

Il processo elementare dell'immissione dei dati personali dell'impiegato e dei dettagli del colloquio di è unico rispetto a tutti gli altri processi elementari identificati.

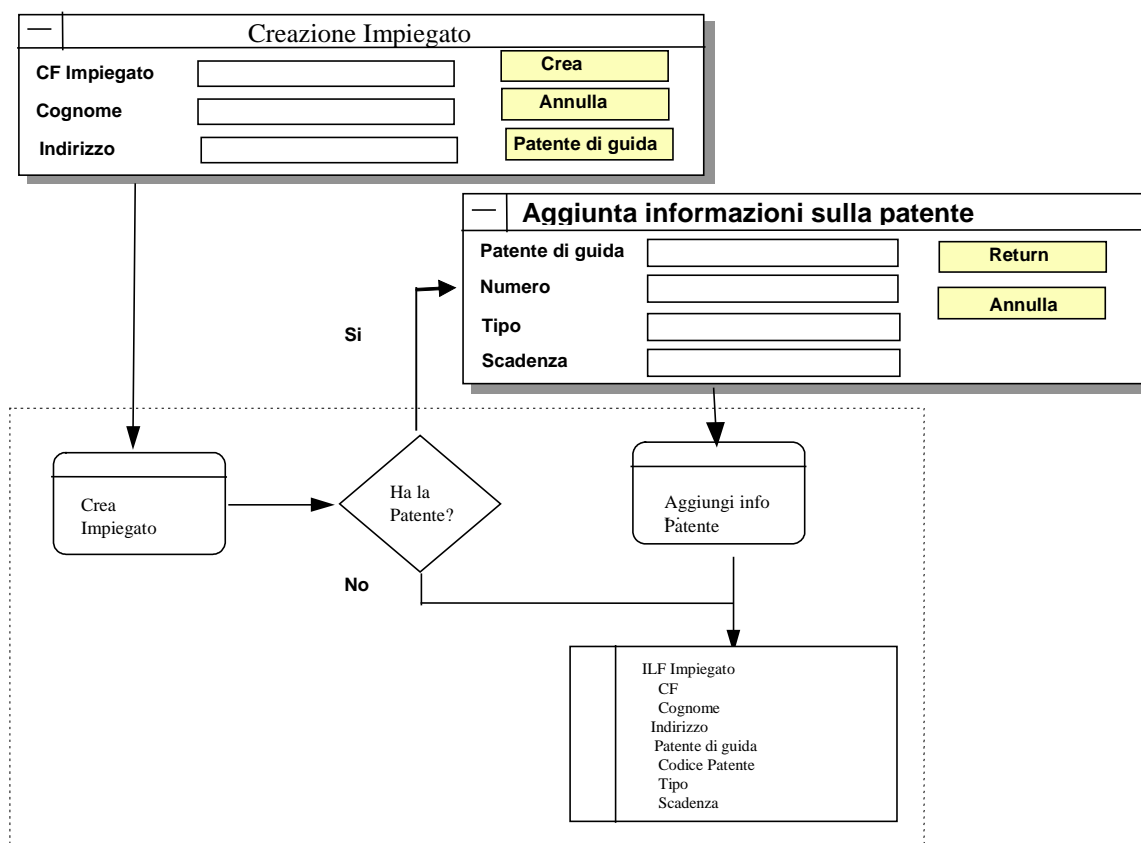
## Esempio: Impiegato / Informazioni sulla patente

### Requisiti utente

Quando si aggiunge un nuovo impiegato, sono inserite le informazioni relative al codice fiscale, nome, indirizzo e se l'impiegato ha o meno la patente. Se l'impiegato ha la patente è effettuato un secondo passo per l'immissione dei dati relativi quali codice della patente, tipo e data di scadenza.

**Nota:** La modifica delle informazioni di un impiegato esistente, incluse quelle sulla patente, non sono incluse in questo esempio.

Il seguente schema illustra il flusso dei dati per questo esempio.



**Immissione delle informazioni personali dell'impiegato**

Determinare se l'immissione delle informazioni personali dell'impiegato sia un processo elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. L'immissione delle informazioni dell'impiegato è parte dei requisiti funzionali.</p> <p>No. La transazione completa consiste nell'immissione dei dati personali dell'impiegato e dei relativi dati sulla patente di guida (se ce l'ha). Questi passi non possono essere separati logicamente</p> <p>No. L'immissione dei dati della patente di guida (se ce l'ha) è un passo successivo necessario per il completamento del processo elementare.</p> <p>No. L'immissione dei dati personali dell'impiegato senza l'immissione dei dati sulla patente di guida (se ce l'ha) non lascia l'applicazione in uno stato di coerenza funzionale. Affinchè l'applicazione sia in uno stato di coerenza funzionale occorre che sia soddisfatto il requisito utente, nulla più. Per soddisfare il requisito utente occorre che siano inseriti anche i dati sulla patente di guida (se ce l'ha).</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>La sola immissione dei dati personali dell'impiegato non soddisfa tutti i criteri.</p>

**Conclusione** L'immissione dei dati personali dell'impiegato senza l'immissione delle informazioni sulla patente di guida (se ce l'ha) non soddisfa i requisiti per un processo elementare.

**Immissione delle Informazioni sulla patente di guida**

Determinare se l'immissione delle informazioni sulla patente di guida senza l'immissione dei dati personali dell'impiegato sia un processo elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. L'immissione delle informazioni sulla patente di guida è parte dei requisiti funzionali.</p> <p>No. La transazione completa consiste nell'immissione dei dati personali dell'impiegato e dei relativi dati sulla patente di guida (se ce l'ha). Questi passi non possono essere separati logicamente</p> <p>No. L'immissione dei dati personali dell'impiegato è un passo precedente necessario per il completamento del processo elementare.</p> <p>No. L'immissione delle informazioni sulla patente di guida senza l'immissione dei dati personali dell'impiegato non lascia l'applicazione in uno stato di coerenza funzionale. Affinchè l'applicazione sia in uno stato di coerenza funzionale occorre che sia soddisfatto il requisito utente, nulla più. Per soddisfare il requisito utente occorre che siano inseriti sia i dati personali dell'impiegato che i relativi dati sulla patente di guida (se ce l'ha).</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>La sola immissione dei dati sulla patente di guida non soddisfa tutti i criteri.</p>

**Conclusione**

L'immissione delle informazioni sulla patente di guida senza l'immissione dei dati personali dell'impiegato non soddisfa i requisiti per un processo elementare.

**Immissione dell'Impiegato e della Patente di guida**

Determinare se l'immissione delle informazioni personali dell'impiegato insieme alle informazioni sulla patente di guida sia un processo elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. L'aggiunta di un impiegato e la memorizzazione dei dati relativi alla patente è parte dei requisiti funzionali dell'utente.</p> <p>Si. La transazione completa consiste nell'immissione dei dati personali dell'impiegato e dei relativi dati sulla patente di guida (se ce l'ha). Questi passi non possono essere separati logicamente.</p> <p>Si. L'aggiunta di un impiegato e la memorizzazione dei dati relativi alla patente è una funzionalità autoconsistente. Non vi sono passi né precedenti, né successivi che devono essere eseguiti.</p> <p>Si. L'aggiunta di un impiegato e la memorizzazione dei dati relativi alla patente (se ce l'ha) lasciano l'applicazione in uno stato di coerenza funzionale. Per soddisfare il requisito utente occorre che siano eseguiti entrambi i passi.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>L'aggiunta di un impiegato e la memorizzazione dei dati relativi alla patente soddisfano tutti i criteri.</p>

**Conclusione**

Se due processi di input sono sempre sequenziali e dipendenti, ma il secondo processo è opzionale (ma obbligatorio se si può applicare) allora vi è un unico processo elementare e una sola funzionalità.

L'aggiunta di un impiegato e delle informazioni sulla patente è un unico processo elementare. Se un impiegato non ha la patente, il passo "Immissione dei dati sulla patente di guida" è irrilevante. Se, viceversa, l'impiegato ha la patente un ulteriore passo di inserimento delle informazioni sulla patente di guida si rende necessario per completare il processo elementare al fine di lasciare l'applicazione in uno stato di coerenza funzionale.

### Determinare l'univocità del processo elementare

La seguente tabella illustra le analisi per determinare se il processo elementare sia unico

Determinare l'univocità del Processo Elementare	Si applica la regola?
<p>1. Dopo il confronto con un altro Processo Elementare (EP) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi:</p> <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>No. L'insieme dei DET dell'aggiunta dell'impiegato e dei relativi dati sulla patente di guida è differente da quello di qualunque altro PE.</p> <p>No. L'insieme dei FTR dell'aggiunta dell'impiegato e dei relativi dati sulla patente di guida è differente da quello di qualunque altro PE.</p> <p>No. L'insieme dei trattamenti logici dell'aggiunta dell'impiegato e dei relativi dati sulla patente di guida è differente da quello di qualunque altro PE.</p>
<p>2. Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.</p>	<p>Se un impiegato ha la patente di guida, un nuovo impiegato non può essere inserito fintantoché non siano stati inseriti i dati personali dell'impiegato insieme ai dati sulla patente di guida.</p>

### Conclusione

Il Processo elementare di Immissione dei dati personali dell'impiegato unitamente alle informazioni sulla sua patente di guida (se ce l'ha) è unico rispetto agli altri processi elementari identificati.

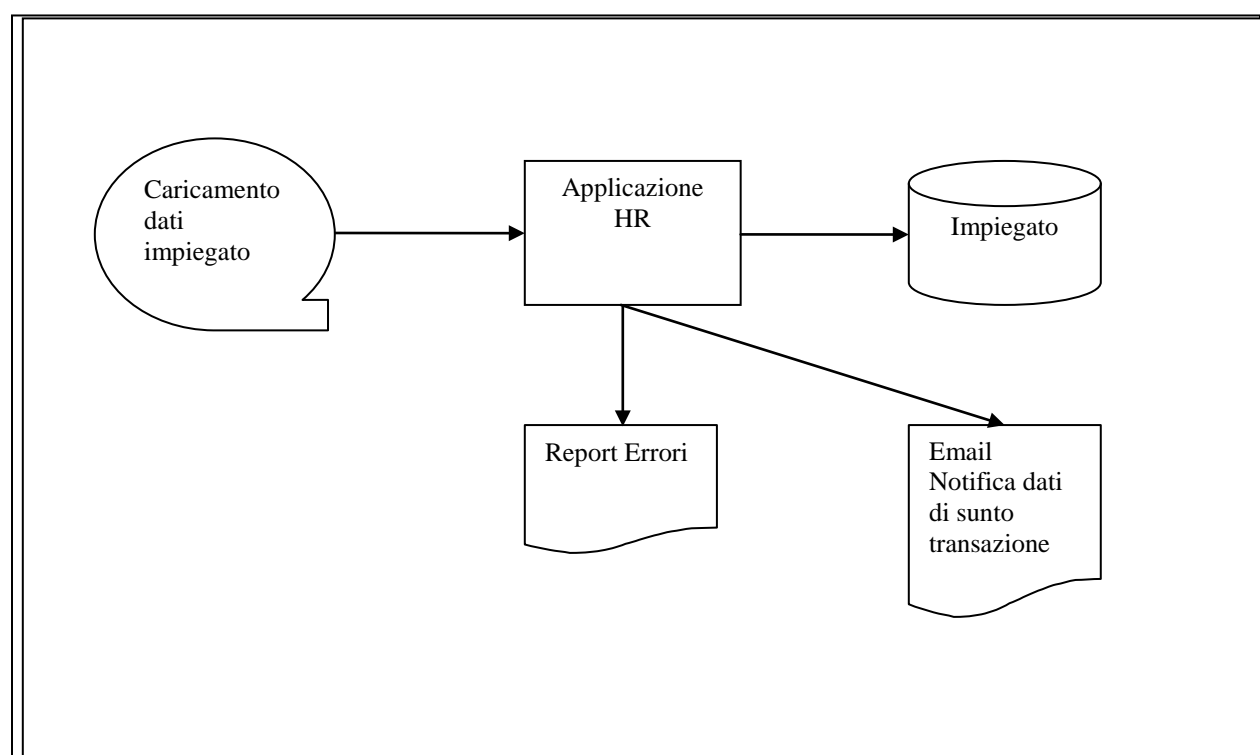


## Esempio: Batch di alimentazione dati dell'impiegato

### Requisiti utente

Un batch di caricamento dei dati, provenienti da un'altra applicazione, relativi ad un nuovo impiegato devono essere accettati. I dati dell'impiegato devono essere validati e memorizzati nel file Impiegato.E' prodotto un report di errori contenente ogni anomalia riscontrata. L'applicazione HR riceve tramite email una notifica con dei dati di sunto della transazione.

Il seguente diagramma illustra il flusso dati per questo esempio.



**Processo di caricamento dati dell'impiegato**

se la transazione di accettazione e caricamento dei dati dell'impiegato senza il report di errori e la notifica via email costituisca un Processo Elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. La transazione di accettazione e caricamento dei dati dell'impiegato è parte dei requisiti funzionali dell'utente.</p> <p>No. La transazione completa comprende il report di errori e la notifica dei dati relativamente alla funzione di caricamento. Questi passi non possono essere separati logicamente.</p> <p>No. La produzione del report di errori e della notifica via email sono passi successivi necessari per il completamento del processo elementare.</p> <p>No. La transazione di accettazione e caricamento dei dati dell'impiegato senza la produzione del report di errori e della notifica via email non lascia l'applicazione in uno stato di coerenza funzionale. Affinchè l'applicazione sia in uno stato di coerenza funzionale occorre che sia soddisfatto il requisito utente, nulla più. Per soddisfare il requisito utente occorre che la produzione del report di errori e della notifica via e.mail siano eseguiti dalla transazione di accettazione e caricamento dei dati dell'impiegato.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>Tutti i criteri sopracitati non sono soddisfatti.</p>

**Conclusione**

La transazione di accettazione e caricamento dei dati dell'impiegato senza il report di errori e la notifica via email dei dati di sunto non soddisfa i criteri di un processo elementare.

**Produzione di un report di errori**

Determinare se produrre un report di errori senza la transazione di accettazione e caricamento dei dati dell'impiegato costituisca un Processo Elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Sì. La produzione di un report di errori è parte dei requisiti funzionali dell'utente.</p> <p>No. La produzione di un report di errori è parte integrata del processo di aggiornamento e gli errori non possono essere definiti al di fuori di tale processo. Questi passi non possono essere separati logicamente.</p> <p>No. La produzione di un report di errori senza la transazione di accettazione e caricamento dei dati dell'impiegato non è una funzionalità autoconsistente. Il Report di errori non può essere prodotto indipendentemente dalla transazione di accettazione e validazione dei dati.</p> <p>No. La produzione di un report di errori senza la transazione di accettazione non lascia l'applicazione in uno stato di coerenza funzionale. Affinchè l'applicazione sia in uno stato di coerenza funzionale occorre che sia soddisfatto il requisito utente, nulla più. Per soddisfare il requisito utente occorre che la produzione del report di errori sia prodotto dalla transazione di accettazione e caricamento dei dati dell'impiegato</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>Tutti i criteri sopracitati non sono soddisfatti.</p>

**Conclusione**

La produzione di un report di errori senza la transazione di accettazione non soddisfa i requisiti di un processo elementare.

**Produzione di una notifica via email**

Determinare se la produzione di una notifica via email senza la transazione di accettazione e caricamento dei dati dell'impiegato costituisca un Processo Elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Sì. La produzione di una notifica via email con un sunto dei dati è parte dei requisiti funzionali dell'utente.</p> <p>No. La produzione di una notifica via email è parte del processo di aggiornamento e di statistica e non può essere fatta al di fuori di tale contesto. Questi passi non possono essere separati logicamente.</p> <p>No. La produzione di una notifica via email senza la transazione di accettazione e caricamento dei dati dell'impiegato non è una funzionalità autoconsistente. Il processo statistico non può essere fatto indipendentemente dalla transazione di accettazione e caricamento dei dati dell'impiegato.</p> <p>No. La produzione di una notifica via email senza la transazione di accettazione e caricamento dei dati dell'impiegato non lascia l'applicazione in uno stato di coerenza funzionale. Affinchè l'applicazione sia in uno stato di coerenza funzionale occorre che sia soddisfatto il requisito utente, nulla più. Per soddisfare il requisito utente occorre che La produzione di una notifica via email sia eseguita dalla transazione di accettazione e caricamento dei dati dell'impiegato.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>Tutti i criteri sopracitati non sono soddisfatti.</p>

**Conclusione**

La produzione di una notifica via email senza la transazione di accettazione e caricamento dei dati dell'impiegato non soddisfa i requisiti di un processo elementare

**Processo di caricamento dato, produzione report di errori e notifica via email**

Determinare se la transazione di accettazione e caricamento dei dati dell'impiegato, la produzione di un report di errori e la produzione di una notifica via email costituisca un Processo Elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Sì. La transazione di accettazione e caricamento dei dati dell'impiegato, la produzione di un report di errori e la produzione di una notifica via email sono parte dei requisiti funzionali dell'utente.</p> <p>Sì. La transazione di accettazione e caricamento dei dati dell'impiegato, la produzione di un report di errori e la produzione di una notifica via email è una transazione completa. Questi passi non possono essere separati logicamente.</p> <p>Sì. La transazione di accettazione e caricamento dei dati dell'impiegato, la produzione di un report di errori e la produzione di una notifica via email è una funzionalità autoconsistente.</p> <p>Sì. La transazione di accettazione e caricamento dei dati dell'impiegato, la produzione di un report di errori e la produzione di una notifica via email lascia l'applicazione in uno stato di coerenza funzionale. Per soddisfare il requisito utente tutti i passi devono essere eseguiti nella transazione di accettazione e caricamento dei dati dell'impiegato.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>Tutti i criteri sono soddisfatti.</p>

**Conclusione**

La transazione di accettazione e caricamento dei dati dell'impiegato, la produzione di un report di errori e la produzione di una notifica via email soddisfano tutti insieme i requisiti utente.

La transazione di accettazione e caricamento dei dati dell'impiegato, la produzione di un report di errori e la produzione di una notifica via email è un solo processo elementare. Se l'accettazione dati ed il caricamento non sono eseguiti i passi "Produzione report di errori" e "Produzione notifica via email" sono irrilevanti. Tutti i passi del processo devono essere eseguiti per completare il processo elementare e lasciare l'applicazione in uno stato di coerenza funzionale.

### Determinare l'univocità del processo elementare

La seguente tabella illustra le analisi per determinare se il processo elementare sia unico

Determinare l'univocità del Processo Elementare	Si applica la regola?
<p>1. Dopo il confronto con un altro Processo Elementare (EP) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi:</p> <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>No. L'insieme dei DET della transazione di accettazione e caricamento dei dati dell'impiegato è differente da quello di qualunque altro PE.</p> <p>No. L'insieme dei FTR della transazione di accettazione e caricamento dei dati dell'impiegato è differente da quello di qualunque altro PE.</p> <p>No. L'insieme dei trattamenti logici della transazione di accettazione e caricamento dei dati dell'impiegato è differente da quello di qualunque altro PE.</p>
<p>2. Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.</p>	<p>Come precedentemente discusso, suddividere un processo elementare non è appropriato.</p>

### Conclusione

Il processo elementare “Transazione di accettazione e caricamento dei dati dell'impiegato” è unico rispetto agli altri processi elementari identificati.

## Esempio: Assegnazione dell'impiegato ad un lavoro

### Requisiti utente

Assegnazione di un impiegato ad un lavoro. Assegnare un lavoro tramite l'inserimento di queste informazioni per ogni lavoro e impiegato:

- Data effettiva
- Salario
- Punteggio di performance

Al fine di agevolare le assegnazioni, l'utente ha richiesto una lista di assegnazioni lavoro per impiegato e/o una lista dei lavori assegnati con gli impiegati.

### Finestra della lista degli assegnamenti degli impiegati

Questa finestra mostra la lista degli impiegati e dei lavori associati per ogni impiegato.

Sistema Risorse Umane																																																							
Impiegato	Lavori	Assegnazioni	Sedi																																																				
Prospetti																																																							
Sicurezza																																																							
Help																																																							
<table border="1"> <thead> <tr> <th colspan="4">Assegnazioni agli Impiegati</th> </tr> <tr> <th>Cognome</th> <th>Nome</th> <th>MI</th> <th>TipoLavoro</th> </tr> </thead> <tr> <td>Keller</td> <td>Caroline</td> <td></td> <td>RD21379305</td> </tr> <tr> <td>Keller</td> <td>Nicky</td> <td>A</td> <td>RD21379305</td> </tr> <tr> <td>Latta</td> <td>Nicky</td> <td>A</td> <td>RD90379305</td> </tr> <tr> <td>Manship</td> <td>Mark</td> <td>J</td> <td>RD14379305</td> </tr> <tr> <td>Manship</td> <td>Mark</td> <td>J</td> <td>RD15379305</td> </tr> <tr> <td>Schmidt -Taylor</td> <td>Kathleen</td> <td></td> <td>RD14379305</td> </tr> <tr> <td>Smith</td> <td>David</td> <td>E</td> <td>RD16379305</td> </tr> <tr> <td>Smith</td> <td>Loretta</td> <td>M</td> <td>RD16379305</td> </tr> <tr> <td>Waldenson</td> <td>Pat</td> <td></td> <td>RD90379305</td> </tr> <tr> <td>White</td> <td>William</td> <td>W</td> <td>RD21379305</td> </tr> <tr> <td>White</td> <td>William</td> <td>W</td> <td>RD90379305</td> </tr> </table>				Assegnazioni agli Impiegati				Cognome	Nome	MI	TipoLavoro	Keller	Caroline		RD21379305	Keller	Nicky	A	RD21379305	Latta	Nicky	A	RD90379305	Manship	Mark	J	RD14379305	Manship	Mark	J	RD15379305	Schmidt -Taylor	Kathleen		RD14379305	Smith	David	E	RD16379305	Smith	Loretta	M	RD16379305	Waldenson	Pat		RD90379305	White	William	W	RD21379305	White	William	W	RD90379305
Assegnazioni agli Impiegati																																																							
Cognome	Nome	MI	TipoLavoro																																																				
Keller	Caroline		RD21379305																																																				
Keller	Nicky	A	RD21379305																																																				
Latta	Nicky	A	RD90379305																																																				
Manship	Mark	J	RD14379305																																																				
Manship	Mark	J	RD15379305																																																				
Schmidt -Taylor	Kathleen		RD14379305																																																				
Smith	David	E	RD16379305																																																				
Smith	Loretta	M	RD16379305																																																				
Waldenson	Pat		RD90379305																																																				
White	William	W	RD21379305																																																				
White	William	W	RD90379305																																																				
<div> <div>Vista</div> <div>Nuovo</div> <div>Modifica</div> <div>Cancella</div> </div>																																																							

#### AE-1 Lista Assegnazione Impiegati

Vista	Inizia la visualizzazione dei dati su AE-4 Dati su assegnazioni lavoro
Nuovo	Ritorna una form pulita a AE-3 Setup Assegnaz. Lavoro (dati default già inseriti)
Modifica	Inizia la visualizzazione dei dati su AE-5 Modifica Assegnazione Lavoro
Cancella	Rimuove i lavori assegnati (previa conferma)

**Finestra dell'assegnazione del lavoro**

La finestra mostra l'assegnazione di un impiegato a un lavoro (dall'impiegato).

**Sistema Risorse Umane**

Impiegato Lavori Assegnazioni Sedi Prospetti Sicurezza Help

**Assegnazioni impiegati**

**Impostazione Assegnazione Lavoro**

Nome: Mark MI: J Cognome: Manship

Codice Fiscale: 345-67-8901

Sede: Main Plant Tipo di contratto: UFPCA

Numero lavoro: [dropdown] [input]

Data: [input]

Salario: [input]

Livello performance: [input]

OK Annulla

**AE-3 Impostazione Assegnazione Lavoro**

- OK** Salva le nuove assegnazioni e visualizza a AE-1 lista delle assegnazioni impiegati
- Annulla** Ignora i dati immessi , ritorna a AE-1-lista delle assegnazioni impiegati

Se l'utente non inserisce correttamente i dati, un messaggio di errore è visualizzato.

**Finestra Lista assegnazioni lavori**

La finestra mostra la lista dei lavori e degli impiegati assegnati.

**Sistema Risorse Umane**

Impiegato Lavori Assegnazioni Sedi Prospetti Sicurezza Help

**Assegnazioni agli Impiegati**

ID Lavoro	Cognome	Nome	MI
RD14379305	Manship	Mark	J
RD14379305	Schmidt-Taylor	Kathleen	J
RD15379305	Manship	Mark	J
RD16379305	Smith	David	E
RD16379305	Smith	Loretta	M
RD21379305	Keller	Caroline	A
RD21379305	Latta	Nicky	W
RD21379305	White	William	A
RD90379305	Latta	Nicky	A
RD90379305	Waldenson	Pat	W
RD90379305	White	William	W

Vista Nuovo Modifica Cancella

**AJ-1 Lista delle Assgnazioni di Lavoro**

- Vista** Mostra dati dalla schermata AJ-4 "Dati Assegnazioni Impiegati"
- Nuovo** Presenta una nuova schermata su AJ-3 "Setup Assegnazioni Impiegati" (dati di default)
- Modifica** Mostra dati dalla schermata AJ-5 "Modifica Assegnazioni Impiegati"
- Cancella** Rimuove gli inserimenti di assegnazioni lavori agli impiegati (a seguito di conferma)



**Finestra  
dell'assegnazione  
dell'impiegato a  
un lavoro**

La finestra mostra l'assegnazione di un impiegato ad un lavoro (dal lavoro).

The screenshot shows a window titled "Sistema Risorse Umane" with a menu bar: "Impiegato", "Lavori", "Assegnazioni", "Sedi", "Prospetti", "Sicurezza", "Help". Inside, there's a sub-window titled "Assegnazioni di lavoro" with a menu bar: "Impostazione Assegnazioni lavoro". Below this, there's a form for assigning an employee to a job. The form includes a text box with "RD14379305" and a button "Edizione di Maggio - Stampa Test". The form fields are: "Cognome", "Nome", "MI", "Codice Fiscale" (with a hyphen), "Sede", "Contratto di Lavoro", "Data", "Stipendio", and "Valutaz. performance". There are "OK" and "Annulla" buttons at the bottom right of the form.

AJ-3 Settaggio Assegnazione Impiegato

- OK** Salva la nuova assegnazione per l'impiegato
- Annulla** Ignora i dati inseriti, ritorna alla schermata AJ-1 Lista Assegnazioni Lavori

Se l'utente non inserisce correttamente i dati, un messaggio di errore è visualizzato.

**Lista  
assegnazioni  
impiegati**

Determinare se la Lista Assegnazioni Impiegati sia un processo elementare La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Sì. La lista Assegnazioni Impiegati è parte dei requisiti funzionali dell'utente.</p> <p>Sì. La lista Assegnazioni Impiegati è una transazione completa. Questa è una funzionalità logicamente indipendente.</p> <p>Sì. La lista Assegnazioni Impiegati è una funzionalità autoconsistente. La lista Assegnazioni Impiegati è eseguita indipendentemente dall'assegnare un impiegato a un lavoro.</p> <p>Sì. La lista Assegnazioni Impiegati lascia l'applicazione in uno stato di coerenza funzionale. I requisiti utente funzionali sono soddisfatti senza dover fare nulla più.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>La lista Assegnazioni Impiegati soddisfa tutti i criteri.</p>

**Conclusione**

La lista Assegnazioni Impiegati soddisfa i requisiti per essere un processo elementare.

**Assegnazione  
impiegato ad  
un lavoro  
(dall'impiegato)**

Determinare se l'Assegnazione di un impiegato al lavoro(dal lato impiegato) sia un processo elementare La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. L'Assegnazione di un impiegato al lavoro(dal lato impiegato) è parte dei requisiti funzionali dell'utente</p> <p>Si. L'Assegnazione di un impiegato al lavoro(dal lato impiegato) è una transazione completa Questa è una funzionalità logicamente indipendente.</p> <p>Si. L'Assegnazione di un impiegato al lavoro(dal lato impiegato) è una funzione autoconsistente l'Assegnazione di un impiegato al lavoro(dal lato impiegato) è eseguita indipendentemente dalla Lista Assegnazioni. Impiegati.</p> <p>Si. L'Assegnazione di un impiegato al lavoro(dal lato impiegato) lascia l'applicazione in uno stato di coerenza funzionale. I requisiti utente funzionali sono soddisfatti senza dover fare nulla più.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>L'Assegnazione di un impiegato al lavoro(dal lato impiegato) soddisfa tutti i criteri.</p>

**Conclusione**

L'assegnazione di un impiegato al lavoro (dal lato impiegato) soddisfa i requisiti di un processo elementare.

**Lista  
assegnazione  
lavori**

Determinare se la Lista Assegnazioni sia un processo elementare.

La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. La Lista Assegnazioni Lavoro è parte dei requisiti funzionali dell'utente.</p> <p>Si. La Lista Assegnazioni Lavoro è una transazione completa. Questa è una funzionalità logicamente indipendente.</p> <p>Si. La Lista Assegnazioni Lavoro è una funzionalità autoconsistente. La lista Assegnazioni Impiegati è eseguita indipendentemente dall'assegnare un impiegato a un lavoro.</p> <p>Si La Lista Assegnazioni Lavoro lascia l'applicazione in uno stato di coerenza funzionale. I requisiti utente funzionali sono soddisfatti senza dover fare nulla più.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>La Lista Assegnazioni Lavoro soddisfa tutti I criteri.</p>

**Conclusione** La Lista Assegnazioni Lavoro soddisfa i requisiti di un processo elementare.

**Assegnazione  
impiegato ad  
un lavoro (dal  
lavoro)**

Determinare se l'Assegnazione di un impiegato al lavoro(dal lato lavoro) sia un processo elementare La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. L'Assegnazione di un impiegato al lavoro (dal lato lavoro) è parte dei requisiti funzionali dell'utente.</p> <p>Si. L'Assegnazione di un impiegato al lavoro(dal lato lavoro) è una transazione completa Questa è una funzionalità logicamente indipendente.</p> <p>Si. L'Assegnazione di un impiegato al lavoro(dal lato lavoro) è una funzione autoconsistente l'Assegnazione di un impiegato al lavoro(dal lato lavoro) è eseguita indipendentemente dalla Lista Assegnazioni Lavoro.</p> <p>Si. L'Assegnazione di un impiegato al lavoro(dal lato lavoro) lascia l'applicazione in uno stato di coerenza funzionale. I requisiti utente funzionali sono soddisfatti senza dover fare nulla più.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>L'Assegnazione di un impiegato al lavoro (dal lato lavoro) soddisfa tutti i criteri.</p>

**Conclusione**

L'Assegnazione di un impiegato al lavoro (dal lato lavoro) soddisfa i requisiti di un processo elementare.

### Determinare l'univocità del processo elementare

La seguente tabella illustra le analisi per determinare se i processi elementari Lista Assegnazioni Impiegato e Lista Assegnazioni Lavoro siano univochi.

Determinare l'univocità del Processo Elementare	Si applica la regola?
<p>1. Dopo il confronto con un altro Processo Elementare (EP) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi:</p> <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>Si. L'insieme dei DET per Lista Assegnazioni Impiegati è lo stesso che l'insieme dei DET per Lista Assegnazioni Lavoro.</p> <p>Si. L'insieme dei FTR per Lista Assegnazioni Impiegati è lo stesso che l'insieme dei FTR per Lista Assegnazioni Lavoro.</p> <p>Si. L'insieme dei trattamenti logici per Lista Assegnazioni Impiegati è lo stesso che l'insieme dei trattamenti logici per Lista Assegnazioni Lavoro. La sola differenza tra loro consiste nella disposizione dei campi e nell'ordine di visualizzazione delle righe. Come stabilito nelle Forme di Processo Logico differenze nella disposizione o nell'ordinamento non sono tali da costituire PE univoci.</p>
<p>2. Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.</p>	<p>Non vi è niente da scomporre.</p>

### Conclusione

Lista Assegnazioni Impiegato e Lista Assegnazioni Lavoro rappresenta un singolo e univoco processo elementare.

### Determinare l'univocità del processo elementare

La seguente tabella mostra l'analisi per determinare se i processi elementari Assegnazione di un impiegato al lavoro(dal lato Impiegato) e Assegnazione di un impiegato al lavoro(dal lato lavoro) siano univoci.

Determinare l'univocità del Processo Elementare	Si applica la regola?
<p>1. Dopo il confronto con un altro Processo Elementare (EP) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi:</p> <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>Si. L'insieme dei DET per Assegnazione di un impiegato al lavoro(dal lato Impiegato) è lo stesso che l'insieme dei DET per Assegnazione di un impiegato al lavoro(dal lato lavoro).</p> <p>Si. L'insieme dei FTR per Assegnazione di un impiegato al lavoro(dal lato Impiegato) è lo stesso che l'insieme dei FTR per Assegnazione di un impiegato al lavoro(dal lato lavoro) .</p> <p>Si. L'insieme dei FTR per Assegnazione di un impiegato al lavoro(dal lato Impiegato) è lo stesso che l'insieme dei FTR per Assegnazione di un impiegato al lavoro(dal lato lavoro). La sola differenza tra loro consiste nella disposizione dei campi . Come stabilito nelle Forme di Processo Logico differenze nella disposizione non sono tali da costituire PE univoci.</p>
<ul style="list-style-type: none"> <li>• Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.</li> </ul>	Non vi è niente da scomporre.

**Conclusione** Due processi elementari univoci sono identificati:

- Lista Assegnazioni (da Impiegato) e/o Lista Assegnazioni (da Lavoro)
- Assegnazione di un impiegato al lavoro(dal lato Impiegato) e/o Assegnazione di un impiegato al lavoro(dal lato lavoro)

Quando due processi elementari simili sono confrontati ed è stabilito che essi contengono lo stesso insieme di DET, FTR e trattamenti logici, essi sono identificati come un singolo ed unico processo elementare.

## Esempio: Processi elementari simili

### Requisiti utente

L'utente richiede due stampe molto simili relativamente alle informazioni di un impiegato.

Una delle due stampe includerà anche l'informazione dell'indirizzo email dell'impiegato, essa sarà infatti data al personale di staff dell'azienda al fine di comunicare via email piuttosto che via telefono.

Si vedano gli esempi di dettaglio per ognuna delle due stampe.

Tutti i dati delle stampe provengono dallo stesso file logico, ad eccezione del dato relativo all'indirizzo email che appartiene ad un differente file logico mantenuto all'interno della applicazione.

### Lista Impiegati (con indirizzo email)

#### Elenco degli Impiegati

ID #	Nome	Tel. Casa	Tel. Lavoro	Sede	Email
894253	Jay Smith	888-555-1212	999-666-0202	Stabilimento A	Jsmith@USACorp.com
999524	Mel Dunn	777-258-489	999-125-3589	Stabilimento B	MDunn@USACorp.com
125896	Val Jones	555-678-6892	999-258-4789	Stabilimento C	VJones@USACorp.com

### Lista Impiegati (senza indirizzo email)

#### Elenco degli Impiegati

ID #	Nome	Tel. Casa	Tel. Lavoro	Sede
894253	Jay Smith	888-555-1212	999-666-0202	Stabilimento A
999524	Mel Dunn	777-258-489	999-125-3589	Stabilimento B
125896	Val Jones	555-678-6892	999-258-4789	Stabilimento C



**Lista Impiegati  
(senza indirizzo  
email)**

Determinare se la Lista Impiegati (senza indirizzo email) sia un processo elementare. La seguente tabella mostra l'analisi.

Determinare l'univocità del processo elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"> <li>• è significativa per l'utente.</li> <li>• costituisce una transazione completa</li> <li>• è autonoma e</li> <li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li> </ul>	<p>Si. La Lista Impiegati (senza indirizzo email) è un requisito funzionale dell'utente.</p> <p>Si. La Lista Impiegati (senza indirizzo email) è una transazione completa che soddisfa un gruppo di utenti.</p> <p>Si. La Lista Impiegati (senza indirizzo email) è di per se stessa autoconsistente.</p> <p>Si. L'applicazione, quando la Lista Impiegati (senza indirizzo email) è creata, è lasciata in uno stato di coerenza funzionale.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>Si. La Lista Impiegati (senza indirizzo email) soddisfa tutti i criteri sopra citati.</p>

**Conclusione**

La Lista Impiegati (senza indirizzo email), soddisfa i requisiti di un processo elementare.

**Determinare l'univocità del processo elementare**

La seguente tabella illustra le analisi per determinare se il processo elementare sia unico

Determinare l'univocità del processo elementare	Si applica la regola?
<p>1. Dopo il confronto con un altro Processo Elementare (EP) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi:</p> <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>No. L'insieme dei DET per Lista Impiegati (senza indirizzo email) richiede un differente insieme di DET rispetto a Lista Impiegati (con indirizzo email).</p> <p>No. L'insieme dei FTR per Lista Impiegati (senza indirizzo email) richiede un differente insieme di FTR rispetto a Lista Impiegati (con indirizzo email).</p> <p>No. Lista Impiegati (senza indirizzo email) ha un differente insieme di trattamenti logici rispetto a Lista Impiegati (con indirizzo email).</p>
<p>1. Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.</p>	<p>Non vi è niente da scomporre.</p>

**Conclusione**

Il processo elementare Lista Impiegati (senza indirizzo email) è univoco rispetto agli altri processi elementari identificati.

**Lista Impiegati  
(con indirizzo  
email)**

Determinare se Lista Impiegati (con indirizzo email) costituisca un Processo Elementare. La seguente tabella illustra l'analisi.

Regole conteggio Processo Elementare	Si applica la regola?
<p>1. Componi e/o decomponi i Requisiti Funzionali Utente (FUR) nella più piccola unità di attività, che soddisfi tutti i seguenti punti:</p> <ul style="list-style-type: none"><li>• è significativa per l'utente.</li><li>• costituisce una transazione completa</li><li>• è autonoma e</li><li>• lascia l'applicazione oggetto di conteggio in uno stato di coerenza funzionale</li></ul>	<p>Sì. La Lista Impiegati (con indirizzo email) è un requisito funzionale dell'utente.</p> <p>Sì. La Lista Impiegati (con indirizzo email) è una transazione completa che soddisfa un gruppo di utenti.</p> <p>Sì. La Lista Impiegati (con indirizzo email) è di per se stessa autoconsistente.</p> <p>Sì. L'applicazione, quando la Lista Impiegati (con indirizzo email) è creata, è lasciata in uno stato di coerenza funzionale.</p>
<p>2. Identifica un processo elementare per ogni unità di attività identificata che soddisfi tutti i criteri sopraindicati.</p>	<p>Sì. La Lista Impiegati (con indirizzo email) soddisfa tutti i criteri sopra citati.</p>

**Conclusione**

La Lista Impiegati (con indirizzo email), soddisfa i requisiti di un processo elementare.

### Determinare l'univocità del processo elementare

La seguente tabella illustra le analisi per determinare se il processo elementare è unico

Determinare l'univocità del processo elementare	Si applica la regola?
1. Dopo il confronto con un altro Processo Elementare (EP) già identificato, contare i due Processi Elementari simili come lo stesso Processo Elementare se questi: <ul style="list-style-type: none"> <li>• Utilizzano lo stesso insieme di DET e</li> <li>• Utilizzano lo stesso insieme di FTR e</li> <li>• Richiedono lo stesso insieme di trattamenti logici per completare il processo elementare</li> </ul>	<p>No. L'insieme dei DET per Lista Impiegati (con indirizzo email) richiede un differente insieme di DET rispetto a Lista Impiegati (senza indirizzo email).</p> <p>No. L'insieme dei FTR per Lista Impiegati (con indirizzo email) richiede un differente insieme di FTR rispetto a Lista Impiegati (senza indirizzo email).</p> <p>No. La Lista Impiegati (con indirizzo email) ha un differente insieme di trattamenti logici rispetto a Lista Impiegati (senza indirizzo email).</p>
1. Non scomporre un processo elementare con differenti modalità di logica elaborativa in processi elementari multipli.	Non vi è niente da scomporre.

**Conclusione** Il processo elementare Lista Impiegati (con indirizzo email) è univoco rispetto a tutti gli altri processi elementari identificati.

Due processi sono in prima istanza analizzati per verificare se soddisfano i criteri di un processo elementare, poi i processi sono confrontati uno con l'altro al fine di verificare se contengono differenti insiemi di DET, FTR e Trattamenti logici.

Quando due processi elementari contengono insiemi differenti di DET, FTR e Trattamenti Logici essi sono identificati come processi elementari separati se sono specificati dall'utente come requisiti funzionali distinti.

In questo caso sono identificati due processi elementari:

- Lista Impiegati (senza indirizzo email) e
- Lista Impiegati (con indirizzo email)

---

## EI: Esempi di conteggio

---

**Introduzione** Questa sezione utilizza un'applicazione Risorse Umane (RU) per illustrare le procedure di conteggio degli external input (EI). In aggiunta a questa sezione, ulteriori esempi sono presentati nei Casi di Studio.

**Contenuti** Questa sezione include i seguenti esempi:

Argomento	Pagina
Descrizioni sintetiche degli esempi di conteggio di EI	2-62
Esempio: Informazioni di controllo	2-63
Esempio: Input a video	2-67
Esempio: Batch con EI multipli ed EI duplicati	2-70
Esempio: Correzione delle transazioni sospese	2-73
Esempio: EI con tipi di file referenziati (FTR) multipli	2-76
Esempio: Conversione di dati	2-80
Esempio: Riferimento a dati di un'altra applicazione	2-83
Esempio: EI con output a video – 1	2-85
Esempio: EI con output a video – 2	2-88
Esempio: EI con attributi ricavati da un EIF	2-91
Esempio: EI cancellazione	2-97
Esempio: Aggiungere parametri di sicurezza	2-100

## Descrizioni sintetiche degli esempi di conteggio di EI

Nella seguente tabella sono elencati e descritti gli esempi per gli EI.

Esempio	Descrizione Sintetica	Pagina
Informazioni di controllo	Questo esempio riguarda le informazioni di controllo usate per la produzione di report.	2-63
Input a video	Questo esempio illustra il conteggio di una transazione di inserimento interattivo.	2-67
Batch con EI multipli ed EI duplicati	Questo esempio spiega come contare un file di transazione con tipi di record multipli o formattati.	2-70
Correzione delle transazioni sospese	Questo esempio illustra il conteggio delle correzioni apportate ai lavori sospesi durante l'elaborazione batch di inserimento o variazione lavori.	2-73
EI con tipi di file referenziati multipli	Questo esempio illustra l'uso di un diagramma di flusso dei dati per contare un EI che referencia file multipli.	2-76
Conversione di dati	Questo esempio spiega come contare il processo di conversione di un gruppo di dati in un diverso formato con elementi di dati addizionali.	2-80
Riferimento a dati di un'altra applicazione	Questo esempio spiega perché un file d'interfaccia esterno (discusso nella Parte 2, Capitolo 6) non è contato come EI.	2-83
EI con output a video -1	Questo esempio illustra un EI con un campo calcolato che viene visualizzato.	2-85
EI con output a video -2	Questo esempio illustra un EI con un campo calcolato e con due EQ incluse.	2-88
EI con attributi ricavati da un eif	Questo esempio illustra un EI con attributi ricavati da un EIF che non attraversano il confine dell'applicazione.	2-91
EI di cancellazione	Questo esempio illustra il conteggio dei DET per una transazione di Cancellazione.	2-97
Aggiunta di parametri di sicurezza	Questo esempio illustra il conteggio delle funzionalità per mantenere la sicurezza di un sistema.	2-100

## Esempio: Informazioni di controllo

### Requisiti utente

L'utente richiede la possibilità di controllare come e quando i report delle assegnazioni vengono stampati. La seguente lista illustra gli specifici requisiti utente per generare il report.

1. Controllo dei seguenti aspetti dell'elaborazione del report:
  - sequenza di ordinamento,
  - porta della stampante,
  - possibilità di generare o non generare una copia su microfiche,
  - possibilità di generare o non generare una copia cartacea.
2. Salvataggio dei controlli per la stampa delle assegnazioni lavori.
3. Effettuazione e salvataggio delle variazioni.
4. Invio di un messaggio per confermare che i controlli per i report delle assegnazioni sono stati aggiunti o variati e che il report è stato generato.

**Nota:** Questo esempio illustra solo il requisito per inserire il gruppo di informazioni di controllo per il report delle assegnazioni. Il Caso di Studio illustra il conteggio di tutti i requisiti utente.

### Finestra di esempio

La seguente finestra Report Assegnazioni Lavori è utilizzata per stabilire i controlli per la stampa dei report delle assegnazioni.

The screenshot shows a window titled "Sistema Risorse Umane" with a menu bar containing "Impiegato", "Lavori", "Assegnazioni", "Sedi", and "Help". The "Assegnazioni" menu is active, displaying a sub-window titled "Prospetto Assegnazione Lavori".

Inside the "Prospetto Assegnazione Lavori" window, there are three main sections:

- Sequenza Ordinamento:** A list box showing the order of sorting: "3 ID Lavoro", "2 Cod Impiegato", and "1 Nome Impiegato". Below it, the text "Identificare con 1, 2, 3" is displayed.
- Porta Stampante:** A list box showing the printer port: "( ) LPT 1", "( ) LPT 2", and "( ) LPT 3".
- Generazione Copie:** Two checkboxes: "Genera Copia su Microfiche" (unchecked) and "Genera Copia Cartacea" (checked).

At the bottom right of the dialog box are three buttons: "OK", "Annulla", and "Ripristina".

JR-1

- |                   |                              |
|-------------------|------------------------------|
| <b>OK</b>         | Elabora richiesta prospetti  |
| <b>Annulla</b>    | Torna al menu a tendina      |
| <b>Ripristina</b> | Ripristina valori precedenti |

**Passo 1 Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

**Passo 2 Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--

**Passo 3 Classifica il Processo Elementare**

Regole di conteggio dell'EI	Si applica la regola?
1. Il suo intento primario è di <ul style="list-style-type: none"> <li>Mantenere uno o più ILF oppure</li> <li>alterare il comportamento del sistema</li> </ul>	Sì. I dati che entrano nell'applicazione saranno eventualmente usati come dati di controllo. Si tratta di dati applicativi memorizzati nell'ILF Controllo Report.  No. Il comportamento dell'applicazione non è alterato.
2. Include processi logici per l'inserimento dei dati o informazioni di controllo che attraversano il confine dell'applicazione.	Sì. Le informazioni di controllo attraversano il confine dell'applicazione.

**Conclusione** Si è stabilito che le informazioni di controllo dei report sono un EI.

**Passo 4 Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	Il Report di Controllo ILF è mantenuto. E' conteggiato una sola volta.



**Passo 5****Conteggia i Tipi di Elementi Dati (DET)**

<b>Regole di conteggio dei DET</b>	<b>Si applica la regola?</b>
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Sequenza di ordinamento, Porta Stampante, Formato di Output.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Messaggio utente.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Pulsante OK.
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• Costanti quali titoli dei report, identificativi delle schermate o dei pannelli, intestazione delle colonne o descrizione degli attributi</li> <li>• Notazioni generate automaticamente dall'applicazione quali data e orario</li> <li>• Numeri di pagina variabili o informazioni riguardanti il posizionamento (ad esempio Righe da 37 a 54 di 211)</li> <li>• Aiuti navigazionali quali la possibilità di movimento all'interno di una lista usando "precedente", "prossimo", "primo", "ultimo" e oggetti grafici equivalenti.</li> <li>• Attributi generati all'interno del confine da una funzione transazionale e salvati all'interno di un ILF che non escono dal confine stesso.</li> <li>• Attributi reperiti o referenziati da un ILF o EIF necessari alla elaborazione ma che non escono dal confine.</li> </ul>	Nessun testo è conteggiato.  Non esistono elementi di questo tipo.  Non esistono elementi di questo tipo.  Non esistono elementi di questo tipo.  Non esistono elementi di questo tipo.  Non esistono elementi di questo tipo.

**Passo 6**      **Determina la complessità funzionale**

1 FTR e 5 DET	La complessità è Bassa.
---------------	-------------------------

**Passo 7**      **Determina la dimensione funzionale**

La dimensione funzionale di 1 EI a Bassa complessità	3 FP
---	------

## Esempio: Input a video

### Requisiti utente

L'utente richiede la possibilità di:

- inserire le informazioni sul lavoro in modo interattivo;
- generare un messaggio di errore ed evidenziare i campi errati, in modo che l'errore può essere corretto interattivamente;
- salvare le informazioni inserite sul lavoro.

### Schermata di esempio

La seguente schermata Dati sul Lavoro è usata per inserire un nuovo lavoro.

```

Azione:_ 7=Precedente 8=Successivo 9=Salva
                        Dati sul Lavoro

Numero Lavoro:   RD15379305
Nome Lavoro:     Edizione di Maggio - Stampa delle Copertine
Liv. Retrib.:    JRNY05A

N. Linea         Descrizione Lavoro
01               Stampa delle copertine 4-Up, Patinatura fine
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

F1=Help F7=Pag. Prec. F8=Pag. Succ. F12=Annulla
  
```

Invio: ritorna alla schemata di partenza.	F1: mostra help a livello campo o finestra.
Action 7: mostra dati precedente lavoro, se presente.	F7: scorre di 10 righe su la descrizione.
Action 8: mostra dati successivo lavoro, se presente.	F8: scorre di 10 righe giù la descrizione.
Action 9: salva i dati di lavoro inseriti.	F12: ritorna alla schemata di partenza.

**Passo 1**      **Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

**Passo 2**      **Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità
--	---

**Passo 3**      **Classifica il Processo Elementare**

Regole di conteggio dell'EI	Si applica la regola?
1. Il suo intento primario è di <ul style="list-style-type: none"> <li>Mantenere uno o più ILF oppure</li> <li>alterare il comportamento del sistema</li> </ul>	Sì. L'ILF JOB è mantenuto. No. Il comportamento dell'applicazione non è alterato.
2. Include processi logici per l'inserimento dei dati o informazioni di controllo che attraversano il confine dell'applicazione.	Sì. Le informazioni del JOB attraversano il confine dell'applicazione.

**Conclusione**      Aggiungere un nuovo JOB è un EI.

Fare riferimento ai Casi di Studio per vedere come sono conteggiati i requisiti di variazione e di cancellazione e le schermate associate.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	L'ILF JOB è mantenuto e letto, ma viene conteggiato una sola volta.

**Passo 5 Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Numero Lavoro, Nome Lavoro, Livello Retributivo, Descrizione Lavoro (ripetuto). Numero Linea Descrizione Lavoro esiste solo per ragioni tecniche e non dovrebbe essere contato come DET.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Messaggi di errore.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Tasto Funzione 9 (ADD).
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Testi quali "Numero Lavoro" non sono conteggiati.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>F7 e F8.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6 Determina la complessità funzionale**

1 FTR e 6 DET	La complessità è Bassa.
---------------	-------------------------

**Passo 7 Determina la dimensione funzionale**

La dimensione funzionale di 1 EI a Bassa complessità	3 FP
--	------

## Esempio: Batch con EI multipli ed EI duplicati

### Requisiti utente

L'utente richiede la possibilità di:

- inserire le informazioni sul lavoro in modalità batch.,
- modificare le informazioni sul lavoro in modalità batch.

**Nota:** L'obiettivo di questo esempio è l'inserimento di un lavoro in modalità batch. L'esempio precedente si riferiva alla modalità interattiva. I Casi di Studio illustrano il conteggio di tutti requisiti utente per l'inserimento dei lavori sia in modo interattivo che batch.

### Requisiti di Realizzazione

È stato deciso che durante il processo batch, qualsiasi lavoro non aggiornato con successo sarà inserito in un file Sospesi, da mantenere separatamente (vedi il prossimo esempio).

### Formato Record

Il seguente diagramma riporta il formato del record per questo esempio.

	123456789010123456789012345678901234567890123456789012345678901234567890															
1	ADD	01	SRENG	INGEGNERE SENIOR SISTEMI INFORMATIVI						05						
2	ADD	02	SRENG	01	PARTE DA LIV.RETRIB.05											
3	ADD	02	SRENG	02	ALTRI LIV.RETRIB.:06 E 07											
4	CHG	01	STENG	04												
5	CHG	02	STENG	02	ALTRI LIV.RETRIB.:05 E 06											
6																
7																
8																
9																
0																

### Descrizione dei Record

La seguente tabella include le descrizioni per ciascun tipo record.

Record	Posizione	Descrizione
01	1-3	Tipo Transazione
	4-5	Tipo Record
	6-10	Codice Lavoro
	11-45	Nome Lavoro
	46-47	Livello Retributivo Lavoro
02	1-3	Tipo Transazione
	4-5	Tipo Record
	6-10	Codice Lavoro
	11-12	Progressivo Linea Descrizione Lavoro
	13-41	Linee Descrizione Lavoro

Dove i Tipi Record sono:

- 01 Inserimento record per un nuovo lavoro
- 02 Inserimento record per le descrizioni di un nuovo lavoro

**Passo 1 Identifica il Processo Elementare – Tipo Transazione 01**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	No. Un lavoro senza una descrizione non è significativo per l'utente.
---	---

**Passo 1 Identifica il Processo Elementare – Tipo Transazione 02**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	No. Una descrizione non può esistere senza il lavoro che essa sta descrivendo. I dati sarebbero incoerenti.
---	---

**Passo 1 Identifica il Processo Elementare – Tipo Transazione 1 + 2**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì. Lavoro e descrizione sono significativi per l'utente.
---	---

**Passo 2 Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. La funzione Inserisci Lavoro Batch (Tipo Transazione 1+2) è simile alla funzione Inserisci Lavoro Online del precedente esempio. La funzione Inserisci Lavoro Batch mantiene un ILF addizionale (Lavori sospesi) che la funzione on line non gestisce.
--	--

**Passo 3 Classificare il processo elementare – Tipo Transazione 1 + 2**

Regole di conteggio dell'EI	Si applica la regola?
1. Il suo intento primario è di <ul style="list-style-type: none"> <li>Mantenere uno o più ILF oppure</li> <li>alterare il comportamento del sistema</li> </ul>	<p>Sì. L'intento primario è mantenere l'ILF Lavori.</p> <p>No. Il comportamento dell'applicazione non è alterato.</p>
2. Include processi logici per l'inserimento dei dati o informazioni di controllo che attraversano il confine dell'applicazione.	Sì. La funzione Inserisci lavoro include logiche di controllo e di processo utili alla validazione delle informazioni sul Lavoro.

**Conclusione** Inserisci un lavoro (Tipo Transazione 1 + 2) è un EI.

**Passo 4**

**Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	L'ILF Lavori è mantenuto e letto, ma è conteggiato una sola volta. L'ILF Lavori Sospesi è mantenuto.

**Passo 5**

**Conteggiare i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Numero Lavoro, Nome Lavoro, Livello Retributivo, Descrizione Lavoro (linee ripetute). Tipo Transazione e Numero linea Descrizione Lavoro esistono per ragioni tecniche e non sono conteggiati come DET.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non applicabile. Nessun errore è memorizzato all'interno del file Lavori Sospesi.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Tipo transazione
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo.

**Passo 6**

**Determina la complessità funzionale**

2 FTR e 5 DET	La complessità è Media.
---------------	-------------------------

**Passo 7**

**Determina la dimensione funzionale**

La dimensione funzionale di 1 EI a Media complessità	4 FP
--	------



## Esempio: Correzione delle transazioni sospese

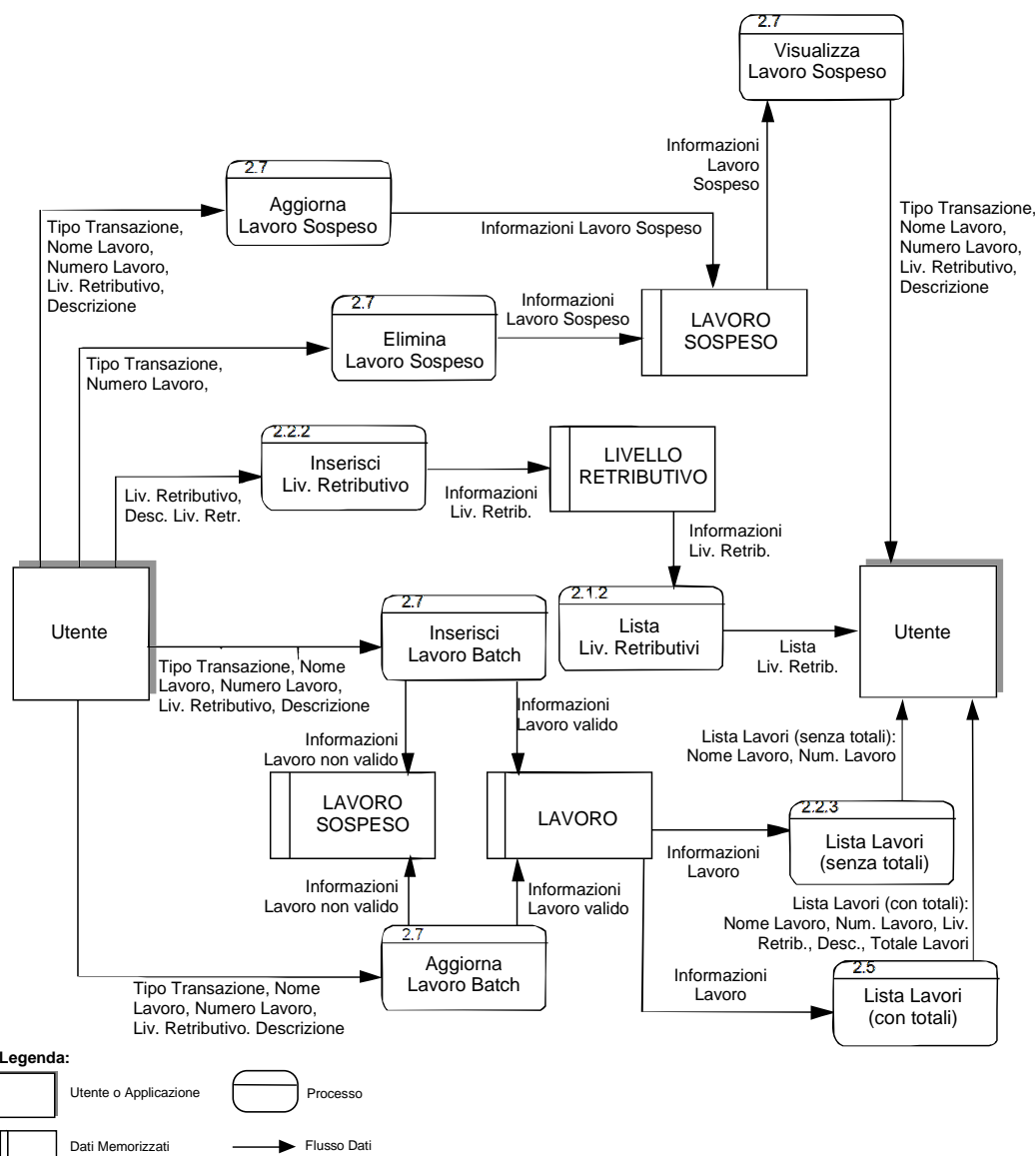
### Requisiti utente

È stato deciso che durante il processo batch qualsiasi lavoro non aggiornato con successo deve essere inserito in un file Sospesi. L'utente richiede una schermata per accedere e correggere le transazioni errate.

**Nota:** Questo esempio esamina solo il requisito di correggere le transazioni sospese. I Casi di Studio illustrano il conteggio dell'intero requisito utente.

### Diagramma di flusso dei dati

Il seguente diagramma illustra il flusso dei dati per questo esempio.



**Passo 1**      **Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì
---	----

**Passo 2**      **Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--

**Passo 3**      **Classifica il Processo Elementare**

Regole di conteggio dell'EI	Si applica la regola?
1. Il suo intento primario è di <ul style="list-style-type: none"> <li>Mantenere uno o più ILF oppure</li> <li>alterare il comportamento del sistema</li> </ul>	<p>Sì. La funzione di correzione dei Lavori sospesi ha l'intento primario di mantenere l'ILF Lavori sospesi.</p> <p>No. Il comportamento dell'applicazione non è alterato.</p>
2. Include processi logici per l'inserimento dei dati o informazioni di controllo che attraversano il confine dell'applicazione.	Sì. La funzione Inserisci Lavoro include logiche di controllo e di processo utili alla validazione delle informazioni sul Lavoro.

**Conclusione**      La transazione Correggere un Lavori sospeso è un EI.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	L'ILF Lavori sospesi è mantenuto e referenziato, ma è conteggiato una sola volta.

**Passo 5****Conteggiare i Tipi di Elementi Dati (DET)**

<b>Regole di conteggio dei DET</b>	<b>Si applica la regola?</b>
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Tipo Transazione, Numero Lavoro, Nome Lavoro, Livello Retributivo Lavoro, Linea Descrizione Lavoro (ripetuta).  Tipo Transazione e Numero Linea Lavoro description esistono per ragioni tecniche e non sono conteggiati come DET.  Tutti gli altri campi sono riconoscibili dall'utente.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono messaggi.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Tasto Invio
4. Non contare I seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	Non esistono elementi di questo tipo.  Non esistono elementi di questo tipo.  Non esistono elementi di questo tipo.  Non esistono elementi di questo tipo.  Non esistono elementi di questo tipo.  Non esistono elementi di questo tipo.

**Passo 6****Determina la complessità funzionale**

1 FTR e 6 DET	La complessità è bassa.
---------------	-------------------------

**Passo 7****Determina la dimensione funzionale**

La dimensione funzionale di 1 EI a Bassa complessità	3 FP
--	------

## Esempio: El con tipi di file referenziati (FTR) multipli

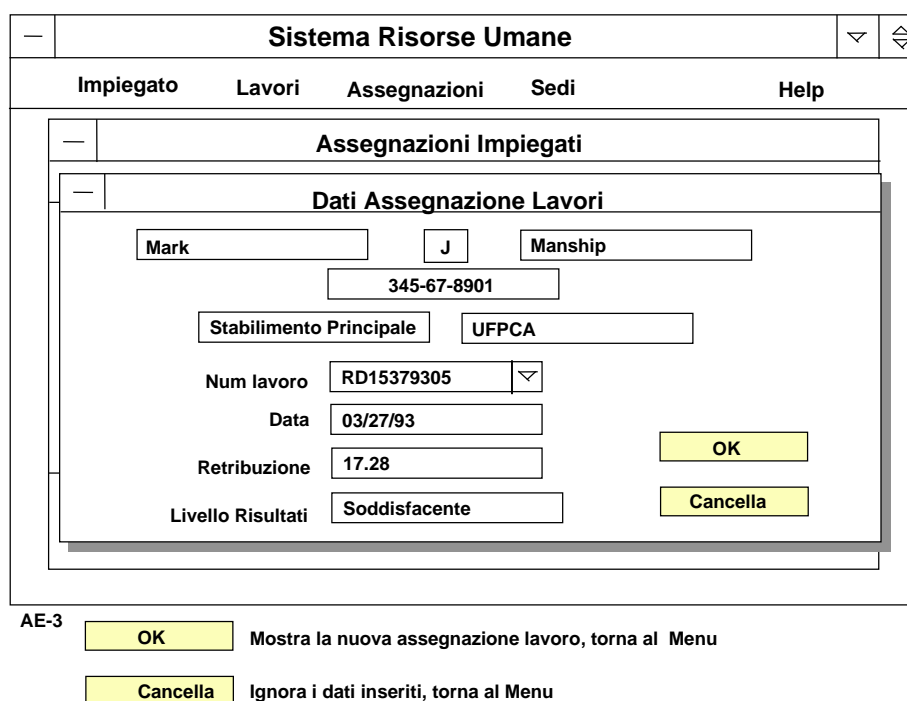
### Requisiti utente

L'utente richiede la possibilità di inserire le assegnazioni lavori.

**Nota:** Questo esempio esamina il requisito di inserire le assegnazioni lavori. I Casi di Studio illustrano il conteggio dell'intero requisito utente.

### Schermata di esempio

La seguente figura illustra un esempio della finestra di inserimento assegnazioni lavori.



**Sistema Risorse Umane**

Impiegato Lavori Assegnazioni Sedi Help

**Assegnazioni Impiegati**

**Dati Assegnazione Lavori**

Mark J Manship

345-67-8901

Stabilimento Principale UFPCA

Num lavoro RD15379305

Data 03/27/93

Retribuzione 17.28

Livello Risultati Soddisfacente

OK

Cancella

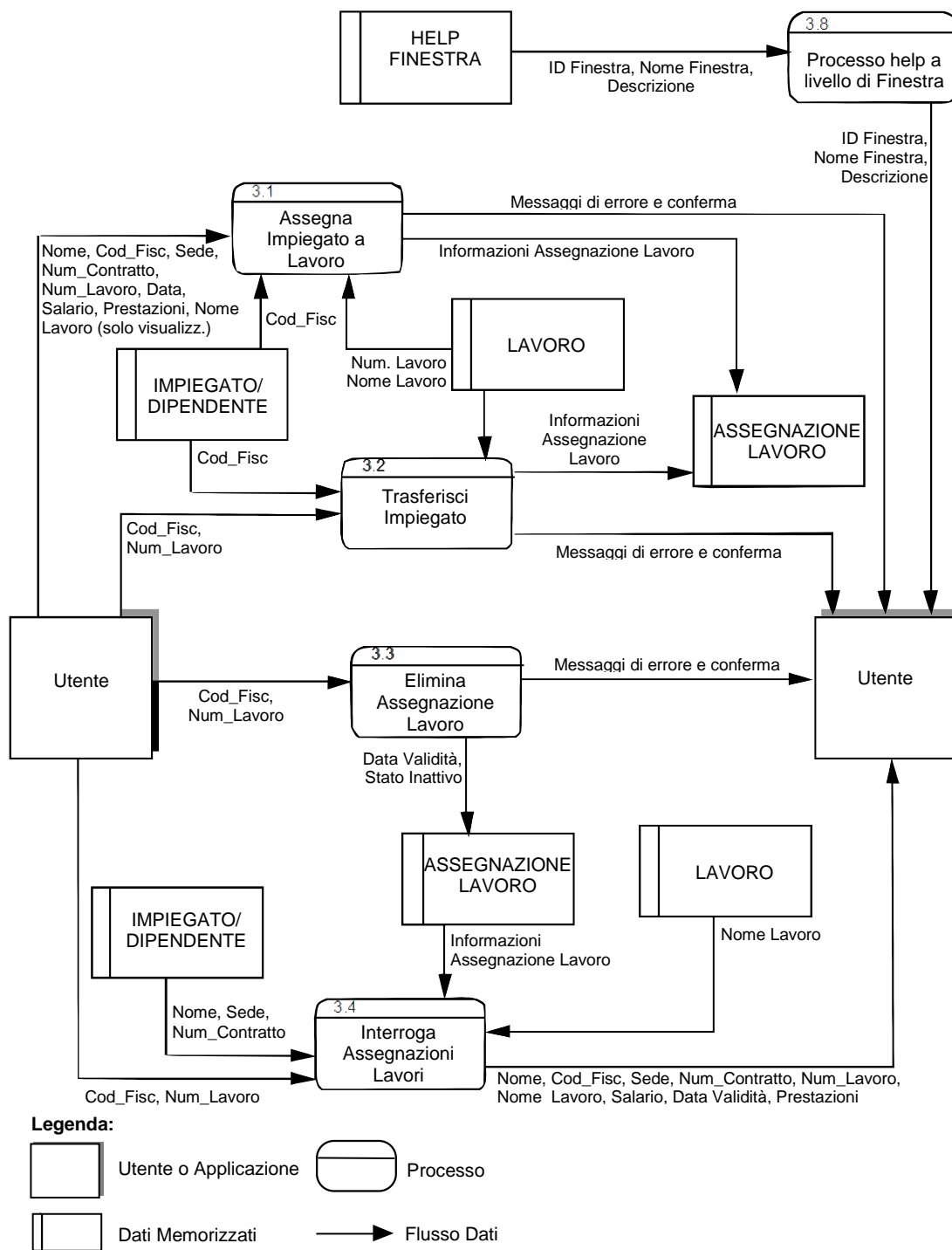
AE-3

OK Mostra la nuova assegnazione lavoro, torna al Menu

Cancella Ignora i dati inseriti, torna al Menu

**Diagramma di flusso dei dati**

Il seguente diagramma illustra il flusso dei dati per il processo assegnazione lavoro.



**Passo 1 Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì
---	----

**Passo 2 Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--

**Passo 3 Classifica il Processo Elementare**

Regole di conteggio dell'EI	Si applica la regola?
1. Il suo intento primario è di <ul style="list-style-type: none"> <li>Mantenere uno o più ILF oppure</li> <li>alterare il comportamento del sistema</li> </ul>	Sì. La funzione di correzione dei Lavori sospesi ha l'intento primario di mantenere l'ILF Assegnazione Lavori.  No. Il comportamento dell'applicazione non è alterato.
2. Include processi logici per l'inserimento dei dati o informazioni di controllo che attraversano il confine dell'applicazione.	Sì. La funzione Aggiungi Assegnazione Lavoro include logiche di controllo e di processo utili alla validazione delle informazioni sul Lavoro.

**Conclusione** La funzione aggiungi un'assegnazione lavoro è un EI.

**Passo 4 Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	L'ILF Impiegati è letto per assicurare l'esistenza dell'anagrafica dipendente.  L'ILF Lavori è letto per assicurare l'esistenza dell'anagrafica lavori  L'ILF Assegnazione Lavori è mantenuto e letto, ma è conteggiato una sola volta.

**Passo 5****Conteggiare i Tipi di Elementi Dati (DET)**

<b>Regole di conteggio dei DET</b>	<b>Si applica la regola?</b>
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Nome Impiegato, Codice Fiscale, Sede, Tipo Contratto, Codice Lavoro, Nome Lavoro, Data Effettiva, Retribuzione, Livello Risultati.  Lo schermo raffigura il Nome Impiegato in tre campi fisici sebbene il diagramma di flusso lo raffiguri come singolo elemento. Basandoci sulla revisione delle funzionalità applicative, il Nome Impiegato è sempre usato nella sua completezza. Non esistono schermate o reports dove una sola parte del Nome Impiegato venga utilizzata senza le altre  E' conteggiato una sola volta.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Messaggio di errore
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Bottone OK
2. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	Testi quali "Numero Lavoro" non sono conteggiati  Non esistono elementi di questo tipo.  Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo.  Non esistono elementi di questo tipo.

**Passo 6****Determinare la complessità funzionale**

3 FTR e 11 DET	La complessità è Alta.
----------------	------------------------

**Passo 7****Determinare la dimensione funzionale**

La dimensione funzionale di 1 EI a Alta complessità	6 FP
---	------

## Esempio: Conversione di dati

### Requisiti utente

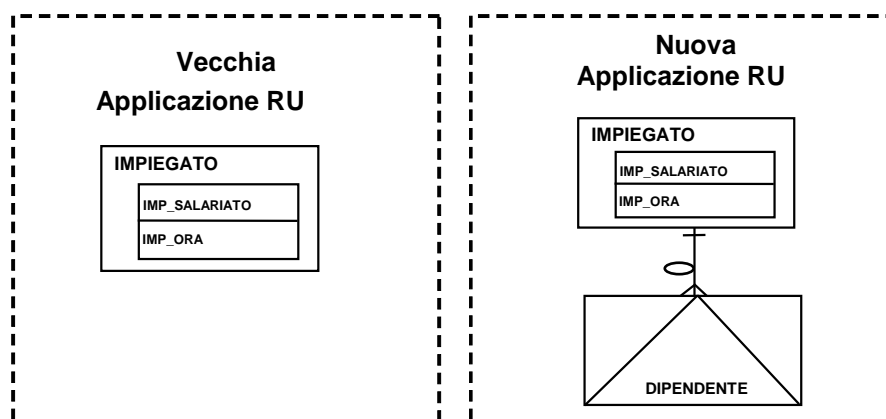
L'utente ha acquistato un nuovo pacchetto applicativo RU. L'utente richiede la possibilità di migrare le attuali informazioni sull'impiegato (Nome, Codice Fiscale, Numero di Dipendenti, Codice tipo, Livello supervisore, Paga oraria standard, Numero Unità Contrattazione Collettiva, Nome Sede) nella nuova applicazione.

Il vecchio sistema non consentiva all'utente di gestire le informazioni sui dipendenti dell'impiegato. Tali informazioni possono essere create dopo che gli impiegati esistenti sono migrati verso la nuova applicazione.






**Nota:** La Parte 3, Capitolo 5 (attività di Data Conversion) spiega come la conversione dati *una tantum* è misurata.

### Diagramma dei Dati

Il seguente diagramma illustra i dati per le applicazioni vecchia e nuova.



#### Legenda:

-  Entità
-  Entità Attributiva
-  Sottotipo di Entità
-  Relazione Obbligatoria 1 : N
-  Relazione Opzionale 1 : N

### Passo 1

#### Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Si
---	----



**Passo 2**      **Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--

**Passo 3**      **Classifica il Processo Elementare**

Regole di conteggio dell'EI	Si applica la regola?
1. Il suo intento primario è di <ul style="list-style-type: none"> <li>Mantenere uno o più ILF oppure</li> <li>alterare il comportamento del sistema</li> </ul>	Sì, L'ILF impiegati è mantenuto.  No. Il comportamento dell'applicazione non è alterato.
2. Include processi logici per l'inserimento dei dati o informazioni di controllo che attraversano il confine dell'applicazione.	Sì. I dati degli impiegati del vecchio sistema HR attraversano il confine dell'applicazione.

**Conclusione**      Convertire le informazioni degli impiegati è un EI.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	L'ILF impiegati è mantenuto e letto, ma è conteggiato una sola volta.

**Passo 5****Conteggiare i Tipi di Elementi Dati (DET)**

<b>Regole di conteggio dei DET</b>	<b>Si applica la regola?</b>
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Nome, Codice Fiscale, Numero di Dipendente, Codice tipo, Livello supervisore, Paga oraria standard, Numero Unità Contrattazione Collettiva, Nome Sede.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Non esistono elementi di questo tipo.
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo.

**Passo 6****Determinare la complessità funzionale**

1 FTR e 8 DET	La complessità è bassa.
---------------	-------------------------

**Passo 7****Determinare la dimensione funzionale**

La dimensione funzionale di 1 EI a Bassa complessità	3 FP
--	------

## Esempio: Riferimento a dati di un'altra applicazione

### Requisiti utente

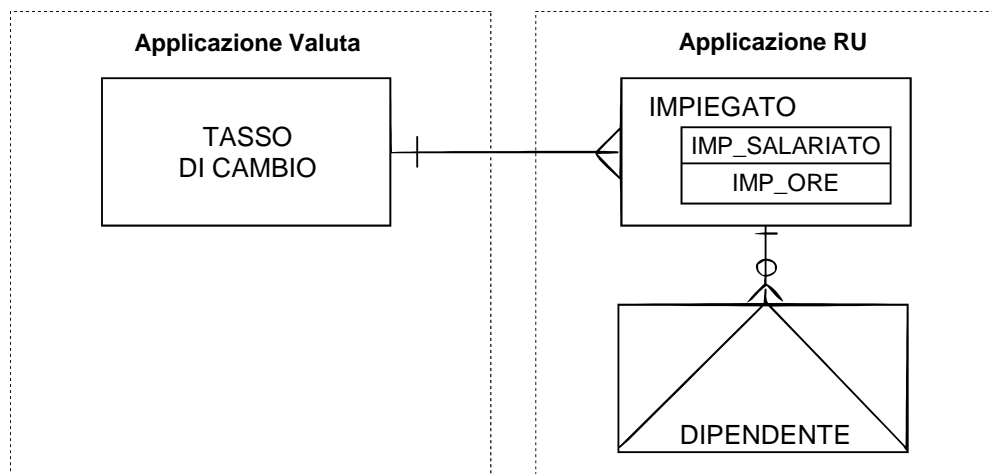
L'utente richiede che l'applicazione Risorse Umane fornisca le seguenti possibilità:

- Tutti gli impiegati ad ore devono essere pagati in dollari statunitensi.
- Quando l'utente aggiunge o modifica le informazioni sull'impiegato, l'applicazione Risorse Umane deve accedere all'applicazione Valuta per reperire il tasso di cambio. Dopo aver reperito il tasso di cambio, l'applicazione RU converte la paga oraria standard locale dell'impiegato in paga oraria in dollari U.S. con il seguente calcolo:






$$\frac{\text{Standard Hourly Rate}}{\text{Conversion Rate}} = \text{U.S. Dollar Hourly Rate}$$

### Diagramma dei Dati

Il seguente diagramma illustra le relazioni per questo esempio.



#### Legenda:

-  Entità
-  Entità Attributiva
-  Sottotipo di Entità
-  Relazione Obbligatoria 1 : N
-  Relazione Opzionale 1 : N

**Informazioni  
sulla  
Conversione**

Le informazioni sul cambio includono:

VALUTA

- Tasso\_di\_Cambio\_alla\_Valuta\_Base
- Nazione

**Passo 1****Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	No. Il riferimento ai dati è significativo solo quando associato all'inserimento di un impiegato.
---	---

**Conclusione**

Non c'è un EI per il recupero delle informazioni sulle valute per la conversione. Fare riferimento agli esempi di conteggio per EIF all'interno degli esempi di conteggio per le funzionalità dati per vedere perchè la conversione valute può essere conteggiata come EIF quando si aggiungono o cambiano le informazioni degli impiegati. Aggiungere o Modificare le informazioni sugli impiegati sono conteggiati come EI.

## Esempio: El con output a video – 1

### Requisiti utente

L'utente richiede la possibilità di memorizzare una transazione Vendite per un cliente. Prima che le informazioni siano memorizzate deve essere mostrato il prezzo di ogni articolo e deve essere visualizzato il totale della transazione per una verifica da parte dell'utente. In caso di errore deve essere visualizzato un messaggio appropriato.

### Schermata di esempio

La seguente schermata della transazione Vendite è una semplificazione utile per illustrare come sono conteggiati i campi di output. L'utente immette il nome del cliente e la data della transazione. Dopo che sono stati immessi tutti gli articoli e le quantità ordinate, il sistema calcola e visualizza i prezzi come di seguito indicato.

Transazione Vendite				
Nome Cliente: _____				
Data Transazione: _____				
Articolo	Qta	Costo Unit	Costo totale	
_____	_____	_____	\$ _____.	\$ _____.
_____	_____	_____	\$ _____.	\$ _____.
_____	_____	_____	\$ _____.	\$ _____.
_____	_____	_____	\$ _____.	\$ _____.
_____	_____	_____	\$ _____.	\$ _____.
_____	_____	_____	\$ _____.	\$ _____.
SubTotale			\$ _____.	
Imposte			\$ _____.	
Totale			\$ _____.	
F1=Salva				

### Passo 1

#### Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?

Si

### Passo 2

#### Determina l'univocità del Processo Elementare

La funzione transazionale è univoca rispetto agli altri processi elementari?

Sì. Nessun altro processo elementare esegue questa funzionalità.

**Passo 3 Classifica il Processo Elementare**

Regole di conteggio dell'EI	Si applica la regola?
1. Il suo intento primario è di <ul style="list-style-type: none"> <li>Mantenere uno o più ILF oppure</li> <li>alterare il comportamento del sistema</li> </ul>	Sì. La transazione di inserimento Vendite ha l'intento primario di mantenere l'ILF Transazioni Vendite.  No. Il comportamento dell'applicazione non è alterato.
2. Include processi logici per l'inserimento dei dati o informazioni di controllo che attraversano il confine dell'applicazione.	Sì. La funzione Inserimento Transazione di Vendita include logiche di controllo e di processo utili alla validazione delle informazioni sulla vendita

**Conclusione** L'inserimento di una transazione vendite è un EI.

**Passo 4 Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	L'ILF Transazioni di Vendita è letto e mantenuto.ma è conteggiato una sola volta.

**Passo 5 Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	<p>Sono contati i seguenti DET di input:  Nome Cliente  Data Transazione  Articolo (ripetuto)  Quantità (ripetuto)</p> <p>Sono contati i seguenti DET di output:  Prezzo Unitario Articolo (ripetuto)  Prezzo Totale Articolo (ripetuto)  Subtotale Transazione  Imposta  Totale Transazione</p>
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Messaggio di errore
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Tasto F1.
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Testi quali "Articolo" non sono conteggiati</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6 Determina la complessità funzionale**

1 FTR e 11 DET	
----------------	--

**Passo 7 Determinare la dimensione funzionale**

La dimensione funzionale di 1 EI a Bassa complessità	3 FP
--	------

## Esempio: El con output a video – 2

### Requisiti utente

L'utente richiede la possibilità di assegnare un lavoro ad un impiegato. Allo scopo di scegliere un impiegato e un lavoro, l'utente richiede la possibilità di far riferimento ai file Impiegato e Lavoro usando due elenchi a tendina. L'elenco Impiegato serve per mostrare numero e nome dell'impiegato. L'elenco Lavori serve per mostrare numero e descrizione del lavoro. Il numero di impiegati assegnati al lavoro deve essere visualizzato dopo la memorizzazione del record. In caso di errore, è visualizzato un messaggio appropriato.

### Schermata di esempio

La seguente schermata Assegnazione Lavoro è una semplificazione utile per illustrare come sono conteggiati i campi di output. L'utente sceglie l'impiegato da un elenco a tendina che mostra nome e numero dell'impiegato. Dopo la selezione, il sistema chiede il numero impiegato per l'assegnazione. L'utente sceglie il lavoro da un elenco a tendina che mostra numero e descrizione del lavoro. Il sistema chiede il codice (numero) lavoro per l'assegnazione. Dopo aver memorizzato l'assegnazione, il sistema determina il numero di impiegati e lo visualizza all'utente.

Sistema Risorse Umane		▼	◀▶
Impiegati	Lavori	Assegnazioni	Sedi Help
<b>Assegnazione Lavoro</b>			
Numero Impiegato	<input type="text" value="1290"/>	<input type="text" value="1290 James, R.W"/>	▼
Numero Lavoro	<input type="text" value="0100"/>	<input type="text" value="0100 Applicare Patinatura"/>	▼
Data Assegnazione	<input type="text" value="12/12/1998"/>		
<input type="button" value="Salva"/>			
Numero Totale di Impiegati assegnati a questo Lavoro		<input type="text" value="3"/>	

Le due caselle a discesa (list box) relative a Lavori e Impiegati costituiscono due processi elementari (EQ) separati e non sono analizzati in questo esempio.



**Passo 1 Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

**Passo 2 Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--

**Passo 3 Classifica il Processo Elementare**

Regole di conteggio dell'EI	Si applica la regola?
1. Il suo intento primario è di <ul style="list-style-type: none"> <li>Mantenere uno o più ILF oppure</li> <li>alterare il comportamento del sistema</li> </ul>	<p>Sì. L'intento primario è mantenere l'ILF Assegna lavori</p> <p>No. Il comportamento dell'applicazione non è alterato.</p>
2. Include processi logici per l'inserimento dei dati o informazioni di controllo che attraversano il confine dell'applicazione.	Sì, la transazione Assegna Lavori include logiche di processo utili alla validazione delle informazioni

**Conclusione** L'assegnazione di un Lavoro è un EI.

**Passo 4 Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	<p>L'ILF Assegnazione Lavoro è mantenuto.</p> <p>Gli ILF Impiegati e Lavori non sono conteggiati come FTR in quanto parte di un processo EQ separato.</p>

**Passo 5**      **Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	<p>Sono contati i seguenti DET di input: Numero Impiegato Codice Lavoro Data Assegnazione</p> <p>Sono contati i seguenti DET di output: Impiegati assegnati ad un lavoro</p> <p>I DET Nome Impiegato e Nome Lavoro nelle liste a tendina non sono contati come DET, in quanto essi sono parte di un'EQ separata.</p>
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Un messaggio è visualizzato in caso di errore.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	C'è un solo modo in cui la funzione può essere attivata, attraverso il tasto Salva.
4. Non contare I seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Testi quali "Numero Lavoro" non sono conteggiati</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6**      **Determina la complessità funzionale**

1 FTR e 6 DET	La complessità è Bassa.
---------------	-------------------------

**Passo 7**      **Determina la dimensione funzionale**

La dimensione funzionale di 1 EI a Bassa complessità	3 FP
--	------

## Esempio: El con attributi ricavati da un EIF

### Requisiti utente

L'utente richiede la possibilità di aggiungere un impiegato inserendo le seguenti informazioni:

- Informazioni relative all'impiegato
- Informazioni sul salario o la paga oraria
- Informazioni sui famigliari a carico
- La sede deve essere validata utilizzando le informazioni del Sistema Fixed Asset (FAS).
- La paga oraria è convertita in US\$; le informazioni sulle valute sono reperite dal Sistema Valute al fine di convertire la paga oraria standard in US\$ basandosi sulla valuta dell'impiegato.

### Finestra di esempio

La seguente figura illustra un esempio della finestra di inserimento di un Impiegato

The screenshot shows a window titled 'Sistema Risorse Umane' with a menu bar: Impiegato, Lavori, Assegnazioni, Sedi, Prospetti, Sicurezza, Help. Inside, a sub-window titled 'Dati Impiegato' contains the following fields and controls:

- Cognome: text input field
- Nome: text input field
- Iniz. 2° nome: text input field
- Codice Fiscale: text input field with dashes
- Numero di Dipendenti: text input field
- Sede: dropdown menu showing 'Sede Centrale' with a downward arrow
- Tipo Stipendio: radio buttons for '( ) Orario' and '( ) Pieno'
- Nuovo Liv. Supervisione: button
- Nuovo Tipo di Contratto: button
- OK: button
- Cancella: button

EN-1 Dati Impiegato

Nuovo Liv. Supervisione	Accede a SL-1 Livello Supervisione
Nuovo Tipo di Contratto	Accede a BU-1 Contratto di Lavoro
OK	Accede alla prossima schermata: • EN-2H, selezionando il tipo stipendio 'orario' (tempo parziale) • EN-2S, selezionando il tipo stipendio 'pieno' (tempo pieno)
Cancella	Torna al menu a tendina

**Sistema Risorse Umane**

Impiegato Lavori Assegnazioni Sedi Prospetti Sicurezza Help

**Dati Impiegato**

Cognome

Nome

Iniziale 2° nome

**Dati Impiegato Tempo Pieno**

Livello Supervisore  Responsabile 1° Liv.

OK

Cancella

Dipendenti

EN-2S Dati Impiegato: Stipendio

- |   |   |
|---|---|
| <input type="button" value="OK"/>         | Salva i dati del nuovo impiegato e ritorna alla schermata EN-1  |
| <input type="button" value="Cancella"/>   | Ritorna alla schermata EN-1 senza salvare i dati  |
| <input type="button" value="Dipendenti"/> | Accede a EN-3 Dati Impiegato Dipendente<br>(i dati in EN-2S Dati Impiegato: Stipendio sono memorizzati) |

**Sistema Risorse Umane**

Impiegato Lavori Assegnazioni Sedi Prospetti Sicurezza Help

**Dati Impiegato**

Cognome

Nome

Iniziale 2° nome

**Dati Impiegato Tempo Parziale**

Tariffa Oraria

Contratto Lavoro

OK

Cancella

Dipendenti

EN-2H Dati Impiegato a Tempo Parziale

- |   |   |
|---|---|
| <input type="button" value="OK"/>         | Salva i dati del nuovo impiegato e ritorna con dati aggiornati a EN-1 Dati Impiegato                    |
| <input type="button" value="Cancella"/>   | Ritorna a EN-1 Dati Impiegato (senza salvare dati)  |
| <input type="button" value="Dipendenti"/> | Accede a EN-3 Dati Impiegato Dipendente<br>(i dati in EN-2S Dati Impiegato: Stipendio sono memorizzati) |

**Sistema Risorse Umane**

Impegato Lavori Assegnazioni Sedi Prospetti Sicurezza Help

**Dati Impiegato**

**Dati Impiegato Tempo Pieno**

**Dati Dipendente**

Cognome

Nome

Iniziale 2° nome

Codice Fiscale

Data Nascita

OK

Cancella

Altri Dipendenti

EN-3 Dati Dipendente

OK	Salva i dati del nuovo dipendente e torna a EN-1 Dati Impiegato
Cancella	Torna a EN-2H Dati Dipendente a Tempo Parziale o EN-2S Dati Dipendente a Tempo Pieno (dati non salvati)
Altri Dipendenti	Accetta i dati del dipendente e ritorna con dati aggiornati a EN-3 Dati Dipendente

**Passo 1****Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Si
---	----

**Passo 2****Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EI	Si applica la regola?
1. Il suo intento primario è di <ul style="list-style-type: none"> <li>Mantenere uno o più ILF oppure</li> <li>Alterare il comportamento del sistema</li> </ul>	Sì. La funzione di Aggiungi Impiegato ha l'intento primario di mantenere l'ILF Impiegati  No. Il comportamento dell'applicazione non è alterato.
2. Include processi logici per l'inserimento dei dati o informazioni di controllo che attraversano il confine dell'applicazione.	Sì. La funzione Aggiungi Impiegato include logiche di processo utili alla validazione delle informazioni dell'impiegato.

**Conclusioni**      Aggiungere un Impiegato è un EI.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	L'ILF Impiegati è mantenuto.  Gli EIF Valute e Sedi sono referenziati.

## Passo 5

## Conteggia i Tipi di Elementi Dati (DET)

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	<p>Name (Last Name, First Name, Middle Initial), Social Security Number, Number of Dependents, Location, Salary Type, Supervisor Level, Hourly Rate, Bargaining Unit, Dependent Name (Last Name, First Name, Middle Initial), Dependent Social Security Number, Dependent Birth Date.</p> <p>Lo schermo raffigura il Nome Impiegato (name) in tre campi fisici sebbene il diagramma di flusso lo raffiguri come singolo elemento. Basandoci sulla revisione delle funzionalità applicative, il Nome Impiegato (name) è sempre usato nella sua completezza. Non esistono schermate o reports dove una sola parte del Nome Impiegato (name) venga utilizzata senza le altre. Lo stesso accade per il Nome Familiare (dependent name).</p> <p>Il nome impiegato (Employee Name) e nome familiare (Dependent Name) sono conteggiati come singoli DET.</p>
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Un Messaggio viene visualizzato in caso di errori.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Pulsante OK
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Testi quali "Last Name" non sono conteggiati</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>La paga oraria in US\$ (US Hourly Rate) è calcolata, ma non attraversa i confini dell'applicazione.</p> <p>Il tasso di conversione della valuta locale in valuta base è recuperato dall'EIF Valute al fine di calcolare la paga oraria in US\$, ma non esce dal confine dell'applicazione.</p>

**Passo 6**      **Determina la complessità funzionale**

3 FTR e 13 DET	La complessità è Alta.
----------------	------------------------

**Passo 7**      **Determina la dimensione funzionale**

La dimensione funzionale di 1 EI ad Alta complessità	6 FP
---	------



## Esempio: El cancellazione

### Requisiti utente

L'utente richiede la possibilità di cancellare un Dipendente

Cancellare tutte le informazioni di un singolo dipendente. Nel caso il dipendente sia al momento assegnato ad un lavoro, aggiornare l'assegnazione lavoro con lo status "inattivo"

- Generare messaggi di errore ed evidenziare i campi incorretti nel caso non siano corretti. Cinque (5) messaggi di errore e uno (1) di conferma sono inclusi nella transazione di cancellazione impiegato.
- Quando l'impiegato viene cancellato, aggiornare il campo Stato Inattivo con il valore "X" ed inserire la data di sistema all'interno del campo Data Validità per ogni assegnazione lavoro associata al dipendente.

### Finestra di esempio

La seguente figura illustra un esempio della finestra di cancellazione impiegato.

**Human Resources System**

Impiegato Lavori Assegnazioni Sedi Report Sicurezza Help

**Modifica Impiegato**

Nome: Kathleen MI: Cognome: Schmidt-Taylor

C.FISC.: 456-78-9012 OK

Sede: Main Office ↓ Annulla

Tipo Salario: Salaried ↓ Resetta

Dipendenti: 4 Dependents Elimina

Nuovo Liv. Supervisore Dati Salario

Nuova Unità Vendita Dipendenti

**EE2 – Employee Edit**

<b>OK</b>	Saves changes made, returns to EE-1 Employee List	<b>Elimina</b>	Asks for confirmation, then deletes employee record & associated dependent records
<b>Annulla</b>	Ignores changes, returns to EE-1 Employee List	<b>Dati Salario</b>	Navigates to: EE-3S Employee Salary, if value is "Salaried" EE-3H Employee Hourly, if value is "Hourly"
<b>Resetta</b>	Restores prior values	<b>Dipendenti</b>	Causes list on EI-4 Dependent List to display
<b>Nuovo Liv. Supervisore</b>	Navigates to SL -1 Supervisor Level		
<b>Nuova Unità Vendita</b>	Navigates to BU -1 Bargaining Unit		

**Passo 1**      **Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì
---	----

**Passo 2**      **Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--

**Passo 3**      **Classifica il Processo Elementare**

Regole di conteggio dell'EI	Si applica la regola?
1. Il suo intento primario è di <ul style="list-style-type: none"> <li>Mantenere uno o più ILF oppure</li> <li>alterare il comportamento del sistema</li> </ul>	<p>Sì. L'intento primario della funzione Elimina Impiegato è di mantenere l'ILF Impiegati</p> <p>No. Il comportamento dell'applicazione non è alterato.</p>
2. Include processi logici per l'inserimento dei dati o informazioni di controllo che attraversano il confine dell'applicazione.	Sì. La funzione Elimina Impiegato include logiche di processo utili alla validazione delle informazioni dell'impiegato.

**Conclusione**      La funzione Cancella un Impiegato è un EI.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	<p>L'ILF Assegnazione Lavori è mantenuto.</p> <p>L'ILF Impiegati è mantenuto e referenziato.</p>

**Passo 5 Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Codice Fiscale. Tutti gli altri attributi presenti sullo schermo fanno parte di un'interrogazione che precede la cancellazione. Queste informazioni non vengono conteggiate come DET della funzione Elimina.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Un messaggio è visualizzato in caso di errore.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Bottone Elimina.
4. Non contare i seguenti elementi come DET: <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Testi quali "C.FISC." non sono conteggiati.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>L'attributo "Stato Inattivo" è aggiornato, ma non esce dal confine applicativo. Non è conteggiato.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6 Determina la complessità funzionale**

2 FTR e 3 DET	La complessità è Bassa.
---------------	-------------------------

**Passo 7 Determina la dimensione funzionale**

La dimensione funzionale di 1 EI a Bassa complessità	3 FP
--	------

## Esempio: Aggiungere parametri di sicurezza

### Requisiti utente

Il requisito di gestire la sicurezza è stato aggiunto durante la fase di programmazione. L'utente ha richiesto di : aggiungere, modificare, cancellare e interrogare, all'interno del sistema Risorse Umane, le informazioni legate alla sicurezza. La funzionalità di aggiunta è illustrata di seguito.

Aggiungere informazioni sulla sicurezza inserendo:

- Identificativo utente (User Identifier)
- Finestre per le funzioni di assegnazione lavori, impiegati, sedi e generazione report.
- Parametri di sicurezza assegnati all'utente per le funzioni sopra descritte (ad esempio permesso o non permesso)

Generare messaggi di errore ed evidenziare i campi incorretti nel caso non siano corretti. Due (2) messaggi di errore e uno (1) di conferma sono inclusi nella transazione di inserimento dei parametri di sicurezza.

### Finestra di esempio

La seguente illustrazione mostra la schermata per l'inserimento delle informazioni sulla sicurezza.

SW-1 Impostazione Parametri Sicurezza RU

OK

Annulla

Conferma le modifiche e ritorna al menu principale del Sistema Risorse Umane (RU)

Ritorna al menu principale del Sistema Risorse Umane (RU)

I box nell'illustrazione sopra riportata rappresentano gli indirizzi delle finestre (Window ID).

**Passo 1**      **Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

**Passo 2**      **Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EI	Si applica la regola?
1. Il suo intento primario è di <ul style="list-style-type: none"> <li>Mantenere uno o più ILF oppure</li> <li>alterare il comportamento del sistema</li> </ul>	Sì, L'ILF Sicurezza Schermate è mantenuto.  No. Il comportamento dell'applicazione non è alterato.  Il comportamento dell'applicazione sarà alterato controllando le funzionalità che l'utente potrà eseguire all'atto del suo collegamento all'applicazione.
2. Include processi logici per l'inserimento dei dati o informazioni di controllo che attraversano il confine dell'applicazione.	Sì, Le informazioni sulla sicurezza delle schermate entrano nel confine applicativo al fine di mantenere l'ILF Sicurezza Schermate.

**Conclusione**      L'inserimento delle informazioni sulla sicurezza è un EI.

Fare riferimento al caso di studio per vedere come i requisiti di modifica e cancellazione e relativi schermi associati sono conteggiati.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	L'ILF Sicurezza Schermate è mantenuto.

**Passo 5****Conteggiare i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	ID Utente, Schermata, Sicurezza Utente.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Messaggio di errore
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Tasto Azione OK.
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Testi quali "User ID" non sono conteggiati</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6****Determina la complessità funzionale**

1 FTR e 5 DET	La complessità è Bassa.
---------------	-------------------------

**Passo 7****Determinare la dimensione funzionale**

La dimensione funzionale di 1 EI a Bassa complessità	3 FP
--	------

---

## EO: Esempi di conteggio

---

**Introduzione** Questa sezione utilizza un'applicazione Risorse Umane (RU) per illustrare le procedure usate per contare gli output esterni (EO). Oltre che in questa sezione, ulteriori esempi sono presentati nei Casi di Studio inclusi nella documentazione aggiuntiva IFPUG.

**Contenuti** Questa sezione include i seguenti esempi:

Argomento	Pagina
Descrizioni sintetiche degli esempi di conteggio degli EO	2-104
Esempio: report mediante hard copy	2-105
Esempio: Reportistica interattiva	2-108
Esempio: transazione inviata ad un'altra applicazione	2-111
Esempio: messaggi di errore/conferma	2-113
Esempio: notifica di riesame	2-114
Esempio: EO innescato senza dati che entrino nel confine	2-117
Esempio: intento primario di un EO	2-120
Esempio: file di transazione EO	2-123

## Descrizioni sintetiche degli esempi di conteggio degli EO

Nella seguente tabella sono elencati e descritti gli esempi per gli EO.

Esempio	Descrizione Sintetica	Pagina
Report mediante <i>hard copy</i>	Questo esempio considera il conteggio di un report hard copy.	2-105
Reportistica interattiva	Questo esempio illustra il conteggio per un report interattivo.	2-108
Transazione inviata ad un'altra applicazione	Questo esempio illustra una transazione generata da un'applicazione e inviata ad un'altra applicazione.	2-111
Messaggi di errore/conferma	Questo esempio illustra il motivo per cui i messaggi di errore o di conferma non sono contati come un output esterno.	2-113
Notifica di revisione delle performance	Questo esempio presenta il caso di una notifica basata su un calcolo.	2-114
EO innescato senza dati che entrino nel confine	Questo esempio illustra il concetto che un EO può essere innescato senza dati che entrino nel confine.	2-117
Intento primario di un EO	Questo esempio illustra che un EO può aggiornare un file.	2-120
File di transazione EO	Questo esempio illustra che l'esistenza di calcoli determina che il processo elementare è un EO e non un'EQ.	2-123



## Esempio: report mediante hard copy

### Requisiti utente

L'utente del Sistema Risorse Umane richiede un elenco delle assegnazioni dei lavori agli impiegati.

Il report è generato reperendo:

- un'assegnazione dall'ILF Assegnazione Lavoro;
- ulteriori informazioni dagli ILF Impiegato e Lavoro.

Viene referenziato l'ILF Definizione Report per determinare come generare il report.

### Report di esempio

Il seguente report Lavori con Impiegati elenca i lavori e gli impiegati ad essi assegnati.

SRU006	Sistema Risorse Umane	Pag. 1	
		Data 99.99.99	
Lavori con Impiegati			
Num Lavoro	Nome Lavoro	Cod_Fisc_Imp	Nome Impiegato
9999	xxxxxxxxxx	xxx-xx-xxxx	xxxxxxxxxxxxxxxxxx
		xxx-xx-xxxx	xxxxxxxxxxxxxxxxxx
		xxx-xx-xxxx	xxxxxxxxxxxxxxxxxx
9999	xxxxxxxxxx	xxx-xx-xxxx	xxxxxxxxxxxxxxxxxx
9999	xxxxxxxxxx	xxx-xx-xxxx	xxxxxxxxxxxxxxxxxx
		xxx-xx-xxxx	xxxxxxxxxxxxxxxxxx
Lavori Totali 9999			

**Passo 1**      **Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì
---	----

**Passo 2**      **Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Non esistono altri processi elementari che eseguano tale funzionalità.
--	--

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EO	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. Il report 'Lavori con Impiegati' risponde all'intento primario di presentare informazioni all'utente.
2. Include almeno una delle seguenti forme di trattamento logico:	
• Vengono eseguiti calcoli matematici	Sì. Il numero totale di lavori è considerato un dato sia calcolato che derivato.
• Sono mantenuti uno o più ILF	No. Nessun ILF è mantenuto.
• Sono create dati derivati oppure	Sì. Il numero totale di lavori è considerato un dato sia calcolato che derivato.
• Risulta alterato il comportamento del sistema	No. Il comportamento dell'applicazione non risulta alterato.

**Conclusione**      Il Report 'Lavori con Impiegati' è un EO.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	I seguenti ILF risultano letti: Impiegato, Lavoro, Assegnazione Lavoro, Controllo Report

**Passo 5****Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Sono stampati i seguenti capi: Codice Lavoro, Nome Lavoro, Codice Fiscale Impiegato, e Totale Lavori. Contare ogni campo solo una volta.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli.	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Non esistono elementi di questo tipo.
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• Costanti quali titoli dei report, identificativi delle schermate o dei pannelli, intestazione delle colonne o descrizione degli attributi.</li> <li>• Notazioni generate automaticamente dall'applicazione quali data e orario.</li> <li>• Numeri di pagina variabili o informazioni riguardanti il posizionamento (ad esempio Righe da 37 a 54 di 211)</li> <li>• Aiuti navigazionali quali la possibilità di movimento all'interno di una lista usando "precedente", "prossimo", "primo", "ultimo" e oggetti grafici equivalenti.</li> <li>• Attributi generati all'interno del confine da una funzione transazionale e salvati all'interno di un ILF che non escono dal confine stesso.</li> <li>• Attributi reperiti o referenziati da un ILF o EIF necessari alla elaborazione ma che non escono dal confine.</li> </ul>	<p>Identificatori di report, titoli del report e intestazioni delle colonne non sono contati.</p> <p>La data del report non è contata.</p> <p>Il numero di pagina non è contato.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6****Determina la complessità funzionale**

4 FTR e 5 DET	La complessità è Media
---------------	------------------------

**Passo 7****Determina la dimensione funzionale**

EO di Media Complessità	5 FP
-------------------------	------

## Esempio: Reportistica interattiva

**Requisiti utente** L'utente richiede un report degli impiegati ordinato in modo discendente in base alla durata della loro attuale assegnazione lavori. Questo report è visualizzato in modo interattivo e contiene dati derivati (ad esempio, la durata dell'assegnazione lavoro).

**Schermata di esempio** La seguente schermata 'Impiegati per Durata Assegnazione' elenca gli impiegati in base alla durata dell'assegnazione.

```

IMPIEGATI PER DURATA ASSEGNAZIONE

Linee da 1 a 18 di 1.316      GG/MM/AA

Cod. Fisc.  Nome  Numero Nome  Durata
Impiegato  Impiegato Lavoro Lavoro Assegnazione

xxx-xx-xxxx xxxxxxxxxxxx  9999  xxxxxxxxxxxx  99 mesi
xxx-xx-xxxx xxxxxxxxxxxx  9999  xxxxxxxxxxxx  99 mesi

    Impiegati oltre 24 mesi: 9999

    Impiegati oltre 12 mesi: 9999

F1=Help F7=Scorri su F8=Scorri giù F16=Fine

```

### Passo 1 Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

### Passo 2 Determina l'univocità del Processo Elementare

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Non esistono altri processi elementari che eseguano tale funzionalità.
--	--

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EO	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. Il report 'Impiegati per Durata Assegnazione' ha l'intento primario di presentare informazioni all'utente.
2. Include almeno una delle seguenti forme di trattamento logico:	
<ul style="list-style-type: none"> <li>Vengono eseguiti calcoli matematici</li> </ul>	Sì. Tre campi calcolati sono considerati sia calcolati che derivati.
<ul style="list-style-type: none"> <li>Sono mantenuti uno o più ILF</li> </ul>	No. Nessun ILF è mantenuto
<ul style="list-style-type: none"> <li>Sono create dati derivati oppure</li> </ul>	Sì. Tre campi calcolati sono considerati sia calcolati che derivati.
<ul style="list-style-type: none"> <li>Risulta alterato il comportamento del sistema</li> </ul>	No. Il comportamento dell'applicazione non risulta alterato.

**Conclusione**      Il report 'Impiegati per Durata Assegnazione' è un EO.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	Gli ILF Impiegati, Lavoro e Assegnazione Lavoro sono letti. Nessun ILF è mantenuto.

**Passo 5****Conteggia i Tipi di Elementi Dati (DET)**

<b>Regole di conteggio dei DET</b>	<b>Si applica la regola?</b>
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Codice Fiscale Impiegato, Nome Impiegato, Codice Fiscale Impiegato, Nome Impiegato, Codice Lavoro, Nome Lavoro e Durata Assegnazione sono ripetuti. Contare ogni campo solo una volta.  Numero degli Impiegati oltre i 24 mesi e oltre i 12 mesi.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Un tasto funzione è usato per visualizzare il report online.
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	Costanti come "Codice Fiscale Impiegato" non sono contate  La data del report non è contata.  Le etichette "Righe da 1 a 18" non sono contate.  I tasti F7 e F8 non sono contati.  Non esistono elementi di questo tipo.  Non esistono elementi di questo tipo.

**Passo 6****Determina la complessità funzionale**

3 FTR e 8 DET	La complessità è Media
---------------	------------------------

**Passo 7****Determina la dimensione funzionale**

La dimensione funzionale è di un EO di Media Complessità	5 FP
--	------

## Esempio: transazione inviata ad un'altra applicazione

**Requisiti utente** Quando il Sistema Risorse Umane inserisce i dati sui dipendenti dell'impiegato, l'utente richiede che queste informazioni siano inviate all'applicazione Benefici, per mantenere coerenti i record di tale applicazione. Queste informazioni sono inviate giornalmente all'applicazione Benefici.

**Requisiti tecnici** Se vengono inseriti i dati sui dipendenti, tali informazioni sono correttamente formattate sul file di output della transazione.

Nell'implementare una soluzione, si è deciso di includere, tra le informazioni destinate al Sistema Benefici, un record di inizio e un record di fine. Tali record sono usati dall'applicazione Benefici per verificare che nella trasmissione del file non si sono verificati inconvenienti tecnici.

**Formato Record di esempio** Il seguente formato del record Dipendenti dell'Impiegato contiene informazioni sull'inserimento e le variazioni dei dipendenti.

```

12345678910123456789012345678901234567890123456789012345678901234567890
1 H | FILE NAME | DATE |
2 D | EMP SSN | DEP SSN | DEPENDENT NAME | BDAY |
3 T | TOT. REC |
4
5
6
7
8
9
0

```

**Descrizione dei Campi** La seguente tabella contiene la descrizione di ogni campo del record.

Tipo Record	Posizione	Descrizione
Inizio	1	Tipo Record I
	2-13	Nome File
	14-19	Data Creazione
Dipendente	1	Tipo Record C
	2-10	Codice Fiscale Impiegato
	11-19	Codice Fiscale Dipendente
	20-39	Nome Dipendente
	40-45	Data di Nascita Dipendente
Fine	1	Tipo Record F
	2-10	Numero Totale dei Record

**Passo 1**      **Identifica il Processo Elementare - Inizio**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	No. Il record iniziale non contiene dati significativi per l'utente.
---	--

**Passo 1**      **Identifica il Processo Elementare- Fine**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	No. Il record finale non contiene dati significativi per l'utente.
---	--

**Passo 1**      **Identifica il Processo Elementare- Dipendente**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì. La sezione Dipendente del file di transazione soddisfa i requisiti per un processo elementare.
---	--

**Passo 2**      **Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Non esistono altri processi elementari che eseguano tale funzionalità.
--	--

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EO	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. Il file Dipendenti risponde all'intento primario di presentare informazioni all'utente.
2. Include almeno una delle seguenti forme di trattamento logico:	
• Vengono eseguiti calcoli matematici	No. Non sono eseguiti calcoli.
• Sono mantenuti uno o più ILF	No. Nessun ILF è mantenuto.
• Sono create dati derivati oppure	No. Nessun dato derivato è stato creato.
• Risulta alterato il comportamento del sistema	No. Il comportamento dell'applicazione non risulta alterato.

**Conclusione**      Il file Dipendente non si caratterizza come un EO; dovrebbe essere contato come un'EQ (non analizzato qui).



## Esempio: messaggi di errore/conferma

## Requisiti utente

Gli utenti richiedono un messaggio di ritorno quando vengono mantenute le informazioni sul lavoro. Più specificatamente, gli utenti richiedono messaggi che indichino errori di editing o di convalida o che indichino che il processo si è completato in maniera positiva.

### Schermata di esempio

La seguente schermata Lavoro illustra un messaggio di conferma (parte bassa della schermata).

```

Azione:___ 7=Precedente 8=Successivo 9=Salva
                        Dati sul Lavoro

Numero Lavoro:      RD15379305

Nome Lavoro:        Edizione di Maggio - Stampa delle copertine

Liv. Retrib.:       JRNY05A

N. Linea            Descrizione Lavoro
01                  Stampa delle copertine, 4-Up, Patinatura fine
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

F1=Help F7=Pag. Prec. F8=Pag. Succ. F12=Annulla
Elaborazione Completata con Successo

```

Invio:	ritorna alla videata di partenza	F12:	ritorna alla videata di partenza
F1:	mostra help a livello campo o finestra	Azione 7:	mostra dati lavoro precedente, se presente
F7:	scorre di 10 righe su la descrizione	Azione 8:	mostra dati lavoro successivo, se presente
F8:	scorre di 10 righe giù la descrizione		

## Passo 1      Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	No. L'output di un messaggio di errore non è un processo autonomo. È un DET di output dell'EI 'Aggiungi Lavoro'.
---	--

## Esempio: notifica di riesame delle prestazioni

### Requisiti utente

L'utente richiede una notifica automatica quando un impiegato ha completato 12 mesi in una assegnazione lavoro. Questo fatto indica che deve essere completata una verifica prestazionale.

### Finestra di esempio

La seguente finestra 'Notifica Riesame delle Prestazioni' descrive il messaggio di notifica.

Notifica Riesame Performance	
Data: xx/xx/xx	Ora: hh.mm
Impiegato: xxx-xx-xxxx	x_____x
Ha completato 12 mesi nell'assegnazione:	
Lavoro: xxxx	x_____x
E deve essere pianificato per un'immediata riesame della performance	

### Passo 1

#### Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

### Passo 2

#### Determina l'univocità del Processo Elementare

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Non esistono altri processi elementari che eseguano tale funzionalità.
--	--

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EO	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Yes. Performance Review Notification has the primary intent to present information to the user.
2. Include almeno una delle seguenti forme di trattamento logico:	
<ul style="list-style-type: none"> <li>Vengono eseguiti calcoli matematici</li> </ul>	Sì. Viene calcolata la data di completamento dei 12 mesi.
<ul style="list-style-type: none"> <li>Sono mantenuti uno o più ILF</li> </ul>	No. Nessun ILF è mantenuto.
<ul style="list-style-type: none"> <li>Sono create dati derivati oppure</li> </ul>	No. Nessun dato derivato è stato creato.
<ul style="list-style-type: none"> <li>Risulta alterato il comportamento del sistema</li> </ul>	No. Il comportamento dell'applicazione non risulta alterato.

**Conclusione**      La 'Notifica di Riesame Performance' è un EO.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	I seguenti ILF risultano letti: Impiegato, Lavoro, Assegnazione Lavoro.

**Passo 5 Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Data, Codice Fiscale Impiegato, Nome Impiegato, Codice Lavoro, Nome Lavoro
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Non esistono elementi di questo tipo.
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Costanti come "Impiegato" non sono contate.</p> <p>Data e orario del report non sono contate.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6 Determina la complessità funzionale**

3 FTR e 5 DET	La complessità è Bassa
---------------	------------------------

**Passo 7 Determina la dimensione funzionale**

La dimensione funzionale è di un EO di Bassa Complessità	4 FP
--	------

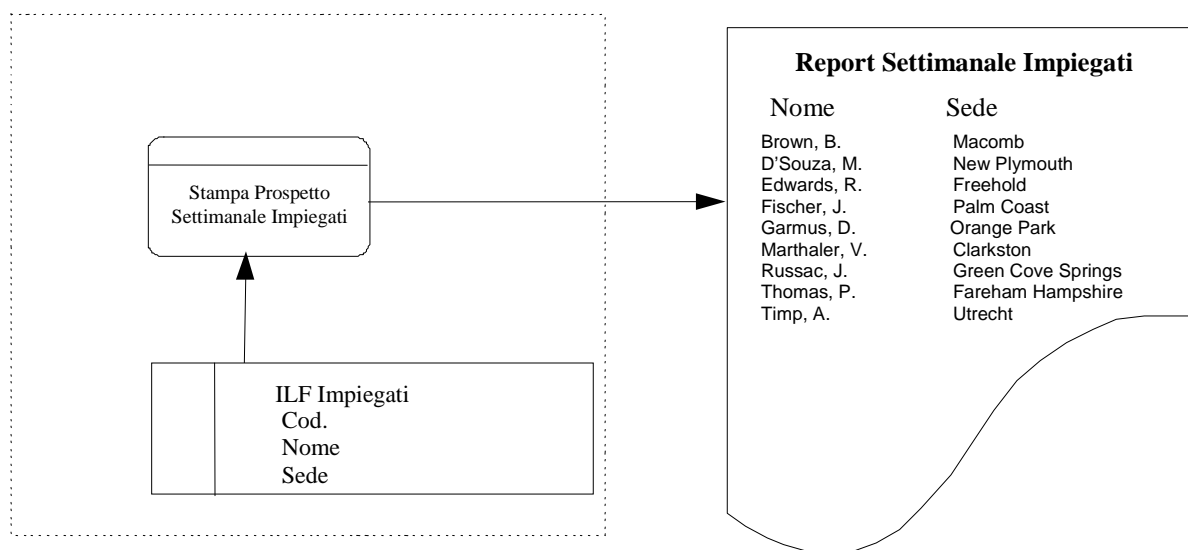
## Esempio: EO innescato senza dati che entrino nel confine

### Requisiti utente

Gli utenti richiedono che l'applicazione stampi automaticamente il Report Settimanale degli Impiegati ogni domenica notte alle ore 23. Il report contiene dettagli per ogni impiegato più un totale di tutti gli impiegati.

### Modello dei Dati

Il seguente diagramma illustra il flusso dei dati per questo esempio.



### Passo 1 Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?

Sì.

### Passo 2 Determina l'univocità del Processo Elementare

La funzione transazionale è univoca rispetto agli altri processi elementari?

Sì. Non esistono altri processi elementari che eseguano tale funzionalità.

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EO	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. Il Report Settimanale Impiegati risponde all'intento primario di presentare informazioni all'utente.
2. Include almeno una delle seguenti forme di trattamento logico:	
• Vengono eseguiti calcoli matematici	Sì. 'Totale Impiegati' è un campo calcolato.
• Sono mantenuti uno o più ILF	No. Nessun ILF è mantenuto.
• Sono create dati derivati oppure	Sì. Il report contiene un campo calcolato.
• Risulta alterato il comportamento del sistema	No. Il comportamento dell'applicazione non risulta alterato.

**Conclusione**      Il Report Settimanale Impiegati è un EO.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	L'ILF Impiegato è letto. Nessun ILF è mantenuto.

**Passo 5 Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della transazione funzionale.	Nome, Sede, Totale Impiegati.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Non esistono elementi di questo tipo.
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Costanti quali "Nome" non sono contate.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6 Determina la complessità funzionale**

1 FTR e 3 DET	La complessità è Bassa
---------------	------------------------

**Passo 7 Determina la dimensione funzionale**

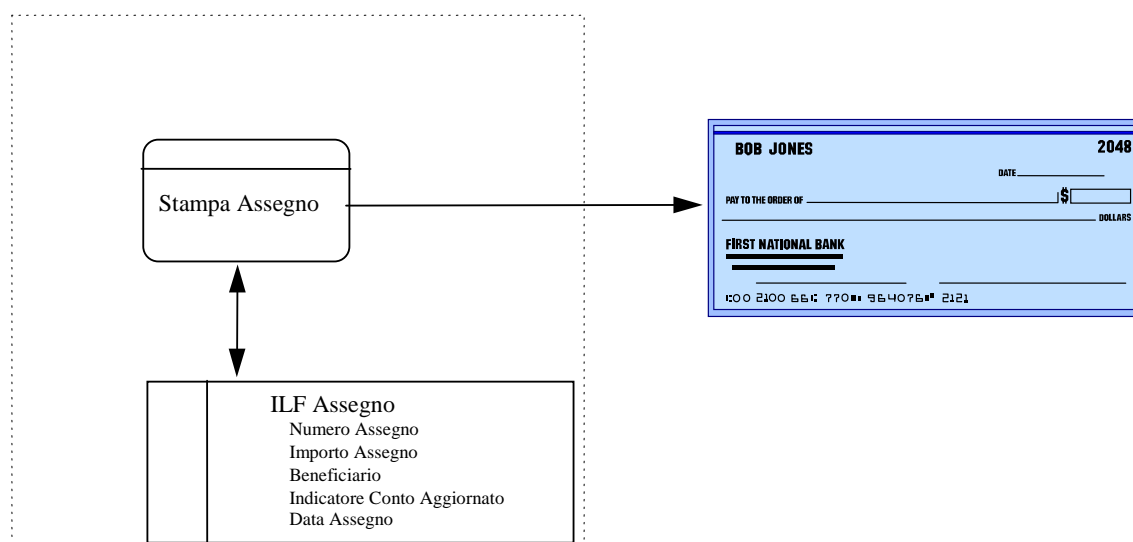
La dimensione funzionale è di un EO di Bassa Complessità	4 FP
--	------

## Esempio: intento primario di un EO

### Requisiti utente

Stampare un assegno e, come risultato, aggiornare il conto per tenere traccia del pagamento dell'assegno. Tutti i dati stampati sull'assegno sono già memorizzati nel file degli assegni.

Il seguente diagramma illustra il flusso dei dati per questo esempio.



### Passo 1

#### Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

### Passo 2

#### Determina l'univocità del Processo Elementare

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Non esistono altri processi elementari che eseguano tale funzionalità.
--	--



**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EO	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. L'intento primario è stampare un assegno. L'aggiornamento dell'ILF è secondario.
2. Include almeno una delle seguenti forme di trattamento logico:	
• Vengono eseguiti calcoli matematici	No. Non sono eseguiti calcoli.
• Sono mantenuti uno o più ILF	Sì. L'ILF Assegno è aggiornato.
• Sono create dati derivati oppure	No. Nessun dato derivato è stato creato
• Risulta alterato il comportamento del sistema	No. Il comportamento dell'applicazione non risulta alterato.

**Conclusione**      Stampare un Assegno è un EO.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	L'ILF Assegno è letto e mantenuto, ma è contato solo una volta.

**Passo 5 Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della transazione funzionale.	Numero Assegno, Importo Assegno, Destinatario, Data Assegno.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Non esistono elementi di questo tipo.
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Non esistono elementi di questo tipo.</p> <p>C'è una data stampata sull'assegno ma rappresenta un dato riconoscibile dall'utente e quindi è contato.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>L'indicatore di Conto Saldato non è contato dato che non attraversa il confine.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6 Determina la complessità funzionale**

1 FTR e 4 DET	La complessità è Bassa
---------------	------------------------

**Passo 7 Determina la dimensione funzionale**

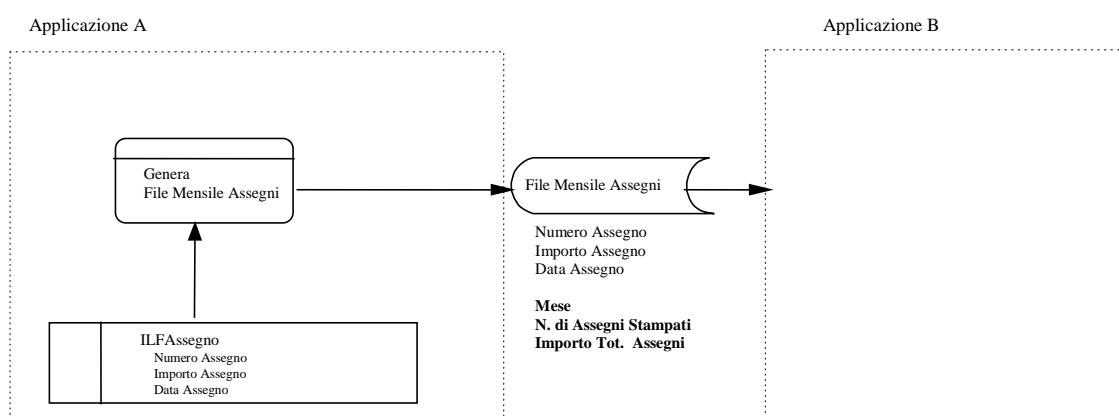
La dimensione funzionale è di un EO di Bassa Complessità	4 FP
--	------

## Esempio: file di transazione EO

### Requisiti utente

Alla fine del mese, si richiede di generare un file di transazione e inviarlo all'Applicazione B. I numeri, le date e gli importi degli assegni sono inclusi nel file con il numero calcolato degli assegni elaborati e l'importo totale di tutti gli assegni stampati per il mese. Il numero di assegni stampati e l'ammontare totale in dollari di tutti gli assegni stampati viene dagli utenti dell'Applicazione B per produrre rapporti informativi bancari.

**Modello dei Dati** Il seguente diagramma illustra il flusso dei dati per questo esempio.



### Passo 1 Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

### Passo 2 Determina l'univocità del Processo Elementare

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Non esistono altri processi elementari che eseguano tale funzionalità.
--	--

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EO	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. L'intento primario è generare un file di transazione.
2. Include almeno una delle seguenti forme di trattamento logico:	
• Vengono eseguiti calcoli matematici	Sì. Il file include due campi calcolati.
• Sono mantenuti uno o più ILF	No. Nessun ILF è mantenuto.
• Sono create dati derivati oppure	Sì. Il file include due campi calcolati.
• Risulta alterato il comportamento del sistema	No. Il comportamento dell'applicazione non risulta alterato.

**Conclusione**    Il File Assegni Mensili è un EO.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	L'ILF Assegni ILF è letto. Non ci sono ILF mantenuti.

**Passo 5 Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della transazione funzionale.	Numero assegno, Importo Assegno, Data Assego, Mese, Numero di assegni stampati, Valore totale degli Assegni
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Non esistono elementi di questo tipo.
4. Non contare i seguenti elementi come DET <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo.

**Passo 6 Determina la complessità funzionale**

1 FTR e 6 DET	La complessità è Bassa
---------------	------------------------

**Passo 7 Determina la dimensione funzionale**

La dimensione funzionale è di un EO di Bassa Complessità	4 FP
--	------

Questa pagina è lasciata intenzionalmente vuota.

---

## EQ: Esempi di conteggio

---

**Introduzione** Questa sezione utilizza un'applicazione Risorse Umane (RU) per illustrare le procedure di conteggio delle interrogazioni esterne. Oltre che in questa sezione, ulteriori esempi sono presentati nei Casi di Studio inclusi nella documentazione aggiuntiva IFPUG.

**Contenuti** Questa sezione include i seguenti esempi:

Topic	Page
Descrizioni sintetiche degli esempi di conteggio di EQ	2-128
Esempio: Menu dell'applicazione	2-129
Esempio: Lista di dati reperiti	2-131
Esempio: Casella di lista a discesa	2-136
Esempio: Help di campo – Prima occorrenza	2-140
Esempio: Help di campo – Seconda occorrenza	2-143
Esempio: Interrogazione implicita	2-145
Esempio: EQ innescata senza dati che entrano nel confine	2-149
Esempio: Dati inviati ad un'altra applicazione	2-152
Esempio: Funzionalità di Help aggiuntiva	2-155
Esempio: Logon di sicurezza	2-159

## Descrizioni sintetiche degli esempi di conteggio di EQ

Nella seguente tabella sono elencati e descritti gli esempi per le EQ.

Esempio	Descrizione Sintetica	Pagina
Menu dell'applicazione	Questo esempio illustra perché i menu di navigazione o altri aiuti per la navigazione non sono contati come EQ.	2-129
Lista di dati reperiti	Questo esempio illustra il conteggio per una lista visualizzata su di una finestra.	2-131
Casella di lista a discesa	Questo esempio illustra come potrebbe essere contata una casella di lista a discesa.	2-136
Help di campo – Prima occorrenza	Questo esempio illustra come è contato un help di campo alla prima occorrenza.	2-140
Help di campo – Seconda occorrenza	Il conteggio di una seconda istanza di help di campo è illustrato in questo esempio.	2-143
Interrogazione implicita	Questo esempio illustra il conteggio dei function point quando il reperimento di dati non è dichiarato esplicitamente, ma è implicito.	2-145
EQ innescata senza dati che attraversano il confine	Questo esempio illustra il conteggio per il reperimento e la visualizzazione di dati innescati internamente dall'orologio del sistema.	2-149
Dati inviati ad un'altra applicazione	Questo esempio illustra il conteggio dei dati inviati ad un'altra applicazione attraverso un file.	2-152
Funzionalità di help aggiuntiva	Questo esempio illustra il conteggio di una funzionalità di help aggiuntiva.	2-155
Logon di sicurezza	Questo esempio illustra il conteggio della funzionalità di logon.	2-159



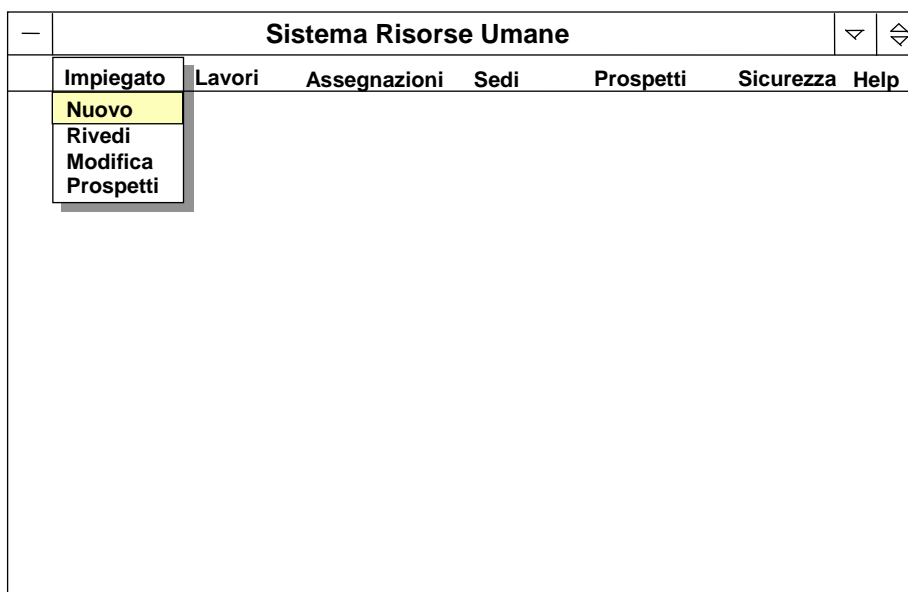
## Esempio: Menu dell'applicazione

### Requisiti utente

L'applicazione Risorse Umane richiede dei menu e degli aiuti per la navigazione.

### Schermata di esempio

Il seguente diagramma illustra la finestra di dialogo Impiegato nel menu principale del Sistema Risorse Umane. Questa è la richiesta di input.



Quando l'utente seleziona Nuovo sul menu a tendina Impiegato, viene visualizzata la seguente finestra vuota Impostazione Impiegato.

EN-1

Annulla

Ritorna al menu precedente

OK

Naviga alla schermata successiva:

- EN-2H, per selezione tipo pagamento a Ore
- EN-2S, per selezione tipo pagamento Salariato

## Passo 1

### Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?

No. La selezione da un menu di opzioni non include nessun dato significativo per l'utente.

**Conclusione** I menu non soddisfano i requisiti di un processo elementare.

## Esempio: Lista di dati reperiti

### Requisiti utente

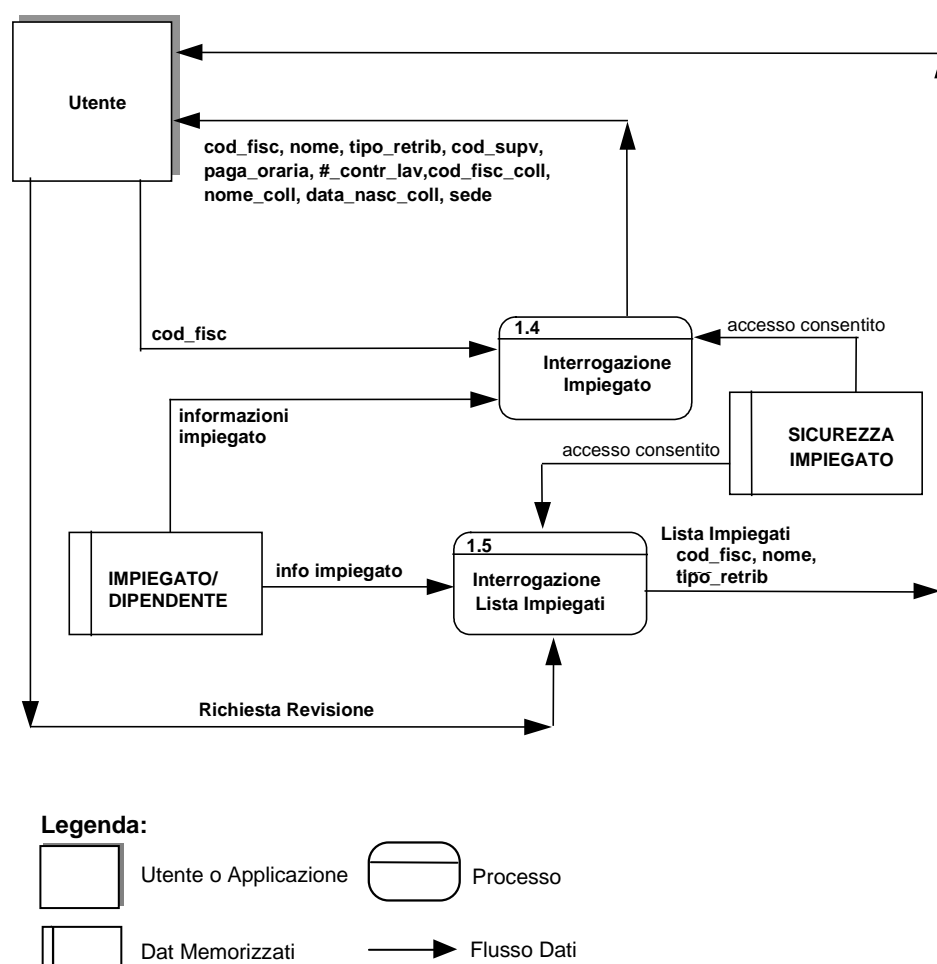
L'utente specifica i seguenti requisiti:

- Visualizzazione di una Lista di Impiegati ordinate per Cognome, Nome e iniziale del secondo nome.

Questo esempio pone l'attenzione sulla visualizzazione di una lista di impiegati nell'applicazione Risorse Umane (RU).

### Diagramma del Flusso Dati (DFD)

Il seguente diagramma illustra il flusso dei dati per questo esempio.



**Schermata di esempio**

Il seguente diagramma illustra il menu a tendina per Impiegato. Il campo Rivedi e il tasto di Invio rappresentano il lato di input di questo esempio.

Sistema Risorse Umane						
Impiegato	Lavori	Assegnazioni	Sedi	Prosp-Dir	Sicurezza	Help
Nuovo						
Rivedi						
Modifica						
Prospetti						

Quando l'utente seleziona il comando 'Rivedi' sul menu Impiegato, viene visualizzata la seguente finestra con una lista di impiegati.

Sistema Risorse Umane																																																					
Impiegati	Lavori	Assegnazioni	Sedi	Prosp_Dir	Sicurezza	Help																																															
<table border="1"> <thead> <tr> <th colspan="6">Lista Impiegati</th> </tr> <tr> <th>Cognome</th> <th>Nome</th> <th>S_N</th> <th>Cod. Fiscale</th> <th>Tipo Retrib.</th> <th></th> </tr> </thead> <tbody> <tr> <td>Keller</td> <td>Caroline</td> <td></td> <td>123-45-6789</td> <td>Salariato</td> <td>↑</td> </tr> <tr> <td>Latta</td> <td>Nicky</td> <td>A</td> <td>234-56-7890</td> <td>Ora</td> <td></td> </tr> <tr> <td>Manship</td> <td>Mark</td> <td>J</td> <td>345-67-8901</td> <td>Ora</td> <td></td> </tr> <tr> <td>Schmidt-Taylor</td> <td>Kathleen</td> <td></td> <td>456-78-9012</td> <td>Salariato</td> <td></td> </tr> <tr> <td>Smith</td> <td>David</td> <td>E</td> <td>567-89-0123</td> <td>Ora</td> <td></td> </tr> <tr> <td>Smith</td> <td>Loretta</td> <td>M</td> <td>678-90-1234</td> <td>Salariato</td> <td>↓</td> </tr> </tbody> </table>						Lista Impiegati						Cognome	Nome	S_N	Cod. Fiscale	Tipo Retrib.		Keller	Caroline		123-45-6789	Salariato	↑	Latta	Nicky	A	234-56-7890	Ora		Manship	Mark	J	345-67-8901	Ora		Schmidt-Taylor	Kathleen		456-78-9012	Salariato		Smith	David	E	567-89-0123	Ora		Smith	Loretta	M	678-90-1234	Salariato	↓
Lista Impiegati																																																					
Cognome	Nome	S_N	Cod. Fiscale	Tipo Retrib.																																																	
Keller	Caroline		123-45-6789	Salariato	↑																																																
Latta	Nicky	A	234-56-7890	Ora																																																	
Manship	Mark	J	345-67-8901	Ora																																																	
Schmidt-Taylor	Kathleen		456-78-9012	Salariato																																																	
Smith	David	E	567-89-0123	Ora																																																	
Smith	Loretta	M	678-90-1234	Salariato	↓																																																
<div>Visualizza</div> <div>Dipendenti</div> <div>Annulla</div>																																																					

EI-1

- |            |   |
|------------|---|
| Visualizza | Attiva visualizzazione dei dati su EI-2 |
| Dipendenti | Attiva lista visualizzata su EI-4       |
| Annulla    | Ritorna al menù precedente              |

**Passo 1**      **Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

**Passo 2**      **Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EQ	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. L'intento primario è presentare informazioni all'utente.
<ul style="list-style-type: none"> <li>• riferenzia una funzione di tipo dati per reperire dati o informazioni di controllo e</li> </ul>	Sì. I dati sono reperiti dall'ILF Impiegato.
<ul style="list-style-type: none"> <li>• non soddisfa i criteri per essere classificato come un EO</li> </ul>	Sì. Non sono eseguiti calcoli, non sono create dati derivati e non è aggiornato nessun ILF.

**Conclusione**      La Lista degli Impiegati è un'EQ.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)	L'ILF Impiegati è letto. Per definizione, un'EQ non può mantenere un ILF.

## Passo 5

## Conteggia i Tipi di Elementi Dati (DET)

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	I seguenti campi sono ripetuti e contati quindi una sola volta. Cognome + Nome + Secondo Nome, S_N, Codice Fiscale, Tipo Retribuzione.  La lista è ordinate usando singoli elementi del nome (es: Cognome, Nome, Secondo Nome, MI). Come risultato, i singoli elementi del Nome sono riconoscibili dall'utente.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Sì. Il tasto Visualizza/Invio che attiva l'azione.
4. Non contare i seguenti elementi come DET: <ul style="list-style-type: none"> <li>• Costanti quali titoli dei report, identificativi delle schermate o dei pannelli, intestazione delle colonne o descrizione degli attributi</li> <li>• Notazioni generate automaticamente dall'applicazione quali data e orario</li> <li>• Numeri di pagina variabili o informazioni riguardanti il posizionamento (ad esempio Righe da 37 a 54 di 211)</li> <li>• Aiuti navigazionali quali la possibilità di movimento all'interno di una lista usando "precedente", "prossimo", "primo", "ultimo" e oggetti grafici equivalenti.</li> <li>• Attributi generati all'interno del confine da una funzione transazionale e salvati all'interno di un ILF che non escono dal confine stesso.</li> <li>• Attributi reperiti o referenziati da un ILF o EIF necessari alla elaborazione ma che non escono dal confine.</li> </ul>	<p>Non sono conteggiate le intestazioni della finestra e delle colonne.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non è conteggiata la barra di scorrimento.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6**      **Determina la complessità funzionale**

1 FTR e 6 DET	La complessità è Bassa
---------------	------------------------

**Passo 7**      **Determina la dimensione funzionale**

La dimensione funzionale di 1 EQ a Bassa Complessità	3 FP
---	------

## Esempio: Casella di lista a discesa

### Requisiti utente

L'utente richiede la possibilità di vedere una lista di contratti di lavoro inseriti da un utente nel Sistema Risorse Umane.

### Schermate di Esempio

Il seguente diagramma illustra la finestra Dati Impiegato ad Ore con il campo Contratto Lavoro.

Il diagramma illustra la struttura delle finestre del sistema "Sistema Risorse Umane".

- La finestra principale, "Sistema Risorse Umane", ha una barra di menu con: Impiegato, Lavori, Assegnazioni, Sedi, Prosp-Dir, Sicurezza, Help.
- Alcune finestre sono aperte:
  - "Lista Impiegati" (sopra "Dati Impiegato").
  - "Dati Impiegato" (contiene campi: Mark, J, Manship, 345-67-8901).
  - "Dati Impiegato ad Ore" (contiene:
    - Campo "Paga Oraria".
    - Campo "Contratto Lavoro" con una casella a discesa (contenitore giallo).
    - Pulsante "Dipendenti" (contenitore giallo).

Etichette di interfaccia:

- Elenco: EI-3H
- Pulsante: Dipendenti
- Link: Visualizza lista su EI-4



Quando l’utente seleziona la freccia, appare la seguente lista a discesa (drop-down list).

Sistema Risorse Umane

Impiegato

Lavori

Assegnazioni

Sedi

Prosp-Dir

Sicurezza

Help

Lista Impiegati

Dati Impiegato

Mark

J

Manship

345-67-8901

Dati Impiegato ad Ore

Paga Oraria

Contratto di Lavoro

UPFCA  
L841  
CPLG

Dipendenti

EI-3H

Dipendenti

Visualizza lista su EI-4

Passo 1

Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

Passo 2

Determina l’univocità del Processo Elementare

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EQ	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. L'intento primario è presentare informazioni all'utente.
<ul style="list-style-type: none"><li>• referencia una funzione di tipo dati per reperire dati o informazioni di controllo e</li></ul>	Sì. I dati sono reperiti dall'ILF Contratto di Lavoro. L'ILF Contratto di Lavoro contiene diversi attributi relativi ad un Contratto di Lavoro
<ul style="list-style-type: none"><li>• non soddisfa i criteri per essere classificato come un EO</li></ul>	Sì. Non sono eseguiti calcoli, non sono create dati derivati e non è aggiornato nessun ILF.

**Conclusione**    La Lista dei Contratti di Lavoro è un'EQ.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)-	L'ILF Contratto di Lavoro è letto. Per definizione, un'EQ non può mantenere un ILF.

**Passo 5****Conteggia i Tipi di Elementi Dati (DET)**

<b>Regole di conteggio dei DET</b>	<b>Si applica la regola?</b>
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Contratti di Lavoro.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Sì. La freccia "giù" attiva la casella a discesa.
4. Non contare i seguenti elementi come DET: <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo.

**Passo 6****Determina la complessità funzionale**

1 FTR e 2 DET	La complessità è Bassa
---------------	------------------------

**Passo 7****Determina la dimensione funzionale**

La dimensione funzionale di 1 EQ a Bassa Complessità	3 FP
--	------

## Esempio: Help di campo – Prima occorrenza

### Requisiti utente

Durante la realizzazione del Sistema Risorse Umane è stato aggiunto un requisito per avere l'help di campo in modalità interattiva. Le informazioni di help sono mantenute da un'applicazione a parte. Le informazioni di help sono referenziate dalle applicazioni Risorse Umane, Valuta, Beni Patrimoniali e Sistema Benefici.

### Schermate di Esempio

Il seguente diagramma illustra la finestra Dati Impiegato.

Quando l'utente è posizionato sul campo Paga Oraria e preme **F1**, una finestra visualizza il testo di help, come mostrato nel seguente diagramma.

**Sistema Risorse Umane**

Impiegato Lavori Assegnazioni Sedi Prospetti Sicurezza Help

**Paga Oraria**  
 Il totale corrisposto ad un impiegato per ogni ora completa di lavoro.  
 Questa informazione deve essere inserita per ogni impiegato con paga oraria.  
 Valori Ammessi: E' accettabile qualsiasi valore per l'orario.  
 Valore Pre-Assegnato: Per questo campo non ci sono valori Pre-Assegnati.

Paga Oraria

Contratto di Lavoro

EI-3H  Visualizza lista su EI-4

**Passo 1 Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

**Passo 2 Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--

**Passo 3 Classifica ciascun Processo Elementare**

Regole di conteggio dell'EQ	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. L'Help di Campo ha l'intento primario di presentare informazioni all'utente.
<ul style="list-style-type: none"> <li>• referencia una funzione di tipo dati per reperire dati o informazioni di controllo e</li> </ul>	Sì. I dati sono reperiti dall'EIF Help.
<ul style="list-style-type: none"> <li>• non soddisfa i criteri per essere classificato come un EO</li> </ul>	Sì. Non sono eseguiti calcoli, non sono create dati derivati e non è aggiornato nessun ILF.

**Conclusione** L'Help di Campo è un'EQ.

**Passo 4****Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)-	L'EIF Help è letto. Per definizione, un'EQ non può mantenere un ILF.

**Passo 5****Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	ID Finestra, ID Campo, Messaggio di Help, Valore Pre-Assegnato, Valori Ammessi.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Sì. Il tasto F1.
4. Non contare i seguenti elementi come DET: <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	Costanti come "Valori Validi" non sono contate. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo.

**Passo 6****Determina la complessità funzionale**

1 FTR e 6 DET	La complessità è Bassa
---------------	------------------------

**Passo 7****Determina la dimensione funzionale**

La dimensione funzionale di 1 EQ a Bassa Complessità	3 FP
--	------

## Esempio: Help di campo – Seconda occorrenza

### Requisiti utente

Durante la realizzazione del Sistema Risorse Umane è stato aggiunto un requisito per avere l'help di campo in modalità interattiva. L'help in modalità interattiva serve ai processi di inserimento, cancellazione e modifica delle informazioni delle Risorse Umane. Le informazioni di help sono mantenute da un'applicazione a parte. Le informazioni di help sono referenziate dalle applicazioni Risorse Umane, Valuta, Beni Patrimoniali e Sistema Benefici.

### Schermate di Esempio

Il seguente diagramma illustra la finestra Dati Impiegato.

Diagramma della finestra Dati Impiegato:

- Titolo: Sistema Risorse Umane
- Menu: Impiegato, Lavori, Assegnazioni, Sedi, Prospetti, Sicurezza, Help
- Pannello Lista Impiegati
- Pannello Dati Impiegato
  - Mark
  - J
  - Manship
  - 345-67-8901
- Pannello Dati Impiegato ad Ore
  - Paga Oraria
  - Contratto di Lavoro (menu a tendina)
  - Dipendenti (pulsante)
- Bottoni e link in basso:
  - Dipendenti
  - Visualizza lista su EI-4

L'utente posiziona il cursore sul campo per il quale vuole l'help e preme **F1** per visualizzare il testo di help relativo a tale campo.

### Passo 1

#### Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?

Sì.

**Passo 2****Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	No. Questa funzione esegue la stessa funzionalità descritta nel precedente esempio. I DET, FTR e trattamento logico per questa funzione sono gli stessi del precedente esempio. Pertanto non è possibile contarla nuovamente.
--	---

**Conclusione** L'Help di Campo – Seconda Occorrenza non rappresenta pertanto un processo elementare univoco e non viene contato come un'EQ.



## Esempio: Interrogazione implicita

### Requisiti utente

L'utente richiede la possibilità di visualizzare le informazioni sull'assegnazione (interrogazione diretta). L'utente richiede anche la possibilità di modificare le informazioni sull'assegnazione del lavoro. Sebbene non sia esplicitamente dichiarato, è implicito che le informazioni sull'assegnazione del lavoro debbano essere reperite prima di poter essere modificate.

### Schermate di Esempio

Il seguente diagramma illustra la finestra Modifica Assegnazione Lavoro con solamente il Nome Impiegato e Codice Lavoro.

The screenshot shows a multi-layered window titled 'Sistema Risorse Umane'. The top menu bar includes 'Impiegato', 'Lavori', 'Assegnazioni', 'Sedi', 'Prospett', 'Sicurezza', and 'Help'. The main window is titled 'Assegnazioni Impiegato' and contains a sub-window titled 'Modifica Assegnazione Lavoro'. Inside this sub-window, there are input fields for 'Mark', 'J', and 'Manship'. Below these are two empty input fields. Further down, there are labels for 'Cod Lavoro', 'Data', 'Retribuzione', and 'Valutazione Prestazioni', each followed by an input field. The 'Cod Lavoro' field contains the text 'RD15379305' and has a small downward arrow icon to its right. To the right of the input fields, there is a label 'Ed. Maggio - Stampa Cop.' and four yellow buttons: 'OK', 'Annulla', 'Ripristina', and 'Cancella'.

AE-5

OK	Esegue modifiche, ritorna a AE-1
Annulla	Ignora modifiche, ritorna a AE-1
Ripristina	Ripristina valori precedenti
Cancella	Richiede conferma, poi cancella l'assegnazione lavoro

Quando l'utente inserisce il Nome Impiegato e il Codice Lavoro, appaiono le informazioni sul lavoro come illustra il seguente diagramma.

**Sistema Risorse Umane**

Impiegato Lavori Assegnazioni Sedi Prospetti Sicurezza Help

**Assegnazioni Impiegato**

**Modifica Assegnazione Lavoro**

Mark J Manship

345-67-8901

Stab. principale UFPCA

Cod Lavoro RD15379305 Ed. Maggio - Stampa Cop.

Data 03/27/93

Retribuzione 17.28

Valutazione Prestazioni Adeguato

OK Annulla Ripristina Cancella

AE-5

OK Esegue modifiche, ritorna a AE-1

Annulla Ignora modifiche, ritorna a AE-1

Ripristina Ripristina valori precedenti

Cancella Richiede conferma, poi cancella l'assegnazione lavoro

### Passo 1 Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

### Passo 2 Determina l'univocità del Processo Elementare

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Non si è contato finora un processo elementare che esprimesse questa funzionalità. Se una interrogazione esplicita esiste, allora questa dovrà essere considerata quale duplicato e non dovrà essere contata.
--	---

**Passo 3**      **Classifica ciascun Processo Elementare**

<b>Regole di conteggio dell'EQ</b>	<b>Si applica la regola?</b>
2. Ha l'intento primario di presentare informazioni all'utente e	Sì. L'interrogazione implicita 'Assegnazione Lavoro' ha l'intento primario di presentare informazioni all'utente.
<ul style="list-style-type: none"> <li>• riferenzia una funzione di tipo dati per reperire dati o informazioni di controllo e</li> </ul>	Sì. I dati sono reperiti dagli ILF Assegnazione Lavoro, Impiegato e Lavoro.
<ul style="list-style-type: none"> <li>• non soddisfa i criteri per essere classificato come un EO</li> </ul>	Sì. Non sono eseguiti calcoli, non sono create dati derivati e non è aggiornato nessun ILF.

**Conclusione**    L'interrogazione implicita 'Assegnazione di Lavoro' è un'EQ.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

<b>Regole di conteggio degli FTR</b>	<b>Si applica la regola?</b>
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)-	Gli ILF Assegnazione di Lavoro, Impiegato e Lavoro sono letti. Per definizione, un'EQ non può mantenere un ILF.

**Passo 5****Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Nome, Codice Fiscale, Sede, Contratto di Lavoro, Codice Lavoro, Titolo Lavoro, Data di Inizio Attività, Retribuzione, e Valutazione Prestazioni.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Sì. Il tasto di comando.
4. Non contare i seguenti elementi come DET: <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Costanti quali il titolo della schermata non sono contate.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6****Determina la complessità funzionale**

3 FTR e 10 DET	La complessità è Media
----------------	------------------------

**Passo 7****Determina la dimensione funzionale**

La dimensione funzionale di 1 EQ a Media Complessità	4 FP
--	------

**Conclusione**

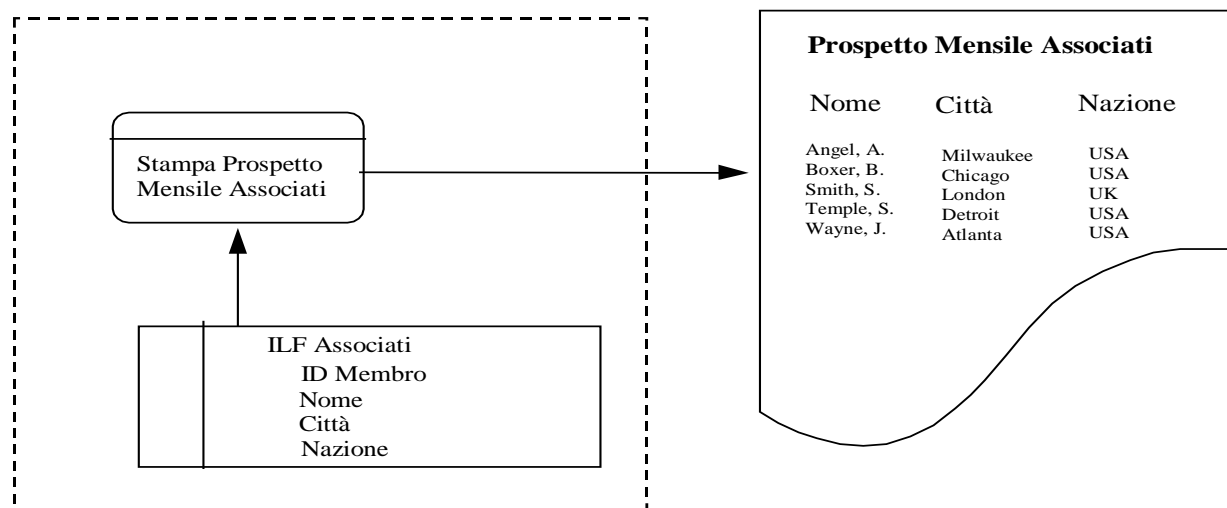
L'interrogazione implicita 'Assegnazione di Lavoro' è un processo elementare unico e pertanto è cantata come un'EQ. Nel caso in cui una interrogazione esplicita già esista, sarebbe considerato un duplicato e pertanto non sarebbe nuovamente conteggiata.

## Esempio: EQ innescata senza dati che entrano nel confine

### Requisiti utente

L'utente richiede che l'applicazione stampi automaticamente ogni mese il Report Mensile Associati.

Il seguente diagramma illustra il flusso dei dati per questo esempio.



### Passo 1

#### Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?

Sì.

### Passo 2

#### Determina l'univocità del Processo Elementare

La funzione transazionale è univoca rispetto agli altri processi elementari?

Sì. Nessun altro processo elementare esegue questa funzionalità.

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EQ	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. Il Report Mensile Associati ha l'intento primario di presentare informazioni all'utente.
<ul style="list-style-type: none"><li>• referenzia una funzione di tipo dati per reperire dati o informazioni di controllo e</li></ul>	Sì. I dati sono reperiti dall'ILF Associati.
<ul style="list-style-type: none"><li>• non soddisfa i criteri per essere classificato come un EO</li></ul>	Sì. Non sono eseguiti calcoli, non sono create dati derivati e non è aggiornato nessun ILF.

**Conclusione**    Il Reporto Mensile Associati è un'EQ. In questo esempio, la transazione è attivata da un evento temporale interno al confine.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)-	L'ILF Associati è letto. Per definizione, un'EQ non può mantenere un ILF.

**Passo 5****Conteggia i Tipi di Elementi Dati (DET)**

<b>Regole di conteggio dei DET</b>	<b>Si applica la regola?</b>
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Nome, Città, Nazione.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Non esistono elementi di questo tipo. La funzione viene eseguita automaticamente ogni mese.
4. Non contare i seguenti elementi come DET: <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Costanti quali "Nome" non sono contate.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6****Determina la complessità funzionale**

1 FTR e 3 DET	La complessità è Bassa
---------------	------------------------

**Passo 7****Determina la dimensione funzionale**

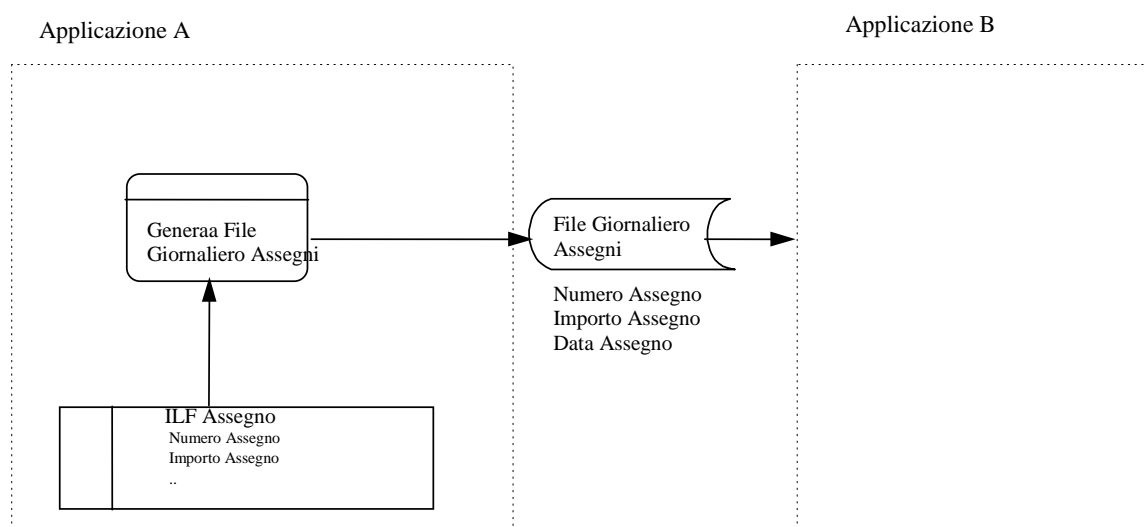
La dimensione funzionale di 1 EQ a Bassa Complessità	3 FP
--	------

## Esempio: Dati inviati ad un'altra applicazione

### Requisiti utente

Alla fine di ogni giornata, inviare un file di transazione all'Applicazione B elencando i numeri degli assegni e l'importo di ogni assegno stampato nel giorno.

Il seguente diagramma illustra il flusso dei dati per questo esempio.



### Passo 1

#### Identifica il Processo Elementare

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

### Passo 2

#### Determina l'univocità del Processo Elementare

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--



**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EQ	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. Il file Assegni Giornalieri ha l'intento primario di presentare informazioni. In questo caso l'utente è l'Applicazione B.
<ul style="list-style-type: none"><li>• referencia una funzione di tipo dati per reperire dati o informazioni di controllo e</li></ul>	Sì. I dati sono reperiti dall'ILF Assegni.
<ul style="list-style-type: none"><li>• non soddisfa i criteri per essere classificato come un EO</li></ul>	Sì. Non sono eseguiti calcoli, non sono create dati derivati e non è aggiornato nessun ILF.

**Conclusione**    Il File Assegni Giornalieri è un'EQ.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)-	L'ILF Assegni è letto. Per definizione, un'EQ non può mantenere un ILF.

**Passo 5**      **Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	Numero Assegno, Importo Assegno, Data Assegno.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Non esistono elementi di questo tipo. La funzione viene eseguita automaticamente ogni mese.
4. Non contare i seguenti elementi come DET: <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo.

**Passo 6**      **Determina la complessità funzionale**

1 FTR e 3 DET	La complessità è Bassa
---------------	------------------------

**Passo 7**      **Determina la dimensione funzionale**

La dimensione funzionale di 1 EQ a Bassa Complessità	3 FP
--	------

## Esempio: Funzionalità di Help aggiuntiva

### Requisiti utente

Durante la realizzazione del Sistema Risorse Umane (RU), è stato formulato un ulteriore requisito relativo ad una funzionalità di help aggiuntivo. L'informazione di help viene mantenuta da un'applicazione distinta. L'informazione di help viene referenziata dalle applicazioni Risorse Umane, Valuta, Beni Patrimoniali e Sistema Benefici.

### Processo di Conteggio

Il seguente diagramma mostra la schermata 'Dati Impiegato'.

— Sistema Risorse Umane —

Impiegato Lavori Assegnazioni Sedi Prospetti Sicurezza Help

— Lista Impiegati

— Dati Impiegato

Mark J Manship

345-67-8901

— Dati Impiegato ad Ore

Paga Oraria

Contratto di Lavoro ↓ Dipendenti

EI-3H

Dipendenti Visualizza lista su EI-4

Quando l'utente seleziona la funzione di Help nella parte alta di ciascuna finestra, si rendono disponibili tre scelte: Schermata, Cerca, Informazioni su.

**Passo 1**      **Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì, Finestra, Cerca e Informazioni su sembrano eseguire tre funzioni separate.
---	--

**Passo 2**      **Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue tali funzioni. Queste funzioni sono contate solo una volta nell'ambito di un'applicazione.
--	--

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EQ	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. Finestra, Cerca e Informazioni su hanno tutte l'intento primario di presentare informazioni all'utente
<ul style="list-style-type: none"> <li>• riferenzia una funzione di tipo dati per reperire dati o informazioni di controllo e</li> </ul>	<p>I dati sulla Finestra sono reperiti dall'EIF Help in base alla finestra correntemente visualizzata.</p> <p>'Cerca' abilita l'utente ad inserire un argomento e a ricercare i dati su tale argomento dall'EIF Help.</p> <p>'Informazioni su' reperisce informazioni statiche non mantenute nell'EIF Help. Pertanto non risponde alla definizione di un'EQ.</p>
<ul style="list-style-type: none"> <li>• non soddisfa i criteri per essere classificato come un EO</li> </ul>	Sì. Non sono eseguiti calcoli, non sono create dati derivati e non è aggiornato nessun ILF.

**Conclusione** L'Help di Finestra è un'EQ e la capacità di ricerca in 'Cerca' rappresenta un ulteriore EQ. 'Informazioni su' non viene contato perchè non è mantenuto da attraverso un processo elementare.

È importante assicurarsi che non si conti un Help statico. Ambienti ed applicazioni popolari, specialmente di tipo web, spesso contengono help statici.

È altresì importante assicurarsi che i confini siano correttamente valutati. La funzionalità di help è spesso fornita da un'applicazione esterna, quale ad es: RoboHelp, che fornisce l'intera gestione della funzionalità di help (es: manutenzione, memorizzazione e presentazione degli help). In questo caso specifico, nessuna funzionalità è stata dichiarata con riferimento all'applicazione a cui l'help fa riferimento.

L'Help di Finestra e l'Help Cerca sono analizzate separatamente nel seguito .

**Passo 4****Conteggia il numero di Tipi di File Referenziati (FTR) Help di Finestra**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)-	L'EIF Help è letto. Per definizione, un'EQ non può mantenere un ILF.

**Passo 5****Conteggia i Tipi di Elementi Dati (DET) per elp di Finestra**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	ID Finestra, Descrizione della finestra (solo testuale).
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Sì. La selezione dal menu di 'Help'.
4. Non contare i seguenti elementi come DET: <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	Costanti come le etichette di campo non sono contate Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo. Non esistono elementi di questo tipo.

**Passo 6****Determina la complessità funzionale per Help di Finestra**

1 FTR e 3 DET	La complessità è Bassa
---------------	------------------------

**Passo 7****Determina la dimensione funzionale per Help di Finestra**

La dimensione funzionale di 1 EQ a Bassa Complessità	3 FP
--	------

**Passo 4 Conteggia il numero di Tipi di File Referenziati (FTR) Funzione Cerca**

Regole di conteggio degli FTR	Si applica la regola?
2. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)-	L'EIF Help è letto. Per definizione, un'EQ non può mantenere un ILF.

**Passo 5 Conteggia i Tipi di Elementi Dati (DET) per Funzione Cerca**

Regole di conteggio dei DET	Si applica la regola?
5. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	ID Finestra, Argomento inserito, Messaggi di Help riportati.
6. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Non esistono elementi di questo tipo.
7. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Sì. La selezione dal menu di 'Help'.
8. Non contare i seguenti elementi come DET: <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Costanti come le etichette di campo non sono contate</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6 Determina la complessità funzionale per Funzione Cerca**

1 FTR e 3 DET	La complessità è Bassa
---------------	------------------------

**Passo 7 Determina la dimensione funzionale per Funzione Cerca**

La dimensione funzionale di 1 EQ a Bassa Complessità	3 FP
--	------

## Esempio: Logon di sicurezza

### Requisiti utente

L'utente desidera una funzionalità di logon per il controllo della sicurezza delle schermate. La funzionalità è illustrata nella seguente immagine.

La funzionalità di logon richiede l'inserimento dei seguenti campi:

- Identificativo Utente
- Password

Quando l'utente intende accedere al sistema, il file Sicurezza viene letto per validarne l'id e la password, al fine di stabilire quali siano le schermate a cui è abilitato all'accesso e modifica

Saranno generati messaggi di errore e sottolineature per i campi non correttamente compilati. Due (2) messaggi di errore e un (1) messaggi di conferma sono inclusi nella transazione per il logon di sicurezza.

### Schermata di Logon di Sicurezza

La seguente figura mostra la finestra per accedere al sistema.

Sistema Risorse Umane	
Logon di Sicurezza	
ID Utente	<input type="text"/>
Password	<input type="password"/>

L'utente preme il tasto 'Invio' dopo che tutte le informazioni sono state immesse. L'Id. Utente e la password sono validate e ovviamente tutti gli accessi utente sono assegnati

**Passo 1**      **Identifica il Processo Elementare**

La funzione di tipo transazionale soddisfa i requisiti di un processo elementare?	Sì.
---	-----

**Passo 2**      **Determina l'univocità del Processo Elementare**

La funzione transazionale è univoca rispetto agli altri processi elementari?	Sì. Nessun altro processo elementare esegue questa funzionalità.
--	--

**Passo 3**      **Classifica ciascun Processo Elementare**

Regole di conteggio dell'EQ	Si applica la regola?
1. Ha l'intento primario di presentare informazioni all'utente e	Sì. L'intento primario è presentare informazioni all'utente.
<ul style="list-style-type: none"> <li>• riferenzia una funzione di tipo dati per reperire dati o informazioni di controllo e</li> </ul>	Sì. I dati sono reperiti dall'ILF Sicurezza.
<ul style="list-style-type: none"> <li>• non soddisfa i criteri per essere classificato come un EO</li> </ul>	Sì. Non sono eseguiti calcoli, non sono create dati derivati e non è aggiornato nessun ILF.

**Conclusione**      Il Logon di Sicurezza è un'EQ.

**Passo 4**      **Conteggia il numero di Tipi di File Referenziati (FTR)**

Regole di conteggio degli FTR	Si applica la regola?
1. Conta un FTR per ogni funzione dati unica alla quale la funzione transazionale accede (in lettura e/o scrittura)-	L'ILF Sicurezza è letto. Per definizione, un'EQ non può mantenere un ILF.



**Passo 5 Conteggia i Tipi di Elementi Dati (DET)**

Regole di conteggio dei DET	Si applica la regola?
1. Conta un DET per ogni attributo riconoscibile dall'utente, non ripetuto, che entra o esce dal confine dell'applicazione durante l'esecuzione della funzione transazionale.	ID Utente, Password.
2. Conta un solo DET per la capacità di inviare un messaggio di risposta del sistema all'esterno del confine dell'applicazione anche nel caso di messaggi multipli	Messaggi di Errore e Conferma.
3. Conta un solo DET per la capacità di specificare un'azione da intraprendere, anche qualora vi siano molteplici modi per richiamare lo stesso trattamento logico.	Tasto 'Invio'.
4. Non contare i seguenti elementi come DET: <ul style="list-style-type: none"> <li>• costanti</li> <li>• notazioni generate in modo automatico</li> <li>• variabili di impaginazione</li> <li>• aiuti navigazionali</li> <li>• attributi generati senza uscire dal confine e</li> <li>• attributi reperiti o referenziati da un ILF o EIF che non escono dal confine</li> </ul>	<p>Costanti come le etichette sulle schermate non sono conteggiati.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p> <p>Non esistono elementi di questo tipo.</p>

**Passo 6 Determina la complessità funzionale per la Capacità di Ricerca**

1 FTR e 4 DET	La complessità è Bassa
---------------	------------------------

**Passo 7 Determina la dimensione funzionale per la Capacità di Ricerca**

La dimensione funzionale di 1 EQ a Bassa Complessità	3 FP
--	------

Questa pagina è lasciata intenzionalmente vuota.

---

## Indice analitico – Parte 4

---

**A**

Esempio di indice secondario, 1-20

Applicazione

    esempio di conteggio dei menu, 2-129

**B**

Batch con EI multipli e duplicati, 2-70

**C**

Informazioni di controllo

    per reportistica, 2-63

Modello per la conversione dati, 1-43

Informazioni di conversione, 1-43

Convertire dati in un nuovo formato con elementi  
    data aggiuntivi, 2-80

**D**

Conversione di dati, 1-58, 2-80

Modello dati per la conversione di dati, 1-58

Diagrammi

    componenti negli esempi di conteggio per

        EI/EO/EQ, 2-3

    componenti negli esempi di conteggio per ILF/EIF,  
        1-5

    modello di conversione di dati, 1-43

    modello dati per la conversione di dati, 1-58

    Employee Data for field help input side of inquiry,  
        2-143

    Schermata Dati Assegnazione Lavoro, 2-76

    Schermata Job Data input, 2-67

    menu per il lato input dell'interrogazione, 2-129

    reportistica interattiva, 2-108

    organizzazione degli esempi di conteggio per  
        EI/EO/EQ, 2-2

    esempio di formato record per layout transaction  
        input file, 1-60

**E**

EI con FTR multipli, 2-76

Esempi di conteggio per EI/EO/EQ, 2-2

Help di campo sui dati dell'impiegato, 2-143

Esempi

    indice secondario, 1-20

    menu dell'applicazione menus, 2-129

    batch con EI multipli e duplicati, 2-70

    informazioni di controllo, 2-63

    informazioni sulla conversione, 1-43

    conversioni dati in un nuovo formato, 2-80

    conversione dati, 1-58, 2-80

    diagramma di flusso per contare riferimento a file  
        multipli, 2-76

    EI con FTR multipli, 2-76

    help di campo, 1-50

    report mediante hard copy, 2-105

    applicazione di help, 1-50

    transazione di inserimento interattiva, 2-67

    fornire dati ad un'altra applicazione, 1-48

    riferimento a file multipli, 2-76

    riferimento a dati di un'altra applicazione, 1-43, 2-  
        83

    riferimento a dati di altre applicazioni, 1-39

    report mediante hard copy, 2-105

    input a video, 2-67

    Descrizioni sintetiche degli esempi di conteggio  
        degli EI, 2-62

    Descrizioni sintetiche degli esempi di conteggio  
        degli EO, 2-104

    transazione con tipi record formattati, 2-70

    transazione con tipi multipli, 2-70

    help di finestra, 1-50

## External inputs (EI)

- batch con EI multipli e duplicati, 2-70
- informazioni di controllo, 2-63
- esempio di conversione di dati, 2-80
- EI con FTR multipli, 2-76
- esempio di input a video, 2-67

## External inquiries (EQ)

- esempio di menu dell'applicazione, 2-129

## External interface files (EIF)

- informazioni di conversione, 1-43
- esempio di conversione dati, 1-58
- help a livello di campo, 1-50
- esempio di applicazione di Help, 1-50
- fornire dati ad altre applicazioni, 1-48
- riferimento a dati di altre applicazioni, 2-83
- riferimento a dati di un'altra applicazione, 1-39, 1-43,
- help di finestra, 1-50

## External outputs (EO)

- esempio di prospetto di output, 2-105

**F**

## Help a livello di campo, 1-50

## Tipi record formattati, 2-70

**H**

## Hard copy report, 2-105

## Applicazione di Help, 1-50

## Schermata Sistema Risorse Umane, 2-129

**I**

## Internal logical files (ILF)

- esempio di indici secondari, 1-20
- Descrizioni sintetiche degli esempi, 1-8, 2-8

**J**

## Schermata Dati Assegnazione Lavoro, 2-76

**M**

## Manuale

- organizzazione degli esempi di conteggio per EI/EO/EQ, 2-2
- organizzazione per esempi di conteggio per ILF/EIF, 1-4

## Tipi record multipli, 2-70

**N**

## Nuovo formato con elementi dati aggiuntivi, 2-80

**O**

## Online add transaction via a screen, 2-67

## Online reporting screen, 2-108

## Organizzazione. Vedi Manuale

- esempi di conteggio per EI/EO/EQ, 2-2

**P**

## Fornire dati ad altre applicazioni, 1-48

**R**

## Record layout for input file transaction, 1-60

## Tipi record, 2-70

## Riferito a file multipli, 2-76

## Riferimento a dati di un'altra applicazione, 1-43, 2-83

## Riferimento a dati da un'altra applicazione, 1-39

## Esempio di report di output, 2-105

## Report

- mediante hard copy, 2-105

**S**

## Screen input, 2-67

## Schermate

- Impiegati per durata assegnazione, 2-108

- reportistica interattiva, 2-108

## Descrizioni sintetiche degli esempi di conteggio degli EI, 2-62

## Descrizioni sintetiche degli esempi di conteggio degli EO, 2-104

## Descrizioni sintetiche degli esempi di conteggio degli ILF, 1-8, 2-8

**T**

## Transazione con tipi record formattati, 2-70

## Transazione con tipi multipli, 2-70

**W**

## Help di finestra, 1-50

## Finestre

- help di campo, 2-143

---

## **Parte 5 – Appendici e Glossario**

---

---

Questa pagina è lasciata intenzionalmente vuota.

---

### Appendice A: Tabella per il calcolo della dimensione funzionale

---

**Introduzione** Questa appendice include una tabella per facilitare la misura della dimensione funzionale.

**Contenuti** Questa appendice include la seguente tabella:

Argomento	Pagina
Tabella per il Calcolo della Dimensione Funzionale	A-2

## Tabella per il calcolo della dimensione funzionale

La seguente tabella è fornita per per facilitare il calcolo del contributo alla dimensione funzionale.

Tipo Funzione	Complessità Funzionale		Totale Complessità	Totale Tipo Funzione
ILF	Bassa	X 7 =		
	Media	X 10 =		
	Alta	X 15 =		
EIF	Bassa	X 5 =		
	Media	X 7 =		
	Alta	X 10 =		
EI	Bassa	X 3 =		
	Media	X 4 =		
	Alta	X 6 =		
EQ	Bassa	X 3 =		
	Media	X 4 =		
	Alta	X 6 =		
EO	Bassa	X 4 =		
	Media	X 5 =		
	Alta	X 7 =		
Totale Dimensione Funzionale				



---

### Appendice B: Cambiamenti dalla precedente versione

---

**Introduzione** Questa appendice contiene informazioni su variazioni e miglioramenti inclusi nel CPM 4.3, il processo decisionale e le raccomandazioni per gli utenti del nuovo manuale.

**Contenuti** Questo capitolo include le seguenti sezioni:

Argomento	Pagina
<b>Introduzione</b>	B-2
<b>Aree di maggior cambiamento strutturale nel CPM 4.3</b>	B-3
<b>Controllo di versione</b>	B-3
<b>Panoramica delle modifiche</b>	B-4
<b>Processo decisionale</b>	B-11
<b>Studio d'impatto</b>	B-11
<b>Conversione dal CPM 4.2 al CPM 4.3</b>	B-11
<b>Impatto su Utenti 4.2.1 che passano alla versione 4.3</b>	B-12
<b>Raccomandazioni</b>	B-12

## Introduzione

Dopo la pubblicazione della versione 4.2 del Manuale sulle Regole del Conteggio (CPM – Counting Practices Manual) IFPUG nel 2004, il Comitato per le Regole del Conteggio (CPC – Counting Practices Committee) ha creato una nuova versione della Parte 1 (Regole) per sostituire lo Standard ISO (ISO 20926:2004); cioè IFPUG CPM 4.1 non pesata. La creazione del nuovo standard ISO ha richiesto dei cambiamenti nel testo nelle rimanenti parti (guida all'implementazione) per mantenerne la consistenza.

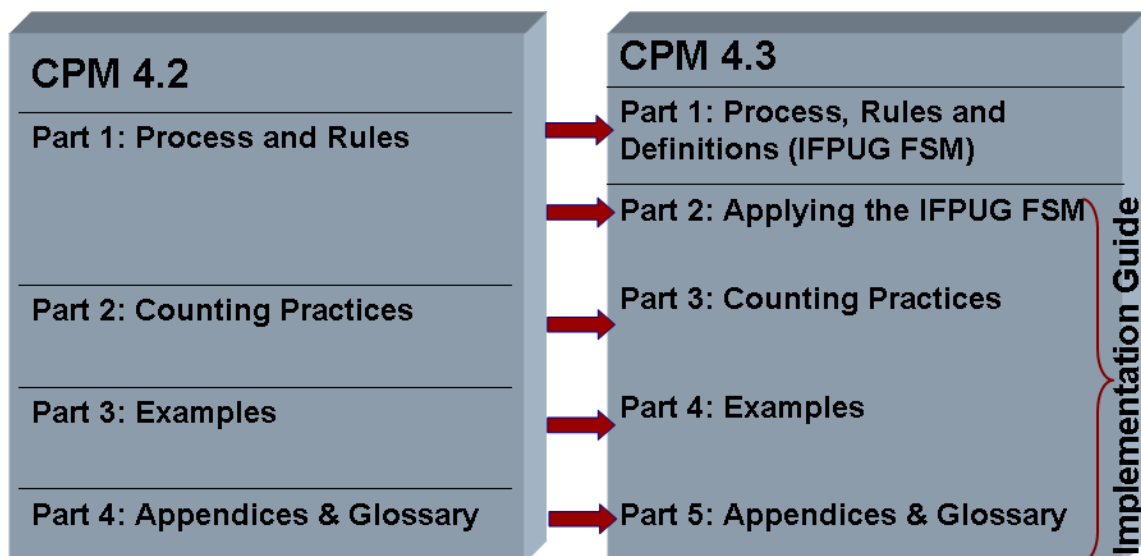
Il CPC, per revisionare il CPM, esegue il seguente processo:

1. Il suggerimento di modifica viene inoltrato al CPC dai membri IFPUG.
2. Il suggerimento di modifica viene assegnato a membri del CPC per l'analisi.
3. Il CPC riesamina e discute il suggerimento.
4. Il CPC presenta la proposta di soluzione ai membri IFPUG.
5. Viene avviato uno studio d'impatto.
6. Viene presa la decisione finale.
7. I membri IFPUG vengono informati della decisione attraverso articoli su MetricViews e presentazioni agli eventi IFPUG.
8. I cambi diventano effettivi con l'emissione di un nuovo CPM.
9. I Casi di Studio vengono aggiornati per renderli aderenti al nuovo CPM.

## Aree di maggior cambiamento strutturale nel CPM 4.3

Le aree di maggior cambiamento strutturale nel CPM 4.3 sono:

- Sostituzione della Parte 1 esistente con il nuovo Standard ISO (ISO/IEC 20926:2009)
- Creato ex-novo il “Collegamento – Applicare il Metodo IFPUG per la Misura della Dimensione Funzionale” (ora Parte 2) che fornisce indicazioni nell’applicazione dei processi e delle regole come definito dallo standard ISO (ora Parte 1)
- Miglioramento delle parti rimanenti per renderle consistenti con la revisione della Parte 1
  - Prassi di Conteggio (Parte 3)
  - Esempi (Parte 4)
  - Appendici e Glossario (Parte 5)



## Controllo di versione

Il CPC ha scelto di denominare questa versione del CPM IFPUG quale ‘4.3’ invece di ‘4.2.2’ o ‘5.0’, per due motivi:

- Un numero di versione ‘4.2.2’ avrebbe suggerito esclusivamente correzioni tipografiche; il numero di versione ‘4.3’ sottolinea maggiormente invece la riscrittura della Parte 1.
- Un numero di versione ‘5.0’ avrebbe suggerito una modifica delle regole di conteggio. Il CPM 4.3 rappresenta invece ancora una evoluzione della metodologia di Albrecht che è alla base di tutti i precedenti CPM IFPUG. Essa fornisce ulteriori chiarimenti rispetto alla precedente versione.

## Panoramica delle modifiche

Oltre alla sostituzione della attuale Parte 1, relativa allo Standard ISO, nel CPM 4.3 sono stati inclusi ulteriori chiarimenti di minor impatto. Al fine di facilitare gli utenti, che desiderano allineare il proprio CPM nel proprio idioma, si fornisce nel seguito un elenco di tutti i cambiamenti apportati.

### Parte 1: Processo e Regole

Parte della terminologia è stata rivista in modo da rendere conforme la Guida all'implementazione IFPUG alle modifiche allo Standard ISO sui metodi FSM (ISO/IEC 14143-1:2007). Le regole e le guide linea sono essenzialmente rimaste invariate; in ogni caso la sequenza delle azioni e le enunciazioni sono lievemente variate. Non ci si aspetta che la dimensione funzionale possa cambiare. Tutti i capitoli nella ex Parte 1 (ora Parte 2) includono cambiamenti nelle enunciazioni, ulteriori esempi e suggerimenti in modo tale da essere coerenti e conformi agli aggiornamenti dello Standard ISO FSM, riemesso nel 2007 e che ora è nella Parte 1. Le GSC ed il Fattore di Aggiustamento del Valore (VAF – Value Adjustment Factor) sono stati eliminati da questa Parte ed inseriti nelle appendici in modo tale da essere conformi con lo Standard ISO FSM, che non li riconosce come facenti parte della misurazione della dimensione funzionale (FSM – Functional Size Measurement). Ulteriori dettagli sono riportati nel seguito.

#### Parte 1, Capitolo 1: Introduzione

Il titolo di questo capitolo è stato cambiato in “Collegamento – Applicare il metodo IFPUG per la Misura della Dimensione Funzionale”. Il Capitolo 1 contiene solo un'introduzione; i cambiamenti riflettono i contenuti dei rimanenti capitoli nella nuova Parte 2.

#### Parte 1, Capitolo 2: Generalità sull'Analisi dei Function Point

Il titolo di questo capitolo è stato cambiato in “Panoramica sul metodo FSM IFPUG”. Sono stati effettuati molti cambiamenti alle enunciazioni al fine di garantire la coerenza e la conformità con gli aggiornamenti apportati nello Standard ISO FSM. Le seguenti definizioni e regole sono state leggermente riformulate.

Il Diagramma della Procedura e le istruzioni sono state modificate al fine di rispecchiare che il primo passo nella procedura di conteggio dei function point è di collezionare la documentazione disponibile, in accordo con lo Standard ISO FSM.

La dimensione funzionale rappresenta ora una dimensione del software derivata dalla quantificazione dei requisiti utente funzionali (FUR – Functional User Requirement) rimpiazzando il termine “function point non pesati” (UFP). Ogni discussione su “non pesato” o “pesato” è ora inserita nelle appendici in modo da essere conforme con lo Standard ISO FSM, che non riconosce le GSC o il VAF come facenti parte della misurazione della dimensione funzionale (FSM).

#### Parte 1, Capitolo 3: Punto di Vista dell'Utente

Il titolo di questo capitolo è stato cambiato in “Raccogliere la documentazione disponibile”. Questo capitolo presenta il concetto del ruolo dell'utente e l'approccio alla misurazione durante il ciclo di vita di un'applicazione; comunque, l'intero capitolo nella sostanza rimane immutato ad eccezione del titolo.

### **Parte 1, Capitolo 4: Determinare il Tipo di Conteggio**

Sono state effettuate modifiche alle enunciazioni per le definizioni di conteggio dei function point per un progetto di sviluppo, conteggio di function point per un progetto di manutenzione evolutiva e conteggio di function point per un'applicazione, allo scopo di renderli conformi e coerenti con l'aggiornamento dello Standard ISO FSM.

### **Parte 1, Capitolo 5: Identificare l'Ambito del Conteggio ed il Confine dell'Applicazione**

Il titolo di questo capitolo è stato modificato in “Determinare l'ambito del conteggio e il confine e identificare i Requisiti Utente Funzionali”. Alcuni enunciati sono stati leggermente variati per mantenere la coerenza e la conformità con gli aggiornamenti dello Standard ISO FSM, ma la maggior parte del capitolo è rimasta invariata.

### **Parte 1, Capitolo 6: Calcolare le Funzioni di Tipo Dati**

Il titolo di questo capitolo è stato cambiato in “Misurare le funzioni di tipo dati” per rispecchiare che le regole sono attualmente contenute nella nuova Parte 1 e che tale capitolo fornisce una guida all'implementazione per la misura delle funzioni di Tipo Dati. Le regole contenute sono ripetute dalla Parte 1 al fine di rendere maggiormente fruibile l'utilizzo del CPM ed evitare la necessità di fare avanti ed indietro tra le parti.

### **Parte 1, Capitolo 7: Calcolare le Funzioni di Tipo Transazionale**

Il titolo di questo capitolo è stato cambiato in “Misurare le funzioni di tipo transazionale” per riflettere che le regole sono attualmente contenute nella nuova Parte 1 e per riflettere che questo capitolo fornisce una guida all'implementazione per la misura delle funzioni di Tipo Transazionale. Come nel Capitolo 6, le regole contenute sono ripetute dalla Parte 1 al fine di rendere maggiormente fruibile l'utilizzo del CPM ed evitare la necessità di fare avanti ed indietro tra le parti. Tra gli aspetti specifici si menzionano:

- Indicazioni addizionali e chiarimenti circa le regole FSM per i processi elementari.
- Semplificazione delle regole di conteggio dei DET e FTR.

### **Parte 1, Capitolo 8: Determinare il Fattore di Aggiustamento del Valore**

L'intero contenuto di questo capitolo è ora posizionato nell'Appendice C con l'obiettivo di allineare la Guida all'Implementazione con lo Standard IFPUG FSM, che non include le GSC ed il VAF.

### **Parte 1, Capitolo 9: Calcolare il Numero di Function Point Pesati**

Al fine di allineare la Guida all'Implementazione con l'IFPUG FSM, che non include le GSC ed il VAF, le formule contenute precedentemente in questo capitolo sono state riposizionate ora nell'Appendice C e nella Parte 3 - Capitolo 4 (Progetti di Manutenzione Evolutiva e Attività di Manutenzione Ordinaria).

## Parte 2: Prassi di Conteggio

Tutti i capitoli della Parte 2 (ora Parte 3) contengono lievi modifiche alle enunciazioni al fine di rendere consistente la riorganizzazione della Parte 1 e/o essere conformi all'aggiornamento dello Standard ISO FSM. E' stato aggiunto un nuovo capitolo per fornire informazioni per il conteggio delle Attività di Conversione dei Dati (Capitolo 5). Ulteriori dettagli sono riportati di seguito.

### Parte 2, Capitolo 1: Dati di Decodifica

La metodologia FPA IFPUG è conforme con lo Standard ISO FSM 14143-1. La decisione di non contare i Dati di Decodifica e di creare il capitolo "Dati di Decodifica" nella Parte 2 del CPM 4.2, trae origine dai requisiti presenti nello Standard ISO FSM (ISO/IEC 14143-1:1998) di non includere nel conteggio i requisiti tecnici e qualitativi.

Nel 2007 la ISO ha rilasciato una nuova versione degli Standard FSM (ISO/IEC 14143-1:2007). Di conseguenza, è stato necessario aggiornare il capitolo sui Dati di Decodifica per poter allineare le modifiche agli enunciati indicate nello Standard ISO FSM.

Non ci sono cambiamenti alle regole o alle indicazioni in questo capitolo, ma sono stati apportate lievi modifiche agli enunciati al fine di renderli conformi agli aggiornamenti dello Standard ISO FSM.

- Aggiunta la definizione ISO di 'Dimensione Funzionale'
- Aggiornata la definizione di 'Requisito Utente Funzionale' (FUR)
- Sostituiti i termini 'Requisiti sulla Qualità' e 'Requisiti Tecnici' con il termine ISO 'Requisiti Utente Non-Funzionali' ed inclusa la definizione ISO di tale concetto
- La sezione Metodologia è stata lievemente rivista per riflettere i cambiamenti nel passo "Identificare i File Logici" nel Capitolo 2 (File logici) illustrato di sotto

### Parte 2, Capitolo 2: File Logici

Questo capitolo è stato creato nel CPM 4.2 al fine di fornire procedure di conteggio e informazioni aggiuntive nell'identificazione e nella valutazione dei File Logici (LF – Logical File).

Nel CPM 4.3, la Parte 1 è stata sostituita dall'aggiornamento dello Standard IFPUG FPA ISO. .

Alcuni dei cambiamenti introdotti nello Standard ISO FSM hanno degli impatti (lievi) per il capitolo 'File Logici' (LF):

- Nel processo di identificazione dei File Logici, il precedente passo 1 ("Escludere i Dati di Decodifica prima di esaminare i File Logici") ora è parte del passo 1 "Identificare i file Logici", che rappresenta una collocazione maggiormente appropriata.
- Inoltre, il precedente passo 2 ("Identificare e Classificare i File Logici") è stato scomposto in due sotto-passi "1. Identificare i File Logici" e "2. Classificare i File Logici"
- I passi 3 e 4 ("Identificare RET e DET") sono stati invertiti
- I sotto-passi del passo 1 sono stati evidenziati maggiormente, esplicitandoli come sotto-passi

Questi cambiamenti alla struttura hanno alcuni (lievi) conseguenze per la struttura del capitolo dei 'File Logici'.

Ciò risulta essere particolarmente vero per l'inversione dei passi 'Identificare i DET' e 'Identificare i RET'. Si è reso quindi necessario invertire le pagine relative a tali passi. E' stato anche necessario mettere insieme le tabelle "Considerare i Tipi di Elemento Record congiuntamente ai File Logici tramite la (In)Dipendenza dell'Entità" (CPM 4.2, pagina 2-34) e "Considerare i Tipi di Elemento Dati, i File Logici e i Tipi di Elemento Record, tramite la (In-)Dipendenza dell'Entità" (CPM 4.2, pagina 2-46) in una nuova tabella "Esaminare i DET e i RET insieme ai File Logici tramite la (In-)Dipendenza delle Entità".

Sono state apportate lievi modifiche agli enunciati per renderli conformi agli aggiornamenti dello Standard ISO FSM (ISO/IEC 14143-1:2007) come è spiegato con maggior dettaglio nella sezione di sopra, dedicata alla Parte 2, Capitolo 1 Dati di Decodifica.

I Requisiti Qualitativi e i Requisiti Tecnici sono stati sostituiti dal nuovo termine ISO 'Requisiti Utente Non-Funzionali'.

Nessuno di tali cambiamenti alla struttura e agli enunciati dovrebbe influenzare il risultato dei conteggi.

### **Parte 2, Capitolo 3: Dati Condivisi**

Questo capitolo è stato creato nel CPM 4.2 al fine di fornire procedure di conteggio e informazioni aggiuntive nell'identificazione e nella valutazione dei dati condivisi tra le applicazioni.

I soli cambiamenti a questo capitolo sono due riferimenti ad altre parti del CPM che ora risultano differenti.

### **Parte 2, Capitolo 4: Progetti di Manutenzione Evolutiva e Attività di Manutenzione Ordinaria**

Questo capitolo è stato creato nel CPM 4.2 al fine di fornire procedure di conteggio e informazioni aggiuntive nell'applicare l'analisi dei function point da effettuare nelle attività di sviluppo. Il conteggio dei Progetti di Manutenzione Evolutiva, introdotto prima nella Parte 1, Capitolo 9 del CPM 4.1, è ora trattato interamente (incluse le formule da applicare) in questo capitolo.

Oltre agli aggiornamenti dei riferimenti alle altre parti del CPM, le definizioni e gli esempi principali per ciascuno degli aspetti del trattamento logico sono stati resi consistenti con quelli della nuova Parte 2. I termini sono stati resi consistenti anche con le Parti 1 e 2 così come il cambiamento del termine "campo" con "attributo".

Il paragrafo relativo al 'Trattamento Logico' in questo capitolo ha subito i seguenti cambiamenti:

3. Convertire valori equivalenti: modificato l'esempio in accordo ai commenti discussi nell'IFPUG Bulletin Board.
4. Filtrare i dati: Modificato l'esempio esistente per escludere dal conteggio un cambiamento che coinvolge solo la sostituzione o l'aggiunta di valori ed aggiunti tre nuovi esempi in accordo ai commenti discussi nell'IFPUG Bulletin Board.
11. Preparare e presentare informazioni all'esterno del confine: Aggiunti tre nuovi esempi per riflettere le risposte del CPC ai commenti discussi nell'IFPUG Bulletin Board.

12. Accettare dati o informazioni di controllo che entrano nel confine dell'applicazione: aggiunti due nuovi esempi per riflettere le risposte del CPC ai commenti discussi nell'IFPUG Bulletin Board.
13. Ordinamento: aggiunti due nuovi esempi per riflettere le risposte del CPC ai commenti discussi nell'IFPUG Bulletin Board.

Nel paragrafo 'Considerazioni e Suggerimenti', si sono aggiunte la discussione relativa alle funzionalità cancellate e spostati i suggerimenti relativi alle GSC in Appendice C, dove sono ora trattati gli aspetti opzionali delle GSC e del VAF.

Nel paragrafo 'Considerazioni sulla Manutenzione Evolutiva e sulla Manutenzione Ordinaria', ogni riferimento alle GSC è stato fatto precedere dal termine "opzionale".

## **Parte 2, Capitolo 5: Attività di Conversione dei Dati (nuovo capitolo)**

Questo nuovo capitolo risponde alla necessità di valutare le funzionalità relative alla presenza di requisiti per la migrazione o conversione di dati insieme per un progetto di nuovo sviluppo o di manutenzione evolutiva o per la migrazione di un'applicazione verso una piattaforma differente. La Parte 4 del CPM fornisce ulteriori esempi di funzioni di Tipo Dati e di Tipo Transazionale per la conversione dei dati.

## **Parte 3: Esempi**

Tutti i capitoli della Parte 3 (ora Parte 4) hanno incluso la revisione completa dei box di questa parte per essere consistenti con le modifiche agli enunciati delle regole delle funzioni di tipo dato, i processi elementari e delle funzioni transazionali. Ulteriori dettagli sono riportati nel seguito.

### **Parte 3, Capitolo 1: Esempi di Conteggio delle Funzioni di Tipo Dati**

- Esempio ILF: Dati di Controllo per Interrogazioni e Report – Rimossi i riferimenti alla manutenzione della Sicurezza Impiegato dal Diagramma di Flusso dei Dati perché creava confusione e non era spiegato a sufficienza.
- Esempio ILF: Definizione Report – Dettagliata la spiegazione del perché Definizione Report non è un'istanza di dati di decodifica.
- Esempio ILF: Dati Condivisi dell'Applicazione – Chiarito l'esempio per rendere più esplicito che la sicurezza descritta in questo esempio non è la sicurezza applicativa (cioè la determinazione di quale utente può accedere all'applicazione).
- Esempio EIF: Fornire Dati ad Altre Applicazioni – Ampliata la spiegazione, rimarcando il motivo per cui Tassi di Cambio non è un'istanza di dati di decodifica.
- Esempio EIF: Applicazione di Help – Ampliata la spiegazione, rimarcando il motivo per cui Help non è un'istanza di dati di decodifica; spiegato anche il motivo per cui Help di Finestra e Help di Campo sono funzioni di tipo dati distinte.

### **Parte 3, Capitolo 2: Esempi di Conteggio delle Funzioni di Tipo Transazionale**

- Esempio Processo Elementare (PE): Nuovo Impiegato / Informazioni sui Dipendenti – Ampliata la spiegazione, rimarcando il motivo per cui l'invio del file verso il Sistema Benefici è un processo elementare distinto.
- Esempio Processo Elementare (PE): Inserimento batch dati impiegati – Questo nuovo esempio illustra che la produzione batch di report di errore e report statistici non sono processi elementari



separati.

- Esempio Processo Elementare (PE): Assegna impiegato a lavoro – Questo nuovo esempio illustra come valutare processi elementari simili per determinare se sono univoci.
- Esempio Processo Elementare (PE): Processi elementari simili – Questo nuovo esempio illustra due processi elementari simili che sono contati come un'unica transazione.
- Esempio EI: EI con attributi reperiti da un EIF – Questo nuovo esempio illustra un EI con attributi reperiti da un EIF che non attraversano il confine dell'applicazione.
- Esempio EI: EI di cancellazione – Questo nuovo esempio illustra come contare i DET per una transazione di cancellazione.
- Esempio EI: Aggiungi sicurezza schermata – Questo nuovo esempio illustra il conteggio di funzionalità che mantengono la sicurezza applicativa.
- Esempio EO: EO innescato senza dati che entrano nel confine – Rinominato l'esempio per eliminare fraintendimenti.
- Esempio EQ: EQ innescato senza dati che entrano nel confine – Rinominato l'esempio per eliminare fraintendimenti.
- Esempio EQ: Funzionalità aggiuntiva di Help – Questo nuovo esempio illustra il conteggio di ulteriori funzionalità di Aiuto (Help).
- Esempio EQ: Sicurezza per gli accessi utente – Questo nuovo esempio illustra il trattamento sicurezza applicativa.
- Esempio EQ: Logon applicativo – Questo nuovo esempio illustra il conteggio della funzione di Login.

## Parte 4: Appendici e Glossario

Tutti i capitoli nella Parte 4 (ora Parte 5) sono stati rivisti. Il dettaglio è riportato nel capitolo seguente.

### Parte 4, Appendice A: Tabelle per i Calcoli

Lievi modifiche alle enunciazioni per eliminare l'uso del termine “non pesato” (*unadjusted*).

### Parte 4, Appendice B: Cambiamento dalla versione precedente

Questo capitolo include le seguenti sezioni:

- Aree di maggior cambiamento Strutturale nel CPM 4.3
- Informazioni sul controllo della versione
- Una panoramica delle modifiche per capitolo
- Il processo decisionale delle modifiche
- Il processo dello studio di impatto
- Impatto per gli utenti che passano alla versione 4.3
- Conversione dal CPM 4.2.1 al CPM 4.3
- Raccomandazioni per gli utenti che passano dalla versione 4.2.1 alla 4.3

### Parte 4, Appendice C: Modulo per Richiesta del Lettore

Il Modulo per la Richiesta del Lettore è stata eliminata. I Lettori possono suggerire cambiamenti inviando una email al CPC ([cpc@ifpug.org](mailto:cpc@ifpug.org)). Per l'Appendice C il titolo è stato modificato in “Dimensione Funzionale Pesata” ed ora contiene indicazioni per l'applicazione delle Caratteristiche Generali del Sistema (GSC) ed il fattore di aggiustamento del valore (VAF). Contiene tutte le formule che usano i GSC ed il VAF.

## Parte 4, Glossario

Sono stati aggiunti i seguenti termini nel glossario:

- Dimensione funzionale pesata di una applicazione (aAFP)
- Dimensione funzionale pesata di una applicazione dopo progetti di manutenzione evolutiva (aAFPA)
- Dimensione funzionale pesata di un progetto di sviluppo (aDFP)
- Dimensione funzionale pesata di un progetto di sviluppo (aEFP)
- Dimensione funzionale dell'Applicazione
- Riorganizzazione
- Componente Funzionale di Base (BFC)
- Confine
- Confine dell'Applicazione
- Stato di coerenza funzionale
- Dimensione funzionale di un progetto di sviluppo
- Dimensione funzionale di un progetto di manutenzione evolutiva
- Dimensione funzionale
- Significativo
- Intento Primario
- Autosufficiente
- Ordinamento
- Dimensione funzionale non pesata

Sono stati riesaminati i seguenti termini nel glossario:

- Manutenzione adeguativa
- Numero di function point pesati (AFP – Adjusted Function Point count) (\*)
- Applicazione
- Confine dell'applicazione
- Numero di function point dell'applicazione (AFP– Application Function Point)
- Contributo
- Informazioni di controllo
- Funzionalità di conversione
- Manutenzione correttiva
- Ambito del conteggio
- Dati derivati
- Sviluppo (\*)
- Numero di function point di un progetto di sviluppo (DFP - Development Function Point)
- Manutenzione evolutiva (\*)
- Numero di function point di un progetto di manutenzione evolutiva (EFP – Enhancement Function Point)
- Dipendenza dell'entità
- Indipendenza dell'entità
- File system
- Tipo di file referenziato (FTR)
- Complessità funzionale
- Requisiti utente funzionali
- Function point (FP)
- Analisi dei function point
- Conteggio di function point
- Tipo di funzione
- Mantenuto (\*)
- Manutenzione
- Molteplicità di siti (GSC)
- Manutenzione migliorativa
- Scopo del conteggio
- Tipo di elemento record (RET)
- Attributo tecnico
- Funzioni di tipo transazionale
- Numero di function point non pesati (UFP – Unadjusted Function Point)
- Utente
- Riconoscibile dall'utente
- Punto di vista dell'utente (User View)

(\*) Indica le voci dove il nome del termine è stato modificato (ad esempio 'Numero di function point pesati' (AFP – Adjusted Function Point count) è diventato 'Dimensione funzionale pesata').

## Processo decisionale

Il processo decisionale interno del CPC è governato da un insieme di caratteristiche (metaregole) del CPM, selezionate e votate dal management IFPUG e dal CPC. Tali principi informatori sono, in ordine di importanza:

1. Dovrebbe essere possibile modellare la correlazione tra la dimensione del software (derivata usando il CPM) e altri attributi (per esempio: impegno, difetti, costo, ecc.).
2. Il CPM contiene un insieme coerente di regole.
3. I risultati dell'analisi dei function point (FPA) sono coerenti tra conteggi effettuati da persone diverse che usano il CPM.
4. Il CPM fornisce regole su come dimensionare un'esigenza funzionale che viene definita e concordata tra l'utente e l'IT.
5. I risultati dell'analisi dei function point (FPA) usando il CPM possono rappresentare un fattore che contribuisce al processo di stima.
6. Il CPM è un metodo basato sulla tecnica originaria di Albrecht.
7. L'analisi dei function point (FPA) usando il CPM è semplice.
8. L'analisi dei function point (FPA) usando il CPM è rapida.

## Studio d'impatto

Il processo e le regole dell'analisi dei function point (FPA – Function Point Analysis) dell'IFPUG sono concise e facili da usare. Per rispettare tali caratteristiche e per rendere il Manuale delle Regole di Conteggio (CPM – Counting Practices Manual) ancor più pratico come manuale di riferimento, il Comitato per le Regole di Conteggio (CPC – Counting Practices Committee) ha ristrutturato il CPM 4.3 per allinearli agli standard di presentazione ISO. Inoltre, la versione 4.3 contiene modifiche minori e fornisce nuovi esempi, chiarimenti e migliora interpretazione delle regole esistenti che aumenteranno ulteriormente la coerenza tra i conteggiatori.

Per misurare l'efficacia di questa nuova versione è stato condotto uno studio di impatto che ha coinvolto 44 Certified Function Point Specialists (CFPS) che non avevano una connessione diretta con il Comitato per le Regole di Conteggio (CPC). A questi volontari è stato richiesto di effettuare un conteggio su un caso di studio usando sia la versione CPM 4.2 che la versione CPM 4.3. I loro risultati sono stati identici utilizzando entrambe le versioni. I partecipanti hanno inoltre applicato le regole del CPM 4.3 a progetti per cui avevano utilizzato la versione CPM 4.2.1. In totale sono stati più di 100 i conteggi che includevano sia conteggi di progetti di sviluppo, che su applicazioni e progetti di manutenzione evolutiva e conversione. E' stato pertanto ricavato che il fattore di conversione è pari a 1.0; ovvero non ci sono differenze tra le due versioni.

## Conversione dal CPM 4.2 al CPM 4.3

Poiché le prassi di conteggio esistenti variano da caso a caso, ciascuna organizzazione deve analizzare le proprie prassi al fine di determinare in quale misura possano essere impattate. Talune organizzazioni potrebbero individuare un fattore di conversione applicabile su tutto il proprio portafoglio di progetti o sistemi. Altre potrebbero invece scoprire che il fattore di conversione varia da tipo a tipo di sistema e che, in certi casi, i sistemi dovranno essere nuovamente misurati.

## Impatto su Utenti 4.2.1 che passano alla versione 4.3

Anche se non sarà richiesta una certificazione supplementare relativa al CPM 4.3 per i conteggiatori, i test di certificazione saranno aggiornati per uniformarsi alla versione 4.3.

## Raccomandazioni

Il CPC raccomanda le seguenti azioni agli utenti che passano dal CPM 4.2 al CPM 4.3:

- Aggiornare tutto il materiale didattico sviluppato internamente in conformità alla nuova versione.
- Verificare che tutte le persone addette ai conteggi all'interno della vostra organizzazione siano state appropriatamente addestrate ed edotte sulle differenze tra 4.2 e 4.3.
- Controllare che tutto il materiale di addestramento offerto dai fornitori sia certificato per la nuova versione.
- Informare sui cambiamenti tutte le persone interne alla vostra organizzazione coinvolte nelle misurazioni della dimensione funzionale e mettere a loro disposizione il nuovo manuale.
- Rivedere tutti gli strumenti di conteggio, sia automatici che manuali, forniti ai clienti, per la certificazione IFPUG alla versione 4.3, se applicabile, e le variazioni per uniformarsi alle regole di conteggio 4.3.
- Qualora si forniscano servizi basati sui Function Point, assicurarsi che la dicitura del contratto sia rivista per determinare quale versione del CPM sarà utilizzata; effettuare delle modifiche laddove necessario.
- Specificare, nella documentazione di ciascuna misurazione della dimensione funzionale effettuata, insieme ai risultati, quale versione del CPM è stata usata.
- Accertarsi di specificare quale versione del CPM IFPUG è stata usata per effettuare i conteggi quando si sottomettono dati di benchmarking, sia verso il proprio database di benchmarking, sia verso il Comitato sul Benchmarking dell'IFPUG, sia verso l'ISBSG.
- Aggiornare tutte le direttive interne e gli altri documenti locali relativi per riflettere i cambiamenti da 4.2 a 4.3.

---

### Appendice C: Dimensione Funzionale Pesata

---

**Introduzione** Questo capitolo illustra le Caratteristiche Generali del Sistema (GSC - General Systems Characteristics) e il Fattore di Aggiustamento del Valore (VAF - Value Adjustment Factor).

**Nota: l'applicazione delle GSC, il calcolo del VAF e il calcolo della dimensione funzionale pesata non sono incluse nell'IFPUG FSM e sono considerate opzionali nell'IFPUG CPM.**

Ad ogni modo, quando ci si riferisce alla dimensione funzionale misurata secondo la metodologia IFPUG, la dimensione riportata è quella misurata prima di ogni aggiustamento (dimensione non pesata). La dimensione riportata è in unità di FP (Function Points). Quando si utilizzano dimensioni riportate da altre fonti, per prima cosa bisogna determinare se sono state pesate (*adjusted*) o se sono non-pesate (*unadjusted*). Una mancanza in tal senso potrebbe comportare dei errori nel comparare o nell'usare i due insiemi di risultati. Bisogna sempre indicare se la dimensione funzionale che si sta riportando è 'pesata' (aFP – adjusted Function Point) o 'non pesata' (FP – Function Point).

**Contenuti** Questo capitolo include le seguenti sezioni:

Argomento	Pagina
Passi per calcolare la Dimensione Funzionale Pesata	C-3
Determinare il Fattore di Aggiustamento del Valore	C-4
Linee Guida per Determinare i Gradi di Influenza per le GSC	C-6
Tabella del Fattore di Aggiustamento del Valore (VAF)	C-31
<b>Function Point Pesati per un Progetto di Sviluppo (aDFP)</b>	<b>C-32</b>

<b>Argomento</b>	<b>Pagina</b>
Formula: Dimensione Funzionale Pesata per un Progetto di Sviluppo (aDFP)	C-32
Esempio: Dimensione Funzionale Pesata per un Progetto di Sviluppo (aDFP)	C-33
<b>Dimensione Funzionale Pesata per un Progetto di Manutenzione Evolutiva (aEFP)</b>	<b>C-37</b>
Formula: Dimensione funzionale pesata per un progetto di manutenzione evolutiva (aEFP)	C-38
Esempio: Dimensione funzionale pesata per un progetto di manutenzione evolutiva (aEFP)	C-39
<b>Dimensione Funzionale pesata per un'Applicazione (aAFP)</b>	<b>C-43</b>
Formula: Dimensione Funzionale pesata Iniziale (aAFP)	C-44
Formula: Dimensione Funzionale Pesata per un'Applicazione dopo un progetto di Manutenzione Evolutiva (aAFPA)	C-44
Esempio: Conteggio per un'Applicazione	C-45

## Passi per calcolare la dimensione funzionale pesata

La seguente lista include i passi per l'analisi dei function point introdotti nella Parte 1, estesi per fornire la dimensione funzionale pesata.

Passo	Azione
1	Determinare la dimensione funzionale usando le regole della Parte 1 e la guida all'implementazione della Parte 2.
2	Determinare il fattore di aggiustamento del valore (VAF) conformemente alle guide riportate in questa appendice
3	Calcolare la dimensione funzionale pesata conformemente alle formule riportate in questa appendice

La parte rimanente di questo capitolo presenta le Caratteristiche Generali del Sistema (GSC), il Fattore di Aggiustamento del Valore (VAF) che ne deriva seguito dalle formule per calcolare la Dimensione Funzionale Pesata. Sono inclusi esempi di calcolo per ciascuno dei tre tipi di conteggio dei function point:

- Progetto di sviluppo
- Progetto di manutenzione evolutiva
- Applicazione

## Determinare il Fattore di Aggiustamento del Valore

Il fattore di aggiustamento del valore (VAF – Value Adjustment Factor) si basa su 14 caratteristiche generali del sistema (GSC – General System Characteristics) che quantificano le funzionalità generali dell'applicazione che si sta misurando. Ad ogni caratteristica sono associate delle descrizioni che aiutano a determinarne il grado di influenza. I gradi di influenza per ogni caratteristica variano in una scala da zero a cinque, da nessuna influenza ad una influenza forte e generalizzata.

Le 14 caratteristiche generali del sistema sono riassunte nel fattore di aggiustamento del valore. Quando applicato, il fattore di aggiustamento del valore corregge il numero di function point non pesati di  $\pm 35\%$  per produrre la dimensione funzionale pesata.

### Procedure per calcolare il VAF

I seguenti passi schematizzano le procedure per calcolare il fattore di aggiustamento del valore.

Passo	Azione
1	Valutare le 14 caratteristiche generali del sistema in una scala da zero a cinque per determinarne il grado di influenza (DI – Degree of Influence).
2	Sommare i gradi di influenza di tutte le 14 caratteristiche generali del sistema per ottenere il grado di influenza totale (TDI – Total Degree of Influence).
3	Sostituire il TDI nella seguente formula per ottenere il fattore di aggiustamento del valore. $\text{VAF} = (\text{TDI} \times 0.01) + 0.65$ <p><u>Ad esempio</u>, nel caso di grado di influenza pari a 3 per ciascuna delle 14 GSC (<math>3 \times 14 = 42</math>), il fattore di aggiustamento del valore è calcolato come segue:</p> $\text{VAF} = (42 \times 0.01) + 0.65$ $\text{VAF} = 1.07$

Nell'Appendice è inclusa una tabella per facilitare il calcolo del VAF.



## Caratteristiche Generali del Sistema

Le *caratteristiche generali del sistema* sono un insieme di 14 proprietà che valutano la complessità globale dell'applicazione.

Le 14 caratteristiche generali del sistema sono:

1. Comunicazione dati
2. Elaborazione dati distribuita
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. Elaborazione complessa
10. Riusabilità
11. Facilità di installazione
12. Facilità di gestione operativa
13. Molteplicità di siti
14. Agevolazione delle modifiche

## Gradi di influenza

In base ai requisiti utente definiti, ciascuna caratteristica generale del sistema (GSC – General System Characteristic) deve essere valutata in termini del suo grado di influenza (DI – Degree of Influence) in una scala da zero a cinque.

Assegna	Influenza sul Sistema
0	Non presente, o di nessuna influenza.
1	Influenza secondaria
2	Influenza moderata.
3	Influenza media.
4	Influenza significativa
5	Influenza forte e generalizzata

## Linee guida per determinare i gradi di influenza per le GSC

Le descrizioni di ognuna delle seguenti caratteristiche generali del sistema includono delle linee guida per la determinazione del grado di influenza.

Ogni linea guida contiene una definizione della caratteristica generale del sistema, le regole per assegnare il grado di influenza e, laddove le regole necessitano di ulteriori chiarimenti, suggerimenti forniti per aiutare ad applicare le regole in modo coerente su tutte le piattaforme.

I suggerimenti non pretendono di coprire tutte le situazioni possibili, ma forniscono un ausilio aggiuntivo per determinare la giusta assegnazione del grado di influenza.

## 1. Comunicazione Dati

**Definizione** La caratteristica *Comunicazione Dati* descrive il grado con cui l'applicazione comunica direttamente con l'elaboratore.

I *dati* e le *informazioni di controllo* usati nell'applicazione sono trasmessi o ricevuti attraverso servizi di comunicazione. Si assume che gli apparati connessi localmente alle unità di controllo usino un sistema di comunicazione. Un protocollo è un insieme di convenzioni che permettono di trasferire o scambiare informazioni fra due sistemi o dispositivi. Tutti i collegamenti di comunicazione dati richiedono qualche tipo di protocollo.

### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	L'applicazione è un'elaborazione puramente batch o un'applicazione standalone.
1	L'applicazione è batch, ma ha inserimento dati o stampa remoti.
2	L'applicazione è batch, ma ha inserimento dati e stampa remoti.
3	L'applicazione utilizza una raccolta dati interattiva o un front-end TP (teleprocessing) verso un processo batch o un sistema di interrogazioni.
4	L'applicazione è più di un front-end, ma supporta <b>un solo</b> tipo di protocollo di comunicazione TP.
5	L'applicazione è più di un front-end e supporta <b>più di un</b> tipo di protocollo di comunicazione.

**Suggerimenti** Esempi di protocollo comprendono FTP, accesso via modem, Token Ring, Ethernet, SNA, TCP/IP, IPX/SPX, HTTP, XML, WAP, NTP, ICQ, e NETBEUI. Questo elenco **non** dovrebbe essere considerato esaustivo.

### Suggerimenti sui Gradi di Influenza 1 e 2

- Gli apparati remoti potrebbero includere un terminale 3270 connesso a un mainframe che permette solo edit semplice (numerico o a caratteri), oppure stampanti connesse alla porta parallela (l'utente è in grado di specificare su quale stampante indirizzare l'output).
- L'inserimento di dati non implica lettura o scrittura diretta di un ILF. I dati sono inseriti in maniera interattiva, ma le transazioni sono memorizzate in un file temporaneo per l'aggiornamento batch di uno o più ILF in un momento successivo.

**Suggerimenti  
sul Grado di  
Influenza 3**

- Possono essere eseguite semplici regole di business e edit minimale (per esempio, edit alfanumerico, convalida di intervalli, dati obbligatori, ecc.). Qualora questi dati siano ulteriormente elaborati dall'applicazione, viene eseguito edit aggiuntivo.
- L'inserimento di dati non implica lettura o scrittura diretta di un ILF. I dati sono inseriti in maniera interattiva, ma le transazioni sono memorizzate in un file temporaneo per l'aggiornamento batch di uno o più ILF in un momento successivo.

**Suggerimenti  
sul Grado di  
Influenza 4**

- I dati dell'applicazione sono raccolti e possono aggiornare direttamente uno o più ILF o essere memorizzati per un'elaborazione successiva tramite un apparato di input, che svolge edit basato su regole di business.
- Viene utilizzato un solo protocollo di comunicazione. Tipicamente, quando questi dati vengono elaborati dall'applicazione, non è necessario ulteriore edit.
- L'inserimento di dati implica lettura o scrittura diretta di un ILF.
- Per esempio, inserimento di dati client-server o Internet, ma non entrambi.

**Suggerimenti  
sul Grado di  
Influenza 5**

- Come per il grado di influenza 4, ma la raccolta dei dati è svolta utilizzando molteplici protocolli di comunicazione.
- Per esempio, inserimento di dati client-server e Internet per la medesima transazione.

**Tipicamente**

- Le applicazioni batch presentano grado di influenza da 0 a 3.
- Le applicazioni interattive presentano grado di influenza pari a 4.
- Le applicazioni web presentano grado di influenza pari a 4 o 5.
- I sistemi real-time, di telecomunicazioni o di controllo di processo presentano grado di influenza pari a 4 o 5.

## 2. Elaborazione Dati Distribuita

**Definizione** La caratteristica *Elaborazione Dati Distribuita* descrive il grado con cui l'applicazione trasferisce dati tra i suoi componenti fisici.

La distribuzione di funzioni di tipo dati o di elaborazione è una caratteristica dell'applicazione, all'interno del confine della stessa.

### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	Non si ha trasferimento o elaborazione di dati tra i componenti del sistema.
1	I dati sono preparati per il trasferimento, poi sono trasferiti ed elaborati su un altro componente del sistema, per elaborazioni da parte dell'utente.
2	I dati sono preparati per il trasferimento, poi sono trasferiti ed elaborati su un altro componente del sistema, <b>non</b> per elaborazioni da parte dell'utente.
3	L'elaborazione distribuita e il trasferimento di dati sono interattivi e <b>unidirezionali</b> .
4	L'elaborazione distribuita e il trasferimento di dati sono interattivi e <b>bidirezionali</b> .
5	L'elaborazione distribuita e il trasferimento di dati sono interattivi e sono eseguiti dinamicamente sul componente più appropriato del sistema.

**Suggerimenti** Per definizione, l'elaborazione distribuita dei dati non è un'applicazione installata su un unico elaboratore centrale, che invia dati ad altre applicazioni. In un ambiente distribuito, l'applicazione è percepita come qualcosa che richiede molteplici componenti (hardware) sui quali risiedono particolari elaborazioni o dati. Un utente esperto tipicamente riconoscerebbe tale configurazione.

#### Suggerimenti sul Grado di Influenza 0

- I componenti di presentazione, di elaborazione e di I/O sono tutti nello stesso punto (applicazioni stand-alone).

#### Suggerimenti sul Grado di Influenza 1

- L'applicazione scarica dati su una macchina client dell'utente, in modo tale che l'utente possa utilizzare Excel o altri strumenti di reporting per predisporre dei grafici o svolgere altre analisi.
- Processi che trasferiscono dati dal mainframe a componenti esterni per l'elaborazione da parte dell'utente. Questi trasferimenti sono svolti tramite un protocollo semplice, come l'FTP.
- Trasferimento all'utente per elaborazione successiva.

- |  |  |
|--|--|
| <b>Suggerimenti<br/>sul Grado di<br/>Influenza 2</b> | <ul style="list-style-type: none"> <li>• Processi che trasferiscono dati dal mainframe a uno strato intermedio. Per esempio, elaborazioni tramite SAS-PC.</li> <li>• L'applicazione invia dati ad un client o server. I dati sono quindi elaborati o utilizzati per produrre report, ecc. Non si hanno dati o conferme re-inviati al client o server.</li> <li>• Trasferimenti verso un componente per elaborazioni.</li> </ul>  |
| <b>Suggerimenti<br/>sul Grado di<br/>Influenza 3</b> | <ul style="list-style-type: none"> <li>• Invio di dati tra client e server in <b>una sola</b> direzione. I dati sono quindi elaborati o utilizzati per produrre report, ecc., da parte dell'applicazione ricevente. Tipicamente questi dati comprendono transazioni di aggiornamento di un ILF sul client o server.</li> <li>• Per esempio, applicazioni client-server o applicazioni web.</li> </ul>  |
| <b>Suggerimenti<br/>sul Grado di<br/>Influenza 4</b> | <ul style="list-style-type: none"> <li>• Invio di dati tra client e server in <b>entrambe</b> le direzioni. I dati sono quindi elaborati o utilizzati per produrre report, ecc., da parte dell'applicazione ricevente. Tipicamente questi dati comprendono transazioni di aggiornamento di un ILF sul client o server.</li> <li>• Per esempio, applicazioni client-server o applicazioni web.</li> <li>• L'applicazione gira su un sistema operativo che gestisce automaticamente l'allocazione tra componenti, <b>tuttavia</b> l'utilizzo del sistema operativo non ha influenzato la progettazione e l'implementazione dell'applicazione.</li> </ul> |
| <b>Suggerimenti<br/>sul Grado di<br/>Influenza 5</b> | <ul style="list-style-type: none"> <li>• Lo sviluppatore deve considerare un software applicativo particolare, che prende in esame processori multipli e fa girare l'applicazione su uno specifico tipo di processore. Questo è trasparente per l'utente.</li> <li>• L'applicazione gira su un sistema operativo che gestisce automaticamente l'allocazione dinamica tra componenti, <b>e</b> l'utilizzo del sistema operativo ha specificatamente influenzato la progettazione e l'implementazione dell'applicazione.</li> </ul>  |

- Tipicamente**
- La maggior parte delle applicazioni, incluse le applicazioni legacy, presenta grado di influenza pari a 0.
  - Applicazioni distribuite elementari, comprese applicazioni batch in cui i dati non sono trasferiti in maniera interattiva, presentano grado di influenza da 1 a 2.
  - Le applicazioni client-server e le applicazioni web presentano grado di influenza da 3 a 4.
  - Un grado di influenza pari a 5 è poco frequente.
  - Per assegnare un grado di influenza pari a 5, devono esserci server o elaboratori multipli, ciascuno dei quali verrebbe selezionato dinamicamente in base alla sua disponibilità real-time.

### 3. Prestazioni

**Definizione** La caratteristica *Prestazioni* descrive il grado con cui considerazioni sulle stesse, relativamente ai tempi di risposta e alla quantità di lavoro svolta nell'unità di tempo (*throughput*), hanno influenzato lo sviluppo dell'applicazione.

Gli obiettivi prestazionali dell'applicazione, dichiarati o approvati (o sottintesi) dall'utente, in termini o di tempo di risposta o di quantità di lavoro svolta nell'unità di tempo, influenzano (o influenzeranno) la progettazione, lo sviluppo, l'installazione e la gestione dell'applicazione.

#### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	Nessun particolare requisito prestazionale è stato espresso dall'utente.
1	Sono stati dichiarati e esaminati requisiti prestazionali e di progettazione, ma tali da non comportare azioni particolari.
2	Il tempo di risposta o il <i>throughput</i> sono critici durante le ore di <b>picco</b> . Non è richiesta alcuna particolarità di progettazione per l'utilizzo della CPU. L'elaborazione può essere completata per il successivo periodo lavorativo.
3	Il tempo di risposta o il <i>throughput</i> sono critici durante <b>tutte</b> le ore <b>lavorative</b> . Non è richiesta alcuna particolarità di progettazione per l'utilizzo della CPU. I requisiti di completamento dell'elaborazione verso i sistemi interfacciati sono vincolanti.
4	In aggiunta, i requisiti prestazionali dichiarati dall'utente sono così stringenti da richiedere un passo di analisi delle prestazioni durante la fase di progettazione.
5	In aggiunta, per soddisfare i requisiti prestazionali dichiarati dall'utente sono stati usati degli strumenti per l'analisi delle prestazioni durante le fasi di progettazione, sviluppo e/o realizzazione.

**Suggerimenti**

- Le GSC 3, 4 e 5 sono alquanto correlate. Per questa GSC, si pensi in termini di: *“Quanto velocemente possiamo far andare l'applicazione e in che misura ciò impatta/ha impattato la progettazione, lo sviluppo e/o l'implementazione?”*.
- Gli utenti possono richiedere un accesso in tempo reale ai propri dati, stabilendo o sottintendendo degli standard per i tempi di risposta e per la quantità di lavoro sostenibile nell'unità di tempo.
- Il tempo di risposta è tipicamente correlato all'elaborazione interattiva; la quantità di lavoro svolta nell'unità di tempo è correlata all'elaborazione batch.

### **Tipicamente**

- Le applicazioni batch presentano grado di influenza da 0 a 4.
- Le applicazioni interattive (incluse applicazioni client-server e applicazioni web interattive) presentano grado di influenza da 0 a 4.
- Le applicazioni web presentano grado di influenza pari a 4 o 5.
- La maggior parte dei sistemi gestionali interattivi (MIS – Management Information System) presenta grado di influenza pari a 2.
- I sistemi real-time, di telecomunicazioni o di controllo di processo presentano grado di influenza da 0 a 5.
- Un grado di influenza pari a 5 richiede l'utilizzo di strumenti per l'analisi delle prestazioni.



## 4. Utilizzo Intensivo della Configurazione

**Definizione** La caratteristica *Utilizzo Intensivo della Configurazione* descrive il grado con cui le limitazioni sulle risorse del computer hanno influenzato lo sviluppo dell'applicazione.

Una configurazione di esercizio utilizzata intensivamente può richiedere particolari considerazioni in fase di progettazione dell'applicazione. Per esempio, l'utente vuole far eseguire l'applicazione su macchine esistenti o prestabilite, che saranno utilizzate pesantemente.

### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	Non ci sono limitazioni operative implicite o esplicite.
1	Esistono limitazioni operative, ma sono meno restrittive di quelle di un'applicazione tipica. Non è necessario alcun impegno particolare per soddisfare tali restrizioni.
2	Esistono limitazioni operative, ma caratteristiche di un'applicazione tipica. Si richiede un impegno particolare, tramite controllori o programmi di controllo, per soddisfare tali restrizioni.
3	Limitazioni operative esplicite impongono per <b>una</b> porzione dell'applicazione vincoli particolari sull'unità centrale o un elaboratore dedicato.
4	Limitazioni operative esplicite impongono per l' <b>intera</b> applicazione vincoli particolari sull'unità centrale o un elaboratore dedicato.
5	In aggiunta, ci sono vincoli particolari per l'applicazione nei componenti distribuiti del sistema.

**Suggerimenti**

- Le GSC 3, 4 e 5 sono alquanto correlate.
- Per questa caratteristica, si pensi in termini di: *“In che misura l'infrastruttura influenza la progettazione?”*

**Esempi** Esempi di limitazioni operative possono comprendere quanto segue (elenco non esaustivo):

- L'applicazione deve girare su un elaboratore che risulta sottodimensionato e che non può gestire adeguatamente le funzionalità, aggiunte o modificate, e, in qualche modo, gli sviluppatori devono risolvere il problema realizzando l'applicazione in maniera differente
- Più applicazioni che accedono agli stessi dati possono dar luogo a vincoli operativi.
- Un'applicazione che si contende le medesime risorse e tecnologie con i potenziali 'punti morti' deve essere messa a punto e sottoposta a vincoli per evitare un degrado delle prestazioni.

- Tipicamente**
- La maggior parte delle applicazioni presenta grado di influenza pari a 2.
  - Applicazioni client-server, web, real-time, di telecomunicazioni o sistemi di controllo di processo presentano grado di influenza da 3 a 5, ma in tal caso sarebbe necessario o un elaboratore dedicato o elaboratori multipli che elaborino le medesime transazioni e ricerchino le modalità più rapide di elaborazione.

## 5. Frequenza delle Transazioni

**Definizione** La caratteristica *Frequenza delle Transazioni* descrive il grado con cui la frequenza delle transazioni di business ha influenzato lo sviluppo dell'applicazione.

La frequenza delle transazioni è importante e influenza la progettazione, lo sviluppo, l'installazione e la gestione dell'applicazione. Gli utenti possono richiedere quello che considerano un tempo di risposta normale anche durante i momenti di massimo volume di picco delle transazioni.

### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	Non è stato previsto un periodo di picco delle transazioni.
1	Basse frequenze di transazioni hanno un impatto minimale sulle fasi di progettazione, sviluppo e installazione.
2	Frequenze medie di transazioni hanno un certo impatto sulle fasi di progettazione, sviluppo e installazione.
3	Alte frequenze di transazioni condizionano le fasi di progettazione, sviluppo e installazione.
4	L'alta frequenza di transazioni dichiarata dall'utente nei requisiti dell'applicazione, o gli accordi sui livelli di servizio (SLA – Service Level Agreement), sono tali da richiedere dei passi di analisi delle prestazioni nelle fasi di progettazione, sviluppo e/o installazione.
5	L'alta frequenza di transazioni dichiarata dall'utente nei requisiti dell'applicazione, o gli accordi sui livelli di servizio (SLA – Service Level Agreement), sono tali da richiedere dei passi di analisi delle prestazioni e, in più, è richiesto l'uso di strumenti per l'analisi delle prestazioni durante le fasi di progettazione, sviluppo e/o installazione.

**Suggerimenti**

- Le GSC 3, 4 e 5 sono alquanto correlate. Per questa caratteristica, si pensi in termini di: *“Quante transazioni possono essere elaborate dall'applicazione in un dato intervallo di tempo?”*
- Spesso il grado di influenza di questa caratteristica è lo stesso della GSC 3, poiché la frequenza delle transazioni spesso impatta i requisiti prestazionali.

**Tipicamente**

- Le applicazioni batch presentano grado di influenza da 0 a 3.
- Le applicazioni interattive (incluse le applicazioni client-server o web) presentano grado di influenza da 0 a 4.
- I sistemi real-time, di telecomunicazioni o di controllo di processo presentano grado di influenza da 0 a 5.
- Un grado di influenza pari a 5 richiede l'utilizzo di strumenti per l'analisi delle prestazioni.

## 6. Inserimento Dati Interattivo

**Definizione** La caratteristica di *Inserimento Dati Interattivo* descrive il grado con cui i dati sono inseriti o reperiti attraverso transazioni.

L'applicazione fornisce un'interfaccia utente interattiva per l'inserimento dei dati, le funzioni di controllo, la reportistica e le interrogazioni.

### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	Tutte le transazioni sono elaborate in modalità batch.
1	Le transazioni interattive sono comprese tra l'1% e il 7%.
2	Le transazioni interattive sono comprese tra l'8% e il 15%.
3	Le transazioni interattive sono comprese tra il 16% e il 23%.
4	Le transazioni interattive sono comprese tra il 24% e il 30%.
5	Le transazioni interattive sono superiori al 30%.

**Suggerimenti**

- Questa caratteristica fa riferimento alle percentuali dei tipi di transazione, *non* ai volumi delle elaborazioni.
- Per esempio, se un'applicazione possiede 45 tra EI, EO ed EQ, quale percentuale degli EI, EO ed EQ è svolta tramite transazioni interattive?

**Tipicamente**

- Le applicazioni batch presentano grado di influenza da 0 a 1.
- Le applicazioni interattive, i sistemi real-time, di telecomunicazioni o di controllo di processo presentano grado di influenza pari a 5.
- La maggior parte applicazioni interattive attuali (incluse applicazioni client-server e applicazioni web interattive) presentano grado di influenza pari a 5.
- I sistemi batch con aspetti interattivi possono avere molte transazioni batch, ma per assegnare un grado di influenza inferiore a 5 la percentuale delle transazioni batch deve essere almeno il 71%.

## 7. Efficienza per l'Utente Finale

- Definizione** La caratteristica *Efficienza per l'Utente Finale* descrive il grado di considerazione per i fattori umani e la facilità d'uso per l'utente dell'applicazione misurata.
- Le funzioni interattive fornite enfatizzano una progettazione rivolta all'efficienza dell'utente (fattore umano/caratteristiche di usabilità – *user friendly*). La progettazione comprende:
- aiuti navigazionali (per esempio, tasti funzionali, salti, menu generati dinamicamente, collegamenti ipertestuali),
  - menu,
  - help e documentazione in linea,
  - spostamento automatico del cursore,
  - scorrimento,
  - stampa remota (tramite trasmissioni interattive),
  - tasti funzionali pre-definiti (per esempio, svuotamento schermata, richiesta di help, duplicazione schermata),
  - richiesta di attivazione di job batch attraverso transazioni interattive,
  - caselle di lista a discesa (listbox),
  - uso intensivo di reverse del video, evidenziazione, sottolineatura a colori e altri indicatori,
  - documentazione mediante hard copy delle transazioni interattive (per esempio, stampa schermata – print screen),
  - mouse,
  - finestre a scomparsa (pop-up),
  - modelli e/o valori predefiniti,
  - supporto bilingue (sono supportate due lingue: conta come quattro voci),
  - supporto multilingue (sono supportate più di due lingue: conta come sei voci).

## Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	Nessuna delle voci precedenti.
1	Da una a tre delle voci precedenti.
2	Da quattro a cinque delle voci precedenti.
3	Sei o più delle voci precedenti, ma non ci sono specifici requisiti utente legati all'efficienza.
4	Sei o più delle voci precedenti ed i requisiti dichiarati per l'efficienza dell'utente sono abbastanza forti da richiedere di includere dei <b>compiti di progettazione</b> per i fattori umani.
5	Sei o più delle voci precedenti ed i requisiti dichiarati per l'efficienza dell'utente sono abbastanza forti da richiedere l' <b>utilizzo di strumenti e processi particolari</b> per dimostrare che gli obiettivi sono stati raggiunti.

## Suggerimenti

- Assegna un grado di influenza convenzionalmente pari a 4 ogni volta che l'applicazione è rilasciata in ambiente GUI (Graphical User Interface), a meno di assegnazione del grado di influenza pari a 5.
- Solitamente, solo gli ambienti software che predispongono applicazioni per il mercato di massa o per utenti non specializzati hanno grado di influenza pari a 5, e solo se il loro processo di sviluppo prevede esperti di ergonomia e/o studi di usabilità.

## Tipicamente

- Le applicazioni puramente batch presentano grado di influenza pari a 0.
- Le interfacce utente a caratteri presentano grado di influenza pari a 1 o eventualmente 2.
- Le interfacce GUI utilizzate per transazioni di volume ridotto presentano grado di influenza pari a 3.
- Le interfacce GUI utilizzate per transazioni di volume elevato e la maggior parte delle interfacce utente web su **intranet** presentano grado di influenza pari a 4 (sono richiesti compiti di progettazione per i fattori umani).
- Le interfacce utente web su **internet** presentano grado di influenza pari a 5 (sono richiesti strumenti e processi particolari per dimostrare che gli obiettivi sono stati raggiunti).

## 8. Aggiornamento Interattivo

**Definizione** La caratteristica *Aggiornamento Interattivo* descrive il grado con cui i file logici interni sono aggiornati interattivamente.

L'applicazione fornisce l'aggiornamento interattivo dei file logici interni.

### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	Nessuno.
1	È previsto l'aggiornamento interattivo da uno a tre dei file di controllo. Il volume di aggiornamenti è basso ed il ripristino è facile.
2	È previsto l'aggiornamento interattivo di quattro o più file di controllo. Il volume di aggiornamenti è basso ed il ripristino è facile.
3	È previsto l'aggiornamento della maggior parte dei file logici interni.
4	In aggiunta, la protezione contro la perdita di dati è essenziale ed è stata progettata e realizzata in modo specifico nel sistema.
5	In aggiunta, elevati volumi portano a tenere conto dei costi nel processo di ripristino. Sono previste procedure di ripristino altamente automatizzate che minimizzano l'intervento umano.

**Suggerimenti**

- L'aggiornamento interattivo solitamente richiede file o database indicizzati con chiavi.
- Il ripristino automatico fornito dal sistema operativo viene preso in considerazione se esso ha impatto sull'applicazione.

**Tipicamente**

- Le applicazioni puramente batch presentano grado di influenza pari a 0.
- Gli aggiornamenti interattivi di file che modificano il modo in cui un'applicazione elabora o convalida i dati presentano grado di influenza pari a 1 o 2.
- Gli aggiornamenti interattivi di dati utente permanenti presentano grado di influenza pari a 3.
- Le applicazioni gestionali (MIS – Management Information System) presentano grado di influenza pari a 3 o inferiore.
- La maggior parte delle applicazioni di tipo GUI presenta grado di influenza pari a 3 o superiore.
- Applicazioni che utilizzano un ripristino da programma, come il rollback e il commit nell'SQL, presentano grado di influenza pari a 4. Il backup operativo/periodico non è considerato una protezione dalla perdita di dati.
- Applicazioni a cui si richiede di ripristinare i dati, riavviarsi o svolgere altre funzioni autonome in caso di errore di sistema presentano grado di influenza pari a 5. Il ripristino può richiedere l'intervento umano per dare l'avvio o effettuare altre funzioni minimali per inizializzare questo processo.

## 9. Elaborazione Complessa

**Definizione** La caratteristica *Elaborazione Complessa* descrive il grado con cui la logica di elaborazione ha influenzato lo sviluppo dell'applicazione. Sono presenti le seguenti componenti:

- controlli delicati e/o elaborazione della sicurezza specifica per l'applicazione,
- notevole elaborazione logica,
- notevole elaborazione matematica,
- molta elaborazione delle eccezioni, che si risolve in transazioni non completate che devono essere nuovamente elaborate,
- elaborazione complessa che gestisce possibilità di input/output multiplo.

### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	Nessuna delle voci precedenti.
1	Una qualsiasi delle voci precedenti.
2	Due qualsiasi delle voci precedenti.
3	Tre qualsiasi delle voci precedenti.
4	Quattro qualsiasi delle voci precedenti.
5	Tutte le cinque voci precedenti.

- Suggerimenti**
- I controlli delicati e/o l'elaborazione della sicurezza (per esempio, singoli utenti avrebbero privilegi di accesso differenti a schermate per visualizzare e/o modificare certi dati) possono includere particolari elaborazioni di *audit* (dati di *audit* verrebbero registrati ogni volta che i dati applicativi fossero oggetto di lettura, scrittura e/o reportistica).
  - L'elaborazione della sicurezza specifica per l'applicazione può includere una gestione della sicurezza sviluppata internamente o l'utilizzo di pacchetti per la sicurezza acquistati esternamente.
  - Notevole elaborazione logica denota la presenza di logica booleana (uso di 'AND', 'OR') di complessità superiore alla media o un minimo di 4 condizioni (IF, CASE) annidate. Nella maggior parte delle applicazioni gestionali (MIS – Management Information System) non si trova notevole elaborazione logica.
  - Notevole elaborazione matematica denota la presenza di aritmetica oltre la combinazione delle 4 operazioni elementari (addizione, sottrazione, moltiplicazione e divisione). Questa non è presente nella maggior parte delle applicazioni gestionali (MIS – Management Information System). Tuttavia, potrebbe trovarsi in un'applicazione di tipo ingegneristico.



- Suggerimenti**
- L’elaborazione delle eccezioni comprende transazioni ATM (Automated Teller Machine, per es. sportelli bancomat) non completate a causa di un’interruzione del sistema TP, valori di dati mancanti, convalide non andate a buon fine o controlli di ridondanza del ciclo di lavoro, che possono essere utilizzati per ricreare porzioni di dati andate perse.
  - Le possibilità di input/output multiplo comprendono elementi multimediali, indipendenza dal dispositivo, comandi e/o output vocali, lettura di testo stampato con riconoscimento dei caratteri, lettura di codici a barre, scan della retina e analisi del tasso alcolico dall’alito.

**Tipicamente** Il grado di influenza per questa caratteristica è indipendente dalla piattaforma.

## 10. Riusabilità

**Definizione** La caratteristica *Riusabilità* descrive il grado con cui l'applicazione ed il relativo codice sono stati specificatamente progettati, sviluppati e forniti degli strumenti di supporto per essere utilizzabili in **altre** applicazioni.

### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	Non esiste codice riusabile.
1	Il codice riusabile è utilizzato all'interno dell'applicazione.
2	Meno del 10% del codice applicativo sviluppato è finalizzato all'uso in più di un'applicazione.
3	Il 10% o più del codice applicativo sviluppato è finalizzato all'uso in più di un'applicazione.
4	L'applicazione è stata specificatamente assemblata e/o documentata per un facile riuso ed è personalizzata a livello di codice sorgente.
5	L'applicazione è stata specificatamente assemblata e/o documentata per un facile riuso ed è personalizzata per l'uso tramite la gestione di parametri utente.

- Suggerimenti**
- |  |   |
|--|---|
| <b>Suggerimenti sul Grado di Influenza 1</b> | <ul style="list-style-type: none"> <li>• Un grado di influenza pari a 1 è assegnato per il riuso di codice a prescindere da dove esso sia stato sviluppato.</li> <li>• Il codice sviluppato specificatamente per il riuso all'interno dell'applicazione e usato più di una volta all'interno dell'applicazione vale tanto quanto il codice reperito da una libreria centralizzata e disponibile per un uso generalizzato.</li> </ul>  |
| <b>Suggerimenti sul Grado di Influenza 2</b> | <ul style="list-style-type: none"> <li>• Per assegnare un grado di influenza pari a 2 o superiore, il codice deve essere: sviluppato per l'uso in più di un'applicazione, memorizzato e gestito in una libreria centralizzata e disponibile per un uso generalizzato. Il codice proveniente da un'applicazione, che sia copiato e incollato in un'altra applicazione, non è considerato riuso.</li> <li>• Il codice riusabile sarebbe supportato da documentazione che ne permetta e ne faciliti il riuso.</li> </ul> |
| <b>Suggerimenti sul Grado di Influenza 5</b> | <ul style="list-style-type: none"> <li>• Esempi di applicazioni personalizzate attraverso l'uso di parametri includono PeopleSoft e SAP e presenterebbero generalmente un grado di influenza pari a 5.</li> <li>• Il codice riusato può essere <b>leggermente</b> modificato nell'applicazione di destinazione.</li> <li>• Esempi di riuso includono oggetti o altro codice statico, mantenuti in una libreria di oggetti/codice.</li> </ul>  |

**Tipicamente** Il grado di influenza per questa GSC è indipendente dalla piattaforma.

## 11. Facilità di Installazione

**Definizione** La *Facilità di Installazione* descrive il grado con cui la conversione da precedenti ambienti ha influenzato lo sviluppo dell'applicazione.

La facilità di conversione ed installazione sono caratteristiche dell'applicazione. Sono stati forniti, e verificati durante la fase di test di sistema, un piano di conversione e installazione e/o strumenti di conversione.

### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	L'utente non ha espresso considerazioni particolari e l'installazione <b>non</b> richiede particolari inizializzazioni.
1	L'utente non ha espresso considerazioni particolari, <b>ma</b> l'installazione richiede particolari inizializzazioni.
2	L'utente ha espresso requisiti per la conversione e l'installazione e sono fornite e verificate guide per la conversione e l'installazione. L'impatto della conversione sul progetto <b>non</b> è considerato importante.
3	L'utente ha espresso requisiti per la conversione e l'installazione e sono fornite e verificate guide per la conversione e l'installazione. L'impatto della conversione sul progetto <b>è</b> considerato importante.
4	In aggiunta al grado di influenza 2 precedente, sono forniti e verificati strumenti automatici per la conversione e l'installazione.
5	In aggiunta al grado di influenza 3 precedente, sono forniti e verificati strumenti automatici per la conversione e l'installazione.

**Suggerimenti**

- Conversione e installazione comprendono la conversione di dati preesistenti in nuovi file di dati, il caricamento di file con dati correnti o lo sviluppo di software per installazioni particolari, come il *porting*.
- Il software, acquistato o sviluppato, deve essere effettivamente utilizzato perché siano riconosciute l'installazione e la conversione.

#### Suggerimenti sul Grado di Influenza 1

- La maggior parte delle applicazioni gestionali (*business application*) richiede un setup particolare per essere installata e presenta un grado di influenza pari a 1.

#### Suggerimenti sul Grado di Influenza 2

- Se l'applicazione possiede requisiti di conversione e installazione, sono fornite guide per l'installazione e la fornitura di tali funzioni e guide non fa parte del percorso critico del progetto, assegna un grado di influenza pari a 2.

#### Suggerimenti sul Grado di Influenza 3

- Se l'applicazione possiede requisiti di conversione e installazione, sono fornite guide per l'installazione e la fornitura di tali funzioni e guide fa parte del percorso critico del progetto, assegna un grado di influenza pari a 3.

**Suggerimenti  
sul Gradi di  
Influenza 4 e 5**

- Se l'applicazione possiede requisiti di conversione e installazione e può essere installata senza alcun intervento esterno, assegna un grado di influenza pari a 4 o 5, in base agli altri requisiti per l'assegnazione dei gradi di influenza 2 o 3.

**Tipicamente** Il grado di influenza per questa caratteristica è indipendente dalla piattaforma.

## 12. Facilità di Gestione Operativa

**Definizione** La caratteristica *Facilità di Gestione Operativa* descrive il grado con cui l'applicazione si fa carico di aspetti operativi, come processi di esecuzione, backup e ripristino.

La facilità di gestione operativa è una caratteristica dell'applicazione. L'applicazione minimizza la necessità di attività manuali, come il montaggio dei nastri, la gestione della carta ed interventi manuali da eseguire sul posto.

### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	L'utente non ha specificato particolari considerazioni nella gestione operativa, fatta eccezione per le normali procedure di backup.
1 - 4	Una, alcune, o tutte le seguenti voci sono valide per l'applicazione. Selezionare tutte quelle che si applicano. Ciascuna voce vale un punto, a meno che non sia diversamente specificato. <ul style="list-style-type: none"> <li>Sono forniti processi efficaci di esecuzione, backup e ripristino, ma è richiesto l'intervento umano.</li> <li>Sono fornite processi efficaci di esecuzione, backup e ripristino, ma <b>non</b> è richiesto alcun intervento umano (conta come due punti).</li> <li>L'applicazione minimizza la necessità di montaggio di nastri e/o di accesso remoto ai dati che richieda un intervento umano.</li> <li>L'applicazione minimizza la necessità di gestione della carta.</li> </ul>
5	L'applicazione è progettata per operazioni non presidiate. Con "operazioni non presidiate" si intende che <i>nessun intervento umano</i> è richiesto per gestire il sistema, ad eccezione dell'esecuzione o della chiusura dell'applicazione. Il recupero automatico degli errori è una caratteristica dell'applicazione.

### Suggerimenti

#### Suggerimento sui Gradi di Influenza 1-4 (a)

- L'applicazione è in grado di svolgere autonomamente l'esecuzione, il backup e il ripristino; tuttavia, è richiesto l'intervento umano per avviare tali funzioni.

#### Suggerimento sui Gradi di Influenza 1-4 (b)

- L'applicazione è in grado di svolgere autonomamente l'esecuzione, il backup e il ripristino; non è richiesto l'intervento umano per avviare tali funzioni.

#### Suggerimento sui Gradi di Influenza 1-4 (c)

- L'applicazione minimizza la necessità di accedere a dati che non siano immediatamente disponibili.
- Può includere l'importazione dei dati da un elaboratore distribuito all'elaboratore locale prima dello svolgimento delle elaborazioni, per eliminare le attese/ritardi di accesso.

**Suggerimento  
sui Gradi di  
Influenza 1-4 (d)**

- L'applicazione è progettata per fornire i dati all'utente in un formato compatto o tramite un supporto differente dal cartaceo.
- Potrebbe includere l'eliminazione della stampa di informazioni di dettaglio o l'accesso a report e interrogazioni in linea, microfiche, CD o altri supporti di questo tipo.

**Suggerimento  
sul Grado di  
Influenza 5**

- Un grado di influenza pari a 5 è assegnato a un'applicazione che funziona autonomamente e recupera automaticamente gli errori – operazione non presidiata.
- Le operazioni non presidiate possono includere la gestione di satelliti senza equipaggio e di reattori nucleari e i processi di controllo del traffico aereo.

**Tipicamente** Il grado di influenza per questa caratteristica è indipendente dalla piattaforma.

## 13. Molteplicità di Siti

**Definizione** La caratteristica *Molteplicità di Siti* descrive il grado con cui l'applicazione è stata sviluppata per funzionare in ambienti hardware e software differenti.

### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	La progettazione ha tenuto conto delle necessità di un <b>solo sito</b> di installazione.
1	La progettazione ha tenuto conto delle necessità di più di un sito di installazione e l'applicazione è progettata per operare solo in ambienti hardware e software <b>identici</b> .
2	La progettazione ha tenuto conto delle necessità di più di un sito di installazione e l'applicazione è progettata per operare solo in ambienti hardware e software <b>simili</b> .
3	La progettazione ha tenuto conto delle necessità di più di un sito di installazione e l'applicazione è progettata per operare solo in ambienti hardware e software <b>differenti</b> .
4	Sono forniti e verificati documenti e piani di supporto per gestire l'applicazione su molteplici siti d'installazione e l'applicazione è descritta dal grado di influenza 2.
5	Sono forniti e verificati documenti e piani di supporto per gestire l'applicazione su molteplici siti d'installazione e l'applicazione è descritta dal grado di influenza 3.

**Suggerimenti** La locuzione “molteplicità di siti” denota un aspetto logico e non necessariamente fisico. Si possono avere siti multipli nella medesima collocazione fisica. Il fattore discriminante si basa sulle necessità delle varie installazioni.

- Suggerimenti sul Grado di Influenza 0**
- La maggior parte delle applicazioni mainframe presenterebbe probabilmente un grado di influenza pari a 0.
  - Tuttavia, se un'applicazione è installata su molteplici computer mainframe aventi configurazioni significativamente differenti o sistemi operativi differenti, si assegnerebbe un grado di influenza maggiore di 0.
- Suggerimenti sul Grado di Influenza 1**
- Per esempio, Windows NT su hardware con esattamente le medesime configurazioni.
- Suggerimenti sul Grado di Influenza 2**
- Per esempio, Windows 95, 98 e NT su hardware con configurazioni simili.
  - Le varianti possono includere dimensioni di memoria differenti, capacità di disco fisso variabile, differenti velocità di processore e differenti tipi di stampante.

**Suggerimenti  
sul Grado di  
Influenza 3**

- Per esempio, Windows, OS X, UNIX, Linux e VOS3 su tipi di hardware differenti.
- Le differenze possono includere PC Intel, MAC, Tandem, Sun e AS400.

**Tipicamente**

Il grado di influenza per questa GSC dipende dalla quantità di piattaforme differenti.



## 14. Agevolazione delle Modifiche

**Definizione** La caratteristica Agevolazione delle Modifiche descrive il grado con cui l'applicazione è stata sviluppata per una facile modifica della logica elaborativa o della struttura dei dati.

Per l'applicazione possono essere valide le seguenti caratteristiche:

### A. Interrogazioni Flessibili

1. Sono fornite interrogazioni flessibili e ausili per la produzione di report che possono gestire richieste **semplici** (conta come 1 voce)
2. Sono fornite interrogazioni flessibili e ausili per la produzione di report che possono gestire richieste di **media** complessità (conta come 2 voci)
3. Sono fornite interrogazioni flessibili e ausili per la produzione di report che possono gestire richieste **complesse** (conta come 3 voci)

### B. Dati di Controllo del Business

1. I dati di controllo del business sono registrati in tabelle mantenute dall'utente attraverso processi in linea interattivi, ma le modifiche hanno effetto solamente dal **successivo** periodo lavorativo (conta come 1 voce)
2. I dati di controllo del business sono registrati in tabelle mantenute dall'utente attraverso processi in linea interattivi e le modifiche hanno effetto **immediatamente** (conta come 2 voci)

### Grado di Influenza

Assegna	Descrizioni per Determinare il Grado di Influenza
0	Nessuna delle voci precedenti.
1	Un totale di una delle voci precedenti.
2	Un totale di due delle voci precedenti.
3	Un totale di tre delle voci precedenti.
4	Un totale di quattro delle voci precedenti.
5	Un totale di cinque delle voci precedenti.

### Suggerimenti Interrogazioni Flessibili e Produzione di Report:

- La locuzione “interrogazioni flessibili e ausili per la produzione di report” significa più di una lista di opzioni in query o report ‘già pronti’.
- È la possibilità per l'utente di controllare i dati, le fonti dei dati, la sequenza e il formato nelle proprie richieste di query o di report.

- Suggerimenti**
- Rappresenta la possibilità di modificare il layout a video, l'ordinamento orizzontale e verticale, il formato di visualizzazione dei dati, i criteri di selezione sia dei file, sia dei singoli dati.
  - Include un'effettiva programmazione delle interrogazioni da parte dell'utente ed è talvolta denotato come *query ad hoc* o *reporting ad hoc*.
  - L'utilizzo di filtri per controllare la quantità di dati visualizzati o stampati in un formato predefinito non è considerato un caso di interrogazioni flessibili e ausili per la produzione di report.
  - La possibilità di creazione di interrogazioni e/o di report è spesso fornita con linguaggi come SQL o Focus, o per mezzo di strumenti più dinamici per il *reporting ad hoc* (per esempio, Crystal Reports).

- Suggerimento sul Grado di Influenza A1**
- Richieste semplici possono includere una logica AND/OR applicata a un solo file logico interno.

- Suggerimento sul Grado di Influenza A2**
- Richieste di media complessità possono includere una logica AND/OR applicata a **più di un** file logico interno.

- Suggerimento sul Grado di Influenza A3**
- Richieste complesse possono includere **combinazioni** di più logiche AND/OR applicate a uno o più file logici interni.

#### **Dati di Controllo del Business:**

- I Dati di Controllo del Business (Dati di Riferimento) vengono memorizzati per supportare le regole di business per la gestione dei Dati di Business; per es. in un'applicazione relativa agli stipendi sarebbero i dati relativi alle aliquote fiscali per ogni scaglione retributivo e la data a partire dalla quale ogni aliquota diventa effettiva.
- Vedi la Parte 3 – Dati di Decodifica per ulteriori informazioni.

**Tipicamente** Il grado di influenza per questa GSC è indipendente dalla piattaforma.

## Tabella del Fattore di Aggiustamento del Valore (VAF)

La seguente tabella è fornita per facilitare il calcolo del fattore di aggiustamento del valore.

Caratteristiche Generali del Sistema (GSCs - General System Characteristics)	Grado di Influenza (DI - Degree of Influence) 0 - 5
1. Comunicazione dati	_____
2. Elaborazione dati distribuita	_____
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. Elaborazione complessa	_____
10. Riutilizzabilità	_____
11. Facilità di installazione	_____
12. Facilità di gestione operativa	_____
13. Molteplicità di siti	_____
14. Agevolazione delle modifiche	_____
Grado di Influenza Totale (TDI - Total Degree of Influence)	_____
Fattore di Aggiustamento del Valore (VAF - Value Adjustment Factor)	_____
$VAF = (TDI \times 0.01) + 0.65$	

## Function Point Pesati per un Progetto di Sviluppo (aDFP)

### Funzionalità Applicative

Le funzionalità applicative consistono in funzioni usate dopo l'installazione al fine di soddisfare le attuali necessità applicative dell'utente.

### Funzionalità di Conversione

Le funzionalità di conversione consistono in funzioni fornite, solo all'installazione, al fine di convertire dati e/o soddisfare altri specifici requisiti utente di conversione, come particolari prospetti per la conversione.

Ad esempio, se un'applicazione software Risorse Umane (RU) era in uso ed una nuova applicazione RU viene installata, gli utenti richiedono che le informazioni sugli impiegati siano convertite e caricate nella nuova applicazione. Il requisito di conversione specificato dall'utente è quello di trasferire gli attuali dati sull'impiegato nel nuovo sistema RU.

### Fattore di Aggiustamento del Valore (VAF) dell'Applicazione

Il fattore di aggiustamento del valore è determinato usando le 14 caratteristiche generali del sistema (GSC) per valutare la complessità dell'applicazione.

### Formula: Dimensione Funzionale Pesata per un Progetto di Sviluppo (aDFP)

Usare la seguente formula per calcolare la dimensione funzionale pesata per un progetto di sviluppo.

$$\text{aDFP} = \text{DFP} \times \text{VAF}$$

Dove:

aDFP è la dimensione funzionale pesata per un progetto di sviluppo

DFP è la dimensione funzionale per un progetto di sviluppo (DFP = ADD + CFP; si veda la Parte 1)

VAF è il fattore di aggiustamento del valore (Value Adjustment Factor)

**Nota:** Dopo l'installazione del software, la dimensione funzionale dell'applicazione è calcolata usando i componenti della dimensione funzionale del progetto di sviluppo.

## Esempio: Dimensione Funzionale Pesata per un Progetto di Sviluppo (aDFP)

Questa sezione illustra un esempio di un progetto di sviluppo. Il progetto include sia funzionalità applicative sia funzionalità di conversione.

**Nota:** Il seguente conteggio è da intendersi a puro scopo di esempio è non specificatamente correlato ad altre sezioni del manuale. Tutte le funzionalità connesse potrebbero non essere incluse (es., potrebbero esserci funzioni mancanti).

### Funzionalità Applicative

La seguente tabella illustra le funzionalità applicative misurate per un progetto di sviluppo.

Funzioni di Tipo Dati	RET	DET	Complessità Funzionale
<b>File Logici Interni (ILF)</b>			
• Lavori	2	5	Bassa
• Lavori Sospesi	2	6	Bassa
• Definizioni Prospetti	1	4	Bassa
• Impiegati	1	6	Bassa
<b>File d'Interfaccia Esterni (EIF)</b>			
• Sedi	1	6	Bassa
• Tassi di Cambio	1	2	Bassa
• Help di Finestra	1	2	Bassa
• Help di Campo	1	5	Bassa

Funzioni di Tipo Transazionale	FTR	DET	Complessità Funzionale
<b>Input Esterni (EI)</b>			
Assegnazione Definizione Prospetto	1	5	Bassa
Inserimento Lavoro (a video)	1	7	Bassa
Inserimento Lavoro (batch)	2	6	Media
Correzione Lavori Sospesi	1	7	Bassa
Assegnazione Lavoro a Impiegato	3	7	Alta
EI con Output a Video – 1	2	11	Media
EI con Output a Video – 2	1	6	Bassa
<b>Output Esterni (EO)</b>			
Prospetto Lavori con Impiegati	4	5	Media
Prospetto Impiegati per Durata Assegnazione	3	7	Media
Notifica Verifica Prestazioni	3	4	Bassa
Prospetto Settimanale Impiegati	1	3	Bassa
Stampa Assegno	1	3	Bassa
File di Transazione Assegni	1	4	Bassa
<b>Interrogazioni Esterne (EQ)</b>			
Lista di Dati Reperiti	1	4	Bassa
Contratti Lavoro	1	2	Bassa
Help di Campo	1	6	Bassa
Prospetto Settimanale Associati	1	3	Bassa
File Giornaliero Assegni	1	2	Bassa

## Funzionalità di Conversione

La seguente tabella illustra le funzionalità di conversione per il progetto di sviluppo.

Funzioni di Tipo Transazionale	FTR	DET	Complessità Funzionale
<b>Input Esterni (EI)</b>			
Migrazione Impiegati	1	11	Bassa

## Contributo dell'Applicazione alla Dimensione Funzionale

La seguente tabella illustra il contributo alla dimensione funzionale non pesata.

Tipo Funzione	Complessità Funzionale			Totali Complessità	Totali Tipo Funzione
<b>ILF</b>	4	Bassa	X 7 =	28	28
	0	Media	X 10 =	0	
	0	Alta	X 15 =	0	
<b>EIF</b>	4	Bassa	X 5 =	20	20
	0	Media	X 7 =	0	
	0	Alta	X 10 =	0	
<b>EI</b>	4	Bassa	X 3 =	12	26
	2	Media	X 4 =	8	
	1	Alta	X 6 =	6	
<b>EO</b>	4	Bassa	X 4 =	16	26
	2	Media	X 5 =	10	
	0	Alta	X 7 =	0	
<b>EQ</b>	5	Bassa	X 3 =	15	15
	0	Media	X 4 =	0	
	0	Alta	X 6 =	0	
					ADD 115

## Contributo della Conversione alla Dimensione Funzionale

La seguente tabella illustra il contributo delle funzionalità di conversione alla dimensione funzionale.

Tipo Funzione	Complessità Funzionale			Totali Complessità	Totali Tipo Funzione
EI	1	Bassa	X 3 =	3	CFP 3
	0	Media	X 4 =	0	
	0	Alta	X 6 =	0	

## Calcolo Finale

Di seguito è illustrata la dimensione funzionale pesata del progetto di sviluppo ottenuta usando il conteggio della complessità e del contributo per questo esempio. Si suppone un fattore di aggiustamento del valore (VAF) per questo esempio pari a 1.05.

$$aDFP = (ADD + CFP) \times VAF$$

$$aDFP = (115 + 3) \times 1.05$$

$$aDFP = 123.9 \text{ (arrotondato a 124)}$$



## Dimensione Funzionale Pesata per un Progetto di Manutenzione Evolutiva (aEFP)

La dimensione funzionale pesata per un progetto di manutenzione evolutiva si basa su tre componenti:

- funzionalità applicative incluse nei requisiti utente per il progetto
- funzionalità di conversione incluse nei requisiti utente per il progetto
- fattore di aggiustamento del valore dell'applicazione

### Considerazioni circa le GSC per i progetti di manutenzione evolutiva

Le 14 Caratteristiche Generali del Sistema (GSC) opzionali, definite in questa appendice, dovrebbero essere riviste a seguito di cambiamenti del sistema. Piccole evolutive normalmente non richiedono tale revisione. Esempi di cambiamenti che possono richiedere la revisione delle GSC includono:

- Aggiunta di funzionalità interattive ad applicazioni batch
- Attività di progettazione delle prestazioni e test in seguito all'aumento del volume delle transazioni e/o al degrado dei tempi di risposta.
- Richiesta di aggiungere aspetti di usabilità
- Aggiunta di un interfaccia Web ad una applicazione interattiva già esistente
- Aggiunta di un nuovo protocollo di comunicazione ad una applicazione già esistente

### Funzionalità Applicative

Le funzionalità applicative si compongono di:

- Function point identificati dalle funzionalità aggiunte (ADD) durante il progetto di manutenzione evolutiva
- Function point misurati a valle delle modifiche (CHGA – Change *after*) apportate alle funzionalità esistenti durante il progetto di manutenzione evolutiva
- Function point misurati per funzionalità cancellate (DEL) durante il progetto di manutenzione evolutiva

### Funzionalità di Conversione

La funzionalità di conversione è data dai function point rilasciati in seguito alla richiesta dell'utente di ottenere funzioni di conversione.

## Fattore di aggiustamento del valore (VAF)

I due fattori di aggiustamento del valore sono:

- il fattore di aggiustamento del valore dell'applicazione *prima* (VAFB – VAF *before*) dell'inizio del progetto di manutenzione evolutiva
- il fattore di aggiustamento del valore dell'applicazione *dopo* (VAFA – VAF *after*) che il progetto di manutenzione evolutiva è completato

## Formula: Dimensione funzionale pesata per un progetto di manutenzione evolutiva (aEFP)

Usare la seguente formula per calcolare la dimensione funzionale pesata per un progetto di manutenzione evolutiva.

**Nota:** i requisiti di conversione dei dati *sono inclusi* in questo calcolo.

$$aEFP = [(ADD + CHGA + CFP) \times VAFA] + (DEL \times VAFB)$$

Where:

aEFP è la dimensione funzionale pesata del progetto di manutenzione evolutiva

ADD è la dimensione delle funzioni aggiunte dal progetto di manutenzione evolutiva (ADDED Function Point)

CHGA è la dimensione delle funzioni modificate dal progetto di manutenzione evolutiva così come sono/saranno *dopo* l'implementazione (CHANGED Function Point After).

CFP è la dimensione delle funzionalità di conversione (Conversion Function Point)

VAFA è il fattore di aggiustamento del valore dell'applicazione *dopo* il completamento del progetto di manutenzione evolutiva (Value Adjustment Factor After)

DEL è la dimensione delle funzioni cancellate dal progetto di manutenzione evolutiva (DELETED Function Point)

VAFB è il fattore di aggiustamento del valore dell'applicazione *prima* che il progetto di manutenzione evolutiva abbia avuto inizio (Value Adjustment Factor Before)

**Nota:** Al momento dell'installazione di un progetto di manutenzione evolutiva, la dimensione funzionale dell'applicazione deve essere aggiornata per riflettere i cambiamenti delle funzionalità applicative.

## Esempio: Dimensione funzionale pesata per un progetto di manutenzione evolutiva (aEFP)

Questa sezione riporta l'esempio di un progetto di manutenzione evolutiva. I requisiti per tale progetto prevedono le seguenti modifiche:

- L'utente non ha più la necessità di inserire un lavoro in modalità interattiva, perciò quella funzionalità deve essere o è stata rimossa.
- L'utente ha la necessità di un ulteriore prospetto sui Lavori che preveda dei totali.
- Sono richiesti ulteriori DET per inserire i lavori in modalità batch e per correggere le transazioni sospese. Inoltre, per questa stessa transazione, è aggiunto un riferimento alla gestione delle autorizzazioni speciali (un ulteriore FTR).

### Funzionalità Applicative

I paragrafi seguenti espongono le funzionalità applicative contate per il progetto di manutenzione evolutiva di esempio. Le funzionalità si dividono in aggiunte, modificate e cancellate.

#### Funzionalità Aggiunte

La seguente tabella illustra la complessità funzionale per la funzionalità aggiunta, misurata quando il progetto è stato completato.

**Nota:** Fornire un nuovo prospetto rappresenta un ulteriore output esterno.

Funzioni di Tipo Transazionale	FTR	DET	Complessità Funzionale
<b>Output Esterno (EO)</b>			
Prospetto Lavori	1	15	Bassa

#### Funzionalità Modificate (CHG)

La seguente tabella illustra la complessità funzionale per le funzionalità modificate, valutata in termini delle funzioni *dopo* il completamento del progetto di manutenzione evolutiva.

**Nota:** La complessità per inserire un lavoro è aumentata a causa di un ulteriore tipo di file referenziato. La complessità per correggere le transazioni sospese rimane bassa.

**Dopo (CHGA)**

Funzioni di Tipo Transazionale	FTR	DET	Complessità Funzionale
<b>Input Esterni (EI)</b>			
Inserimento Lavoro (batch)	3	8	Alta
Correzione Lavori Sospesi	1	8	Bassa

**Prima (CHGB)**

Funzioni di Tipo Transazionale	FTR	DET	Complessità Funzionale
<b>Input Esterni (EI)</b>			
Inserimento Lavoro (batch)	2	6	Media
Correzione Lavori Sospesi	1	7	Bassa

**Funzionalità Cancellate (DEL)**

La seguente tabella illustra la complessità funzionale per le funzionalità cancellate, identificate alla fine del progetto.

Funzioni di Tipo Transazionale	FTR	DET	Complessità Funzionale
<b>Input Esterni (EI)</b>			
Inserimento Lavoro (a video)	1	7	Bassa

## Contributo dell'Applicazione alla dimensione funzionale

I seguenti paragrafi spiegano il contributo delle funzionalità applicative alla dimensione funzionale non pesata.

### Funzionalità Aggiunte (ADD)

La seguente tabella illustra il contributo alla dimensione funzionale per le funzionalità aggiunte, identificate alla fine del progetto.

Tipo Funzione	Complessità Funzionale			Totali Complessità	Totali Tipo Funzione
EO	1	Bassa	X 4 =	4	
	0	Media	X 5 =	0	
	0	Alta	X 7 =	0	
					ADD 4

### Funzionalità Modificate

La seguente tabella illustra il contributo alla dimensione funzionale per le funzionalità modificate, identificate alla fine del progetto.

#### Dopo (CHGA)

Tipo Funzione	Complessità Funzionale			Totali Complessità	Totali Tipo Funzione
EI	1	Bassa	X 3 =	3	
	0	Media	X 4 =	0	
	1	Alta	X 6 =	6	
					CHGA 9

#### Prima (CHGB)

Tipo Funzione	Complessità Funzionale			Totali Complessità	Totali Tipo Funzione
EI	1	Bassa	X 3 =	3	
	1	Media	X 4 =	4	
	0	Alta	X 6 =	0	
					CHGB 7

### Funzionalità Cancellate (DEL)

La seguente tabella illustra il contributo alla dimensione funzionale per le funzionalità cancellate.

Tipo Funzione	Complessità Funzionale			Totali Complessità	Totali Tipo Funzione
EI	1	Bassa	X 3 =	3	
	0	Media	X 4 =	0	
	0	Alta	X 6 =	0	
					DEL 3

### Calcolo Finale

Il fattore di aggiustamento del valore *prima* che il progetto iniziasse era pari a 1.05. Il fattore di aggiustamento del valore è rimasto lo stesso anche *dopo* che il progetto è stato completato.

La dimensione funzionale pesata per il progetto di manutenzione evolutiva è mostrata di seguito usando i valori della complessità e del contributo per questo esempio.

$$aEFP = [(ADD + CHGA + CFP) \times VAFA] + (DEL \times VAFB)$$

$$aEFP = [(4 + 9 + 0) \times 1.05] + (3 \times 1.05)$$

$$aEFP = 16.8 \text{ (arrotondato a 17)}$$

## Dimensione Funzionale pesata per un'Applicazione (aAFP)

Questa sezione riporta le formule per calcolare la dimensione funzionale pesata per un'applicazione. Ci sono due varianti di questa formula:

- Formula per stabilire la dimensione funzionale pesata per un'applicazione
- Formula per aggiornare la dimensione funzionale pesata per un'applicazione *dopo* che un progetto di manutenzione evolutiva ne ha variato le funzionalità

### Formula: Dimensione Funzionale pesata Iniziale (aAFP)

Usa la seguente formula per stabilire la dimensione funzionale pesata iniziale per un'applicazione. Inizialmente, l'utente riceve funzionalità nuove. Non si ha modifica di funzionalità esistenti o cancellazione di funzionalità non necessarie o obsolete. La dimensione funzionale pesata dell'applicazione *non* include i requisiti di conversione.

$$\text{aAFP} = \text{ADD} \times \text{VAF}$$

Dove:

aAFP è la dimensione funzionale pesata iniziale dell'applicazione

ADD è la dimensione delle funzioni consegnate all'utente dal progetto di sviluppo o delle funzionalità che esistono (baseline) quando l'applicazione viene misurata (ADDed Function Point).

VAF è il fattore di aggiustamento del valore dell'applicazione (Value Adjustment Factor After).

## Formula: Dimensione Funzionale Pesata per un'Applicazione dopo un progetto di Manutenzione Evolutiva (aAFPA)

Dopo l'installazione di un progetto di manutenzione evolutiva, la dimensione funzionale esistente dell'applicazione deve essere aggiornata per riflettere le modifiche apportate all'applicazione. La funzionalità per l'applicazione può essere alterata in uno o più dei seguenti modi:

- aggiungere (nuove) funzionalità aumenta la dimensione dell'applicazione
- modificare funzionalità aumenta, diminuisce o non ha effetti sulla dimensione dell'applicazione
- cancellare funzionalità diminuisce la dimensione dell'applicazione
- modifiche al fattore di aggiustamento del valore aumentano o diminuiscono la dimensione dell'applicazione

**Nota:** Poiché le funzionalità di conversione non influenzano la dimensione funzionale pesata dell'applicazione, qualsiasi funzionalità di conversione associata con il progetto di manutenzione evolutiva è interamente omessa dal calcolo della dimensione funzionale pesata dell'applicazione.

Usare la seguente formula per calcolare la dimensione funzionale pesata dell'applicazione dopo un progetto di manutenzione evolutiva:

$$aAFPA = [(AFPB + ADD + CHGA) - (CHGB + DEL)] \times VAFA$$

Dove:

aAFPA è la dimensione funzionale pesata dell'applicazione *dopo* il progetto di manutenzione evolutiva

AFPB è la dimensione funzionale dell'applicazione *prima* dell'inizio del progetto di manutenzione evolutiva (Adjusted Application Functional Point Before)

ADD è la dimensione delle funzioni che sono state aggiunte dal progetto di manutenzione evolutiva (ADDED function point).

CHGA è la dimensione delle funzioni che sono state modificate dal progetto di manutenzione evolutiva così come sono/saranno dopo il rilascio (CHANGED function point After).

CHGB è la dimensione delle funzioni che sono state modificate dal progetto di manutenzione evolutiva così come sono/erano prima che il progetto iniziasse (CHANGED function point Before).

DEL è la dimensione delle funzioni che sono state cancellate dal progetto di manutenzione evolutiva (DELETED function point).

VAFA è il fattore di aggiustamento del valore dell'applicazione *dopo* il completamento del progetto di manutenzione evolutiva



(Value Adjustment Factor After).

**Nota:** se il valore per AFPB non è disponibile, può essere calcolato usando la formula  $AFPB = aAFPB / VAFB$ ; dove  $aAFPB$  è la dimensione funzionale pesata dell'applicazione *prima* del progetto di manutenzione evolutiva e  $VAFB$  è il fattore di aggiustamento del valore dell'applicazione *prima* del progetto di manutenzione evolutiva.

## Esempio: Conteggio per un'Applicazione

Questa sezione illustra un esempio per la dimensione funzionale pesata iniziale e la dimensione funzionale che riflette un progetto di manutenzione evolutiva. I numeri per queste dimensioni funzionali sono quelli del precedente esempio dell'applicazione fatto in questa appendice.

### Dimensione Funzionale Pesata Iniziale dell'Applicazione(aAFP - Initial Adjusted Application Functional Size)

La dimensione funzionale pesata iniziale dell'applicazione è mostrata di seguito. Il fattore di aggiustamento del valore è 1.05.

$$aAFP = ADD \times VAF$$

$$aAFP = 115 \times 1.05$$

$$aAFP = 120.75 \text{ (arrotondato a 121)}$$

**Nota:** la dimensione iniziale include solo la dimensione delle funzionalità applicative installate per l'utente

### Dimensione funzionale pesata dell'applicazione dopo un progetto di manutenzione evolutiva (aAFPA - Adjusted Application Functional Size After)

La dimensione funzionale pesata dell'applicazione che riflette un progetto di manutenzione evolutiva è mostrato di seguito. Il fattore di aggiustamento del valore è 1.05.

$$aAFPA = [(AFPB + ADD + CHGA) - (CHGB + DEL)] \times VAFA$$

$$aAFPA = [(115 + 4 + 9) - (7 + 3)] \times 1.05$$

$$aAFPA = 123.9 \text{ (arrotondato a 124)}$$

Questa pagina è lasciata intenzionalmente bianca.

---

## Glossario IFPUG

---

Questo è un glossario completo dei termini utilizzati nelle pubblicazioni dell'IFPUG.

**Agevolazione delle modifiche (GSC #14).** Una delle 14 caratteristiche generali del sistema che descrive in che misura l'applicazione è stata sviluppata in modo da facilitare la modifica della logica elaborativa o della struttura dei dati.

**Aggiornamento (*Refresh*).** Il processo che ricrea un insieme di dati per renderli aggiornati (attuali) rispetto alla loro fonte di origine.

**Aggiornamento interattivo (GSC #8).** Una delle 14 caratteristiche generali del sistema che definisce in che misura i file logici interni sono aggiornati interattivamente.

**Albrecht 1984.** Documento originale sul concetto di function point, scritto da Allan J. Albrecht nel Novembre del 1984. Anche conosciuto come "313" per il numero associato al documento.

**Ambito del conteggio.** L'ambito del conteggio definisce l'insieme dei Requisiti Utente Funzionali (FUR) da includere nel conteggio dei function point.

**Analisi dei function point.** Il metodo per la misurazione della dimensione funzionale così come definito nel metodo IFPUG di misurazione della dimensione funzionale (FSM).

**Andamento (*Trend*).** Un'analisi temporale che evidenzia occorrenze ripetute di una particolare misura o metrica.

**Applicazione.** Una collezione integrata di procedure automatizzate e dati che forniscono supporto ad un obiettivo applicativo; è formata da uno o più componenti, moduli, o sottosistemi.

**Area Applicativa.** Termine generale che indica il raggruppamento di applicazioni relative ad una specifica area di interesse. Corrisponde ad un livello amministrativo in ambito gestionale.

**Attributo dati.** Una caratteristica di un'entità. Gli attributi dei dati sono generalmente l'analogo dei tipi di elementi dati (DET). Anche conosciuto come 'campo'.

**Attributo del progetto/applicazione.** Una caratteristica di un progetto o di una applicazione che può avere un impatto significativo sulla produttività. Esempi di tali attributi sono la piattaforma hardware, l'esperienza del personale, gli strumenti e la metodologia. L'attributo del progetto/applicazione è usato per classificare i dati del progetto durante l'analisi.

**Attributo tecnico.** Attributo non riconoscibile dall'utente, risultante da considerazioni di progettazione o implementative.

**Autosufficiente.** Nessuna attività elaborativa precedente o successiva è necessaria per iniziare o completare il/i FUR.

**Bene.** (1) un bene capitale dell'impresa. (2) Un profitto o una risorsa.

**Budget.** Una sequenza di spese pianificate nel tempo con costi monetari assegnati a specifici obiettivi o impegni. Spesso usato indifferentemente per indicare l'impegno lavorativo o il costo monetario.

**Campo.** Vedi *Attributo dati*.

**Caratteristiche generali del sistema (GSC – General System Characteristics).** Le caratteristiche generali del sistema sono un insieme di 14 proprietà che valutano la complessità globale dell'applicazione.

**Caricamento (*Load*).** La copia di istruzioni di elaborazione o di dati dalla memoria esterna alla memoria interna. (IEEE).

**Chiave esterna (*foreign key*).** Dato in un ILF o EIF che esiste in quanto l'utente richiede una relazione con un altro ILF o EIF.

**Comitato per le regole del conteggio (CPC – Counting Practices Committee).** Il comitato di lavoro che gestisce il Manuale IFPUG delle Regole di Conteggio.

**Complessità funzionale.** La valutazione della complessità specifica assegnata ad una funzione che usa le regole definite nella presente norma internazionale.

**Componente funzionale di base (BFC - Base Functional Component).** Unità elementare di Requisiti Utente Funzionale (FUR) definita e usata da un metodo FSM (ISO/IEC 14143-1:2007) ILF, EIF, EI, EO, EQ sono tipi di BFC.

**Comunicazione dei dati (GSC #1).** Una delle 14 caratteristiche generali del sistema che descrive il grado di comunicazione diretta tra l'applicazione e l'elaboratore.

**Confine.** Il confine è una interfaccia concettuale tra il software in esame e i suoi utenti.

**Confine applicativo.** Vedi *Confine*.

**Confine dell'applicazione.** Vedi *Confine*.

**Conteggio dei function point.** L'attività di applicazione delle regole del Metodo di Misurazione della Dimensione Funzionale (FSM) IFPUG per misurare la dimensione funzionale di un'applicazione o progetto. NOTA: esistono tre tipi di conteggio dei function point: applicazione (AFP), progetto di sviluppo (DFP) e progetto di manutenzione evolutiva (EFP).

**Conteggio in function point di un'applicazione.** Attività di esecuzione del metodo IFPUG di misurazione della dimensione funzionale (FSM) per misurare la dimensione funzionale di un'applicazione.

**Conteggio di function point non pesati (UFP).** Vedi *Conteggio dei function point*.

**Conteggio in function point dell'installazione.** Vedi *Conteggio in function point dell'applicazione*.

**Conteggio in function point di baseline.** Vedi il *Conteggio in function point di un'applicazione*.

**Conteggio in function point di un progetto di manutenzione evolutiva.** Applicazione del metodo IFPUG di misurazione della dimensione funzionale (FSM – Functional Size Measurement) per misurare la dimensione funzionale di un progetto di manutenzione evolutiva.

**Conteggio in function point di un progetto di sviluppo.** Applicazione del metodo IFPUG di misurazione della dimensione funzionale (FSM-Functional Size Measurement) per misurare la dimensione funzionale di un progetto di sviluppo.

**Contributo.** Il contributo dei tipi di funzione (ILF, EIF, EI, EO, EQ) apportato alla dimensione funzionale.

**Copia.** (1) Lettura di dati da una fonte, senza modifica dei dati di origine, e scrittura degli stessi dati in altra posizione, in un formato fisico che può differire da quello di origine. Per esempio, copiare

dati da un disco magnetico su un nastro magnetico. (2) Il risultato di un processo di copia come descritto sopra. Per esempio, una copia di un file di dati (IEEE).

**Correzione.** L'eliminazione dei difetti causati da errori nell'analisi, nel disegno o nella codifica. Esempi sono funzioni mancanti che non comportano l'interruzione dell'applicazione (errore di analisi) o errori che determinano il blocco dell'esecuzione dell'applicazione (errore nella codifica).

**Dati derivati.** Dati risultanti da un processo che coinvolge passi, oltre che o in aggiunta al, diretto reperimento e validazione di informazioni da funzioni di tipo dati.

**Dati di Business.** Indicati anche come Dati Fondamentali dell'Utente o Oggetti di Business. Questo tipo di dati rispecchia le informazioni che devono essere memorizzate e reperite da parte delle aree funzionali coinvolte dall'applicazione. I Dati di business di solito rappresentano una percentuale significativa delle entità identificate.

**Dati di decodifica.** L'utente non sempre specifica direttamente i Dati di Decodifica, talvolta indicati come Dati delle Liste o Dati di Traduzione. In altri casi sono identificati dallo sviluppatore in risposta a uno o più requisiti tecnici dell'utente. I Dati di Decodifica forniscono una lista di valori validi che un attributo di tipo descrittivo può possedere. Tipicamente, gli attributi dei Dati di Decodifica sono il Codice, la Descrizione e/o altri attributi 'standard' che descrivono il codice; per es. abbreviazione standard, data di inizio validità, data di fine validità, dati di traccia di audit, ecc.

**Dati di riferimento.** Questo tipo di dati viene memorizzato per supportare le regole di business per la gestione dei Dati di Business; per es. in un'applicazione relativa agli stipendi dovrebbero essere memorizzati i dati relativi alle aliquote fiscali per ogni scaglione d'importo e la data a partire dalla quale ogni aliquota diventa effettiva. I Dati di Riferimento di solito rappresentano una piccola percentuale delle entità identificate.

**Delivery Rate.** Vedi *Tasso di rilascio*.

**DET.** Vedi *Tipo di elemento dati*.

**Difetto.** Un problema che, se non corretto, potrebbe causare il malfunzionamento di un'applicazione o risultati inesatti. È considerato un difetto anche la mancanza di funzionalità che erano state specificate o richieste.

**Dimensione funzionale.** Dimensione del software derivata dalla quantificazione dei requisiti utente funzionali. (ISO 14143-1:2007).

**Dimensione funzionale dell'applicazione.** La dimensione funzionale dell'applicazione è una misura delle funzionalità che un'applicazione fornisce all'utente, determinata dall'applicazione del Metodo di Misurazione della Dimensione Funzionale (FSM) IFPUG. Vedi *Parte 1, Sez. 5.8* per la formula.

**Dimensione funzionale di un progetto di manutenzione evolutiva.** La dimensione funzionale di un progetto di manutenzione evolutiva è una misura delle funzionalità aggiunte, modificate o cancellate al termine di un progetto di manutenzione evolutiva, misurata con il conteggio dei function point del progetto di manutenzione evolutiva (EFP) applicando il Metodo di Misurazione della Dimensione Funzionale (FSM) IFPUG. Vedi *Parte 1, Sez. 5.8* per la formula.

**Dimensione funzionale di un progetto di sviluppo.** La dimensione funzionale di un progetto di sviluppo è una misura delle funzionalità fornite agli utenti con il primo rilascio del software, misurata con il conteggio dei function point di un progetto di sviluppo (DFP) applicando il Metodo di Misurazione della Dimensione Funzionale (FSM) IFPUG. Vedi *Parte 1, Sez. 5.8* per la formula.

**Dimensione funzionale non pesata.** Vedi *Dimensione funzionale*.

**Dimensione funzionale pesata (AFP).** La dimensione funzionale pesata è il risultato della dimensione funzionale non pesata moltiplicata per il fattore di aggiustamento del valore (VAF). La dimensione funzionale pesata è calcolata usando una formula specifica per un progetto di sviluppo, o di manutenzione evolutiva o di applicazione, come illustrato in *Appendice C*.

**Dimensione funzionale pesata di una applicazione (aAFP- Adjusted application functional size).** Vedi App. C.

**Dimensione funzionale pesata di una applicazione dopo progetti di manutenzione evolutiva (aAFA- Adjusted application functional size after enhancement projects).** Vedi *Appendice C* per la formula.

**Dimensione funzionale pesata di un progetto di manutenzione evolutiva (aEFP. Adjusted enhancement project functional size).** Vedi *Appendice C* per la formula.

**Dimensione funzionale pesata di un progetto di sviluppo (aDFP – Adjusted development project functional size).** Vedi App. C.

**Efficacia.** Raggiungimento dei risultati fissati o desiderati.

**Efficienza.** Raggiungimento di un risultato riducendo al minimo sforzi non pertinenti o ridondanti.

**Efficienza per l'utente finale (GSC #7).** Una delle 14 caratteristiche generali del sistema che descrive in che misura sono considerati il fattore umano e la facilità d'uso da parte dell'utente dell'applicazione misurata.

**Elaborazione complessa (GSC #9).** Una delle 14 caratteristiche generali del sistema che descrive il grado di influenza della logica elaborativa nello sviluppo dell'applicazione.

**Elaborazione dati distribuita (GSC #2).** Una delle 14 caratteristiche generali del sistema che descrive in che misura l'applicazione trasferisce dati fra le sue componenti.

**Entità (o Tipo di entità).** Un oggetto fondamentale significativo per l'utente, riguardo al quale è conservato un insieme di informazioni. Una relazione fra entità che contiene attributi è essa stessa un'entità.

**Entità dipendente.** Una entità non significativa per il business, dotata di un senso compiuto di per sé, senza la presenza di altre entità tale che:

- Un'occorrenza dell'entità X deve essere collegata ad una occorrenza dell'entità Y, e
- l'eliminazione di una occorrenza dell'entità Y corrisponde alla eliminazione di tutte le occorrenze correlate all'entità X.

**Entità di tipo dati.** Vedi *Entità*.

**Entità indipendente.** Un'entità che è significativa per il business, dotata di senso compiuto di per sé, senza la presenza di altre entità.

**External Input (EI).** Vedi *Input Esterno*.

**External Inquiry (EQ).** Vedi *Interrogazione Esterna*.

**External Interface File (EIF).** Vedi *File d'Interfaccia Esterno*.

**External Output (EO).** Vedi *Output Esterno*.

**Facilità di gestione operativa (GSC #12).** Una delle 14 caratteristiche generali del sistema che descrive in che misura l'applicazione cura gli aspetti operativi, quali procedure di avviamento, salvataggio e ripristino.

**Facilità di installazione (GSC #11).** Una delle 14 caratteristiche generali del sistema che descrive in che misura la conversione da precedenti ambienti ha influenzato lo sviluppo dell'applicazione.

**Fattore di aggiustamento del valore (VAF).** Il fattore che indica le funzionalità generali fornite all'utente dell'applicazione. Il VAF è calcolato sulla base della valorizzazione delle 14 caratteristiche generali (GSC) del sistema per una data applicazione.

**File.** Funzioni di tipo dati come gruppi di dati logici correlati e non le implementazioni fisiche di questi gruppi di dati.

**File d'interfaccia esterno (EIF – External Interface File).** Un EIF è un gruppo di dati logicamente collegati o di informazioni di controllo riconoscibili dall'utente, referenziati dall'applicazione, ma mantenuti all'interno del confine di un'altra applicazione. L'intento primario di un EIF è contenere dati referenziati da uno o più processi elementari all'interno del confine dell'applicazione che si sta misurando. Ciò significa che un EIF contato per un'applicazione deve essere un ILF per un'altra applicazione. Vedi anche *File logico interno (ILF)*.

**File logico.** Vedi *Funzione di tipo dati*.

**File logico interno (ILF – Internal Logical File).** Un ILF è un gruppo di dati logicamente collegati o di informazioni di controllo, riconoscibili dall'utente, mantenuti all'interno del confine dell'applicazione oggetto della misurazione. L'intento primario di un ILF è quello di contenere dati mantenuti da uno o più processi elementari all'interno del confine dell'applicazione che si sta misurando. Vedi anche *File d'interfaccia esterno (EIF)*.

**File system.** È costituito da record e oggetti dati.

**Frequenza delle transazioni (GSC #5).** Una delle 14 caratteristiche generali del sistema che descrive il grado di influenza della logica elaborativa nello sviluppo dell'applicazione.

**Function point (FP).** L'unità di misura per la dimensione funzionale così come definita nel metodo IFPUG di misurazione della dimensione funzionale (FSM).

**Funzionalità.** Vedi *Funzione*.

**Funzionalità di conversione.** Funzioni di tipo dati o transazionali fornite per convertire dati e/o soddisfare altri requisiti di conversione richiesti dall'utente.

**Funzione.** Caratteristiche o capacità di un'applicazione, dal punto di vista dell'utente.

**Funzione di tipo dati.** Una funzione di tipo dati rappresenta una funzionalità fornita all'utente per soddisfare requisiti di memorizzazione di dati interni o esterni. Una funzione di tipo dati può essere rappresentata sia con un ILF che con un EIF.

**Funzione di tipo transazionale.** Una funzione di tipo transazionale è un processo elementare che fornisce funzionalità all'utente per processare dati. Una funzione di tipo transazionale può essere rappresentata tramite un External Input (EI), External Output (EO) o un External Inquiry (EQ).

**Grado di influenza (DI – Degree of Influence).** Un indicatore numerico che quantifica l'impatto di ciascuna delle 14 caratteristiche generali del sistema; può variare da 0 a 5. Questi indicatori sono usati per calcolare il fattore di aggiustamento del valore (VAF).

**Grado totale di influenza (TDI – Total Degree of Influence).** La somma dei gradi di influenza di ciascuna delle 14 caratteristiche generali di sistema (GSC).

**IBM CIS & A Guideline 313.** Vedi *Albrecht 1984*.

**IFPUG.** Il Gruppo Internazionale di Utenti di Function Point è una organizzazione senza fini di lucro, gestita dagli associati, che promuove e sostiene il metodo di misurazione in Function Point ed altre tecniche di misurazione del software.

**Immagine.** Una replica esatta di un altro oggetto, file, o tabella, creata solitamente per mezzo di una funzione di *'utilità'*.

**Impegno lavorativo (Work effort).** Forza-lavoro richiesta per la produzione di uno specifico output. In questo caso riferito all'impegno richiesto per sviluppare o mantenere un'applicazione. La forza-lavoro è tipicamente espressa in ore-lavoro.

**Informazioni di controllo.** Le informazioni di controllo sono dati che influenzano un processo elementare e specificano cosa, quando o come i dati devono essere elaborati.

**Input esterno (EI).** Un EI è un processo elementare che elabora dati o informazioni di controllo che provengono dall'esterno del confine dell'applicazione. L'intento primario di un EI è di mantenere uno o più ILF e/o alterare il comportamento del sistema. Vedi anche *Interrogazione esterna (EQ)* e *Output esterno (EO)*.

**Inserimento dati interattivo (GSC #6).** Una delle 14 caratteristiche generali del sistema che definisce in che misura l'inserimento dei dati avviene attraverso transazioni interattive.

**Intento primario.** L'intento con la massima priorità.

**Internal Logical File (ILF).** Vedi *File logico interno*.

**Interrogazione esterna (EQ).** Una EQ è un processo elementare che invia dati o informazioni di controllo all'esterno del confine dell'applicazione. L'intento primario di un'interrogazione esterna è presentare le informazioni ad un utente, mediante

il reperimento di dati od informazioni di controllo da ILF o EIF. Il processo logico non contiene né formule matematiche né calcoli e non crea dati derivati. Durante il processo non è mantenuto alcun ILF, né è alterato il comportamento del sistema. Vedi anche *Input esterno (EI)* e *Output esterno (EO)*.

**Investimento di capitale.** Una forma di spesa in cui un'impresa investe denaro (capitale) per l'acquisizione di oggetti tangibili come forniture, computer e simili.

**Livello dell'area applicativa.** Il livello di direzione responsabile, per una o più applicazioni, della gestione delle attività di manutenzione, di progetti di nuovo sviluppo o di manutenzione evolutiva di grande entità.

**Livello dell'organizzazione.** Il livello o i livelli di direzione responsabile/i della gestione di una o più unità organizzative di elaborazione di dati o di sistemi informativi.

**Livello di progetto.** Il livello di direzione responsabile della gestione di singoli progetti di nuovo sviluppo o di manutenzione evolutiva di grande entità.

**Livello direttivo di impresa.** Il livello di direzione responsabile dell'impresa nel suo complesso, compreso il Consiglio di Amministrazione.

**Logica elaborativa.** Vedi *Trattamento logico*.

**Mantenere.** Il termine mantenere si riferisce alla capacità di aggiungere, modificare o cancellare dati attraverso un processo elementare.

**Manutenzione.** L'impegno per mantenere un'applicazione conforme alle sue specifiche, generalmente senza cambiarne le funzionalità (o la sua dimensione funzionale). La manutenzione comprende attività di correzione, lieve miglioramento, conversione, supporto utente e manutenzione preventiva. Le attività comprendono la rimozione dei difetti (vedi *Correzione*), gli aggiornamenti hardware o software (vedi *Conversione*), l'ottimizzazione o il miglioramento della qualità (vedi *Manutenzione migliorativa* e *Manutenzione preventiva*) e il supporto all'utente.

**Manutenzione Adeguativa.** La modifica di un prodotto software, eseguita dopo il rilascio, per mantenere fruibile un programma in un ambiente mutato o in corso di cambiamento. La manutenzione adeguativa comprende le modifiche necessarie ad adattare i cambiamenti nell'ambiente nel quale un prodotto software deve operare. Tali cambiamenti sono quelli che devono essere effettuati per essere allineati con il nuovo ambiente. Ad esempio il sistema operativo potrebbe essere aggiornato ed alcune modifiche possono essere effettuate per ospitarlo. (ISO/IEC 14764:2006).

**Manutenzione correttiva.** La modifica reattiva ad un prodotto software effettuata dopo il rilascio al fine di correggerne le problematiche rilevate. La modifica ripara il prodotto software per poter soddisfare i requisiti (ISO/IEC 14764:2006).

**Manutenzione perfettiva.** Modifica di un prodotto software effettuata dopo il rilascio al fine di individuare e correggere errori latenti nel prodotto software prima che degenerino in malfunzionamenti. La manutenzione perfettiva fornisce migliorie per gli utenti, miglioramenti della documentazione dei programmi, e una modifica del codice per migliorare le prestazioni del software, la sua manutenibilità, o altri possibili attributi. Contrapposta a: manutenzione adeguativa; manutenzione correttiva. (ISO/IEC 14764:2006).

**Manutenzione preventiva.** Modifiche all'hardware o al software realizzate per prevenire futuri difetti o malfunzionamenti. Ad esempio, la ristrutturazione di programmi o dati per migliorare la manutenibilità o per prevenire difetti.

**Maturità (SEI - Software Engineering Institute).** La capacità di un'organizzazione di conseguire un processo controllato e misurato, come richiesto nei fondamenti del miglioramento continuo (W. Humphrey). Il livello di esperienza di un'organizzazione o di un progetto con l'uso di un particolare strumento, risorse, tecniche o metodologie.

**Media/Medium.** Mezzo di comunicazione o informazione, per esempio, un rapporto prodotto su carta o su microfiche.

**Metrica.** Non esiste una singola definizione universale di metrica. Nel contesto del presente manuale, una metrica è una combinazione di una o più misure o attributi. Alcuni esempi sono (1) densità di difetti (*difetti per function point*), (2) tasso di rilascio (*function point per ora-lavoro*).

**Misura.** Inteso come sostantivo, esprime un numero che assegna un valore relativo. Alcuni esempi sono il volume, l'altezza, i function point o l'impegno lavorativo. Inteso come verbo, significa accertare o valutare mediante il confronto con uno standard.

**Misurazione.** Assegnazione di un valore relativo. In genere nel processo di miglioramento, le misure ottenute attraverso questa attività sono combinate per ottenere delle metriche.

**Misure di processo.** Informazioni raccolte relative al processo di sviluppo.

**Misure di prodotto.** Informazioni raccolte relative all'applicazione software sviluppata.

**Molteplicità di siti (GSC #13).** Una delle 14 caratteristiche generali del sistema che descrive in che misura l'applicazione è stata sviluppata per essere utilizzata in più siti e da più organizzazioni utente.

**NESMA.** Il Netherlands Software Metrics Association. Un'organizzazione olandese *no profit*, governata dai membri, che si propone di promuovere e supportare l'analisi dei function point e altri metodi di misurazione del software.

**Normalizzazione.** Il processo attraverso il quale qualsiasi struttura dati può essere trasformata da un progettista di base dati in un insieme di relazioni normalizzate che non hanno gruppi ripetuti.

**Ordinamento.** L'attività di sequenziamento di righe o record in una funzione di tipo transazionale.

**Output esterno (EO).** Un EO è un processo elementare che invia dati o informazioni di controllo all'esterno del confine dell'applicazione e include un'elaborazione ulteriore rispetto a quella di un'interrogazione esterna. L'intento primario di un output esterno è presentare informazioni all'utente mediante un processo logico diverso da o in aggiunta al reperimento di dati o informazioni di controllo. Il processo logico deve contenere almeno una formula matematica od un calcolo o creare dati derivati. Un output esterno può anche mantenere uno o più ILF e/o alterare il comportamento del sistema. Vedi anche *Input esterno* (EI) e *Interrogazione esterna* (EQ).

**Prima forma normale.** Risultato di un processo di normalizzazione che trasforma gruppi di dati in modo tale che essi abbiano un identificativo unico, uno o più attributi e nessun valore ripetuto.

**Processo elementare.** Un processo elementare è la più piccola unità di attività che è significativa per l'utente/i.

**Prodotto realizzato (Work product).** Il prodotto creato dal lavoro dei sistemi informativi, in questo caso espresso quale risultato dell'impegno speso per lo sviluppo di un software.

**Produttività.** Il rapporto tra quanto prodotto realizzato e l'impegno per realizzarlo. Vedi anche *Tasso di rilascio*.

**Progetto.** Un insieme di attività da svolgere in un arco temporale delimitato al fine di rilasciare un prodotto.

**Progetto di manutenzione evolutiva.** È un progetto che ha l'obiettivo di sviluppare e rilasciare manutenzione adeguativa.

**Progetto di sviluppo.** È un progetto che ha l'obiettivo di sviluppare e rilasciare la prima versione di un'applicazione software.

**Project leader.** Persona che gestisce o dirige progetti. Può essere inteso come sinonimo di Project Manager.

**Project manager.** Persona che coordina uno o più progetti o gruppi di progetti.

**Punto di vista dell'utente (User View).** Il punto di vista dell'utente è espresso dai Requisiti Utente Funzionali (FUR) così come percepiti dall'utente stesso.

**Qualità.** La qualità comprende: la conformità alle aspettative dell'utente, la conformità ai requisiti dell'utente, la soddisfazione del cliente, l'affidabilità ed il livello di difettosità presente. Il contesto e la politica adottata consentiranno la miglior definizione per una data situazione.

**Rapporto.** Nel contesto di questo documento, il rapporto è definito come il risultato della divisione di una quantità misurata per un'altra.

**Record.** Gruppo di elementi collegati, trattati come un'unità.

**RECUP.** Acronimo per Repair/Enhancement/Conversion/User support/Prevention. Vedi anche *Tasso di manutenzione ordinaria (supporto)*.

**Relazione.** Una associazione significativa fra due entità. Una relazione non ha attributi e non è contata come RET durante il conteggio dei function point.

**Requisiti sulla qualità.** Qualsiasi requisito correlato alla qualità del software così come definito nella norma ISO/IEC 9126.

**Requisiti tecnici.** Requisiti legati alla tecnologia e all'ambiente, per lo sviluppo, la manutenzione, il supporto e l'esecuzione del software.

**Requisiti utente funzionali (FUR).** Un sotto-insieme dei requisiti utente che descrivono cosa il software dovrà fare, in termini di attività e servizi (ISO 14143-1:2007)

**Responsabile applicativo.** Una persona responsabile della gestione di progetti e di attività di supporto per una o più aree applicative.

**RET.** Vedi *Tipo di elemento record*.

**Riconoscibile dall'utente.** Il termine riconoscibile dall'utente è relativo ai requisiti per i processi e/o dati concordati, e compresi sia da utente(i) che sviluppatore(i) software

**Rilascio (Release).** Una versione consegnata di una applicazione che può includere tutta o solo parte di una applicazione.

**Rimozione dei difetti.** Vedi *Correzione*.



**Riorganizzazione.** L'attività di ordinamento degli attributi in una funzione di tipo transazionale.

**Riusabilità (GSC #10).** Una delle 14 caratteristiche generali del sistema che descrive in che misura l'applicazione ed il codice nell'applicazione sono stati specificamente disegnati, sviluppati e supportati per essere riutilizzabili in altre applicazioni.

**Scopo del conteggio.** La motivazione per eseguire di un conteggio in function point.

**Seconda forma normale.** Risultato di un processo di normalizzazione che trasforma gruppi di dati in modo che ogni attributo non chiave dipenda da tutti gli attributi chiave del gruppo di dati e non da una parte di essi.

**Significativo.** È riconoscibile dall'utente e soddisfa un Requisito Utente Funzionale (FUR)

**Sistema.** Vedi *Applicazione*.

**Sottogruppo obbligatorio.** Uno dei due tipi di sottogruppi per i tipi di elementi record (RET). Sottogruppi obbligatori significa che l'utente deve usare almeno uno dei sottogruppi durante un processo elementare che crea un'occorrenza dei dati.

**Sottogruppo opzionale.** Uno dei due tipi di sottogruppi per i tipi di elementi record (RET). 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.

**Sottotipi.** Vedi *Entità Sottotipo*.

**Sottotipo di entità.** Una specializzazione di un tipo di entità. Un sottotipo eredita tutti gli attributi e relazioni dell'entità padre e può avere ulteriori attributi specifici e relazioni. Vedi anche *Entità*.

**Stato di coerenza funzionale.** Il momento nel quale l'elaborazione è stata completamente eseguita; il FUR è stato soddisfatto e nessuna ulteriore azione è richiesta.

**Supporto.** Vedi *Manutenzione*.

**Tasso di manutenzione ordinaria (supporto).** La misura di produttività per la manutenzione ordinaria di un'applicazione. È espresso dal rapporto tra l'impegno lavorativo per categoria di manutenzione ordinaria diviso 1000 function point dell'applicazione, in un dato intervallo di tempo.

**Tasso di rilascio (Delivery Rate).** Misura di produttività per lo sviluppo o la manutenzione evolutiva di un'applicazione. È espresso, dal rapporto tra i function point del progetto e il relativo impegno lavorativo (*work effort*).

**Terza forma normale.** Risultato di un processo di normalizzazione che trasforma gruppi di dati in modo che ogni attributo non chiave non dipenda da un altro attributo non chiave.

**Tipo di elemento dati (DET).** Un tipo di elemento dati è un attributo unico, riconoscibile dall'utente e non ripetuto.

**Tipo di elemento record (RET).** Un tipo di elemento record è un sottogruppo, riconoscibile dall'utente, di tipi di elemento dati all'interno di una funzione di tipo dati.

**Tipo di entità associativa.** Un tipo di entità con attributi che forniscono ulteriori dettagli su una relazione multi-a-molti fra altri due tipi di entità. Vedi anche *Entità*.

**Tipo di entità attributiva.** Un tipo di entità che fornisce dettagli su uno o più attributi di un altro tipo di entità. Vedi anche *Entità*.

**Tipo di file referenziato (FTR).** Un tipo di file referenziato è una funzione di tipo dati letta e/o mantenuta da una funzione di tipo transazionale.

**Tipo di funzione.** I cinque tipi di componenti funzionali base (BFC) identificati nella presente norma internazionale. NOTA: i cinque tipi sono: External Input (EI), External Output (EO), External Inquiry (EQ), Internal Logical File (ILF) e External Interface File (EIF).

**Transazione.** Vedi *Funzioni di Tipo Transazionale*.

**Trattamento logico (Processing Logic).** Ciascun requisito specificamente richiesto dall'utente per completare un processo elementare così come validazioni, algoritmi o calcolo e lettura o scrittura di una funzione di tipo dati.

**Variazione dell'ambito (Scope Creep).** Funzionalità aggiuntive che non erano state specificate nei requisiti originali, ma che vengono identificate nel momento in cui l'obiettivo è stato chiarito e le funzioni definite (in precedenza tradotto con '*obiettivi striscianti*' e successivamente con '*slittamento dell'ambito*').

**Unione (Merge).** File multipli con gli stessi elementi dati, consolidati in un unico file. (IEEE).

**Utente.** Un utente è ogni persona o oggetto che comunica o interagisce con il software in ogni istante.

**Utilizzo intensivo della configurazione (GSC #4).** Una delle 14 caratteristiche generali del sistema che descrive in che misura i vincoli sulle risorse elaborative hanno influenzato lo sviluppo dell'applicazione.

**313.** Vedi *Albrecht 1984*.

Questa pagina è stata lasciata intenzionalmente vuota.