



UNIVERSITÀ DI PISA

# SOCIAL NETWORK

CORSO DI LAUREA IN INGEGNERIA INFORMATICA  
PROGETTO PER IL CORSO DI BASI DI DATI  
KEVIN CORIZI, SALVATORE D'AVELLA, FABRIZIO DEGORTES

# SOMMARIO

|  |    |
|--|----|
| INDICE DELLE FIGURE .....  | 2  |
| I. PROGETTAZIONE CONCETTUALE .....                                 | 3  |
| 1. ANALISI DEI REQUISITI .....                                     | 3  |
| 1.1 Frasi relative agli utenti.....                                | 3  |
| 1.2 Frasi relative ai profili.....                                 | 3  |
| 1.3 Frasi relative al rapporto follower-following tra utenti ..... | 3  |
| 1.4 Frasi relative ai messaggi .....                               | 4  |
| 1.5 Frasi relative ai contenuti.....                               | 4  |
| 1.6 Frasi relative ai gruppi.....                                  | 4  |
| 1.7 Frasi relative agli eventi .....                               | 5  |
| 1.8 Frasi relative a sondaggi.....                                 | 5  |
| 1.9 Frasi relative alla riservatezza .....                         | 5  |
| 1.10 Frasi relative ai blocchi.....                                | 6  |
| 2. GLOSSARIO DEI TERMINI.....                                      | 7  |
| 3. SCHEMA INIZIALE .....   | 9  |
| II. PROGETTAZIONE LOGICA.....                                      | 10 |
| 4. RISTRUTTURAZIONE .....  | 10 |
| 5. SCHEMA RISTRUTTURATO .....                                      | 11 |
| 6. TABELLA DELLE ENTITA' .....                                     | 12 |
| 7. TABELLA DELLE ASSOCIAZIONI .....                                | 14 |
| 8. REGOLE AZIENDALI.....   | 19 |
| 9. ANALISI DELLE PRESTAZIONI .....                                 | 20 |
| 9.1 Tavola dei volumi .....  | 20 |
| 9.2 Operazioni .....   | 22 |
| 9.3 Tavole degli accessi .....                                     | 22 |
| 9.3.1 Operazione 1 .....   | 22 |
| 9.3.2 Operazione 2 .....   | 22 |
| 9.3.3 Operazione 3 .....   | 23 |
| 9.3.4 Operazione 4 .....   | 23 |
| 9.3.5 Operazione 5 .....   | 23 |
| 9.3.6 Operazione 6 .....   | 23 |
| 9.3.7 Operazione 7 .....   | 23 |
| 9.3.8 Operazione 8 .....   | 24 |
| 9.4 Ridondanze.....  | 24 |
| 9.4.1 Operazione 3 con ridondanze .....                            | 24 |

|       |  |    |
|-------|--|----|
| 9.4.2 | Operazione 5 con ridondanze .....  | 24 |
| 9.4.3 | Operazione 2 con ridondanze .....  | 25 |
| 9.4.4 | Operazione 6 con ridondanze .....  | 25 |
| 9.5   | Valutazione delle ridondanze.....  | 25 |
| 9.5.1 | Ridondanza 1 – Numero di like su un contenuto.....   | 25 |
| 9.5.2 | Ridondanza 2 – Numero di partecipanti, forse partecipanti e non partecipanti a evento..... | 25 |
| 10.   | SCHEMA DEFINITIVO .....  | 27 |
| 11.   | TRADUZIONE IN MODELLO RELAZIONALE .....  | 28 |
| 12.   | VINCOLI DI INTEGRITA' REFERENZIALE .....   | 31 |
| III.  | NORMALIZZAZIONE.....   | 33 |
| 13.   | Dipendenze funzionali.....   | 33 |
| IV.   | BUSINESS INTELLIGENCE .....  | 39 |
| 14.   | BUSINESS INTELLIGENCE .....  | 39 |
| 14.1  | Operazione Data Mining 1 .....   | 40 |
| 14.2  | Operazione Data Mining 2 .....   | 44 |
| V.    | IMPLEMENTAZIONE IN MYSQL .....   | 49 |
| 15.   | OPERAZIONI .....   | 49 |
| 16.   | REGOLE AZIENDALI.....  | 52 |

## INDICE DELLE FIGURE

|  |    |
|--|----|
| Figura 1 - Diagramma ER con generalizzazioni .....   | 9  |
| Figura 2 - Generalizzazione CONTENUTO prima e dopo la ristrutturazione.....                              | 10 |
| Figura 3 - Generalizzazione GENERALITA' prima e dopo la ristrutturazione .....                           | 10 |
| Figura 4 – Diagramma ER senza generalizzazioni .....   | 11 |
| Figura 5 - Diagramma ER definitivo con ridondanze.....   | 27 |
| Figura 6 - Porzione del diagramma ER utilizzata per la prima operazione di Business Intelligence .....   | 40 |
| Figura 7 - Porzione del diagramma ER utilizzata per la seconda operazione di Business Intelligence ..... | 44 |

# I. PROGETTAZIONE CONCETTUALE

## 1.ANALISI DEI REQUISITI

---

Si vuole realizzare un database che supporti le potenzialità di un social network, come Facebook o Twitter.

### 1.1 FRASI RELATIVE AGLI UTENTI

Un nuovo utente, una volta raggiunto il sito web che ospita il social network, ha la possibilità di creare un profilo attraverso il processo di registrazione.

La registrazione prevede l'inserimento di alcuni dati personali:

- Nome, Cognome
- Data di nascita
- Luogo di nascita
- Sesso
- Nome utente
- Password

Dopo la registrazione, l'utente effettua l'accesso inserendo i soli Nome utente e Password, da cui è anche univocamente identificato.

### 1.2 FRASI RELATIVE AI PROFILI

Un profilo è lo strumento mediante il quale il singolo utente si interfaccia con il social network, accedendovi tramite le credenziali inserite al momento della registrazione (nome utente e password).

Poiché un utente e il suo profilo sono unici nel social network e legati tra loro tramite le credenziali di accesso, si decide di creare e mantenere solo l'entità UTENTE.

Il profilo mantiene informazioni sull'anagrafica, sugli interessi (musica, film, sport, libri) e sulla vita in generale (istruzione, scuole frequentate, impieghi attuali e passati); può contenere messaggi, immagini (anche organizzate in gallerie), file audio, file video, documenti elettronici di vario genere (testo, fogli di calcolo).

Si suppone che se l'utente cambi scuola a parità di grado di istruzione, il nome della scuola venga semplicemente sostituito con il nuovo.

Inizialmente un profilo non è collegato ad altri profili, quindi non sono presenti legami pensabili come amici o follower.

### 1.3 FRASI RELATIVE AL RAPPORTO FOLLOWER-FOLLOWING TRA UTENTI

Utilizzando il social network, l'utente può instaurare legami di amicizia con altri utenti. Considerando due utenti  $u_1$  e  $u_2$ , se  $u_1$  invia una richiesta ad  $u_2$  possono verificarsi i seguenti tre casi:

1.  $u_2$  accetta:  $u_1$  comparirà tra i followers di  $u_2$  e  $u_2$  tra i following di  $u_1$ ;
2.  $u_2$  non accetta: nei profili di entrambi gli utenti rimane visibile la richiesta pendente. Trascorsi trenta giorni dall'invio della richiesta, se questa non è stata ancora accettata, viene automaticamente rimossa e  $u_1$  non potrà inviare ulteriori richieste a  $u_2$  per i successivi sei mesi.
3.  $u_2$  accetta e ricambia: gli utenti compaiono tra i follower e i following l'uno dell'altro.

## 1.4 FRASI RELATIVE AI MESSAGGI

Gli utenti possono scambiarsi messaggi privati e/o pubblici secondo le seguenti regole:

1. se u1 è follower di u2, u1 può inviare messaggi pubblici a u2: questi compariranno sul profilo di u2 e saranno visibili da tutti i suoi followers. u1 non può inviare messaggi privati a u2;
2. se u1 e u2 sono uno follower dell'altro allora possono scambiarsi messaggi privati. In questo caso il messaggio sarà visibile solo a mittente e destinatario;
3. se u1 e u2 sono uno follower dell'altro, u1 può inviare messaggi pubblici a u2 decidendo a quali follower di quest'ultimo renderlo visibile. A u2 non è permesso modificare i follower scelti da u1.

I messaggi privati contengono solo testo.

I messaggi pubblici (post), possono contenere oltre al testo anche altri contenuti (vedi paragrafo successivo).

## 1.5 FRASI RELATIVE AI CONTENUTI

Oltre che con i messaggi, gli utenti possono interagire tra loro tramite contenuti di vario genere, eventi e sondaggi, oppure riunendosi in gruppi.

I possibili contenuti presenti sul profilo di un utente sono:

- Foto
- Messaggi pubblici
- Video
- Documenti elettronici
- Aggiornamenti sulle informazioni dell'utente
- Conferme di presenza a eventi

Le foto possono essere raccolte in gallerie, e si suppone che esista una galleria di default per ogni profilo. Se l'utente non specifica una galleria di destinazione per un'immagine, questa viene inserita nella galleria di default. Un'immagine può essere inserita in una sola galleria. Se, a seguito di cancellazioni, una galleria rimane vuota, essa viene rimossa.

I contenuti dipendentemente dalle regole di riservatezza impostate, rappresentano un oggetto di interazione tra follower (vedi Paragrafo 1.9).

Un utente può commentare contenuti, aggiungere un "like", inserire un tag o un hashtag in un contenuto pubblicato da lui stesso o da un follower/following:

- tramite un like, si esprime un apprezzamento verso un contenuto, proprio o di altri utenti. Una volta inserito, un like non può essere rimosso.
- tramite un tag, si aggiunge un collegamento tra un contenuto e un utente. Non è possibile taggare uno stesso utente più di una volta in uno stesso contenuto.
- tramite un hashtag, si associa un contenuto ad una parola chiave, che permette di categorizzarlo per argomento (es. *#parola\_chiave*). Un hashtag è caratterizzato da un tipo e da una parola chiave. Un utente può inserire un solo hashtag per contenuto.

## 1.6 FRASI RELATIVE AI GRUPPI

Un gruppo è un insieme di utenti del social network dove ogni membro può pubblicare messaggi, immagini, video, e altri contenuti. L'utente che decide di creare il gruppo ne diventa l'amministratore, che si suppone essere unico.

Un gruppo è caratterizzato da:

- un tema, che rappresenta l'obiettivo per cui è stato creato;

- un livello di visibilità (pubblica o ristretta) stabilito dall'amministratore;
- politiche di accesso ai contenuti dipendenti dal livello di visibilità:
  1. se la visibilità è pubblica chiunque può visualizzare i contenuti del gruppo
  2. se la visibilità è ristretta solo gli utenti che ricevono l'approvazione dell'amministratore possono visualizzare i contenuti del gruppo.

Gli utenti che hanno ricevuto l'approvazione dell'amministratore possono anche inviare messaggi al gruppo. Se un utente cancella la sua iscrizione o viene eliminato dall'amministratore, la sua attività non viene rimossa dal gruppo.

## 1.7 FRASI RELATIVE AGLI EVENTI

Gli utenti possono organizzare eventi speciali.

Un evento è organizzato da un singolo utente, che ne decide:

- Luogo
- Data
- Descrizione
- Invitati (utenti che potrebbero partecipare) tra i suoi follower e following
- Allestimento (insieme di oggetti e/o strutture necessarie allo svolgimento dell'evento)

Un invito è visibile solo al diretto interessato che può accettarlo o rifiutarlo:

1. se lo accetta può indicare una sua presenza sicura e definitiva o potenziale.
2. se lo rifiuta la decisione è definitiva.

Gli invitati possono proporre delle varianti per l'allestimento gestite tramite sondaggio (vedi paragrafo successivo).

## 1.8 FRASI RELATIVE A SONDAGGI

Un sondaggio è creato da uno e un solo utente che ne specifica:

- Argomento
- Persone coinvolte
- Possibili risposte

Un sondaggio è il mezzo attraverso il quale un insieme di utenti esprime la sua opinione riguardo un determinato argomento.

Sono possibili tre tipi di sondaggio:

1. All'interno di un gruppo;
2. Relativo ad un evento;
3. Riservato a una cerchia ristretta di follower (a questo tipo non sarà dedicata una relazione, in quanto tali sondaggi sono ottenibili per differenza da tutti i sondaggi eliminando quelli delle altre due categorie).

Un sondaggio può essere creato anche se nessuno vi parteciperà. Il creatore è l'unico partecipante obbligatorio.

## 1.9 FRASI RELATIVE ALLA RISERVATEZZA

Quando un utente accetta una richiesta di amicizia, dà la possibilità al nuovo amico di visualizzare tutti i contenuti del proprio profilo, a meno che non decida di nascondere alcuni.

Un utente può decidere di applicare regole di visibilità ai suoi contenuti, secondo vari livelli di granularità: di default un contenuto è visibile a tutti i suoi follower al momento dell'upload.

Le informazioni inserite riguardo a istruzione, impieghi, interessi o altro sono visibili a tutti i follower e i following di un utente. Le regole di visibilità si applicano ai soli contenuti.

## 1.10 FRASI RELATIVE AI BLOCCHI

Gli utenti del social network hanno la possibilità di bloccare uno o più follower.

Con una segnalazione, un utente avverte preventivamente un suo follower che quest'ultimo è a rischio di blocco, senza specificarne una ragione. La segnalazione impedisce al follower di visualizzare il profilo dell'utente infastidito per un mese, dopodiché tale profilo torna raggiungibile al follower. Nel caso di una seconda segnalazione, si attiva un blocco che impedisce al follower di visualizzare il profilo per un tempo indeterminato, fino a un eventuale sblocco.

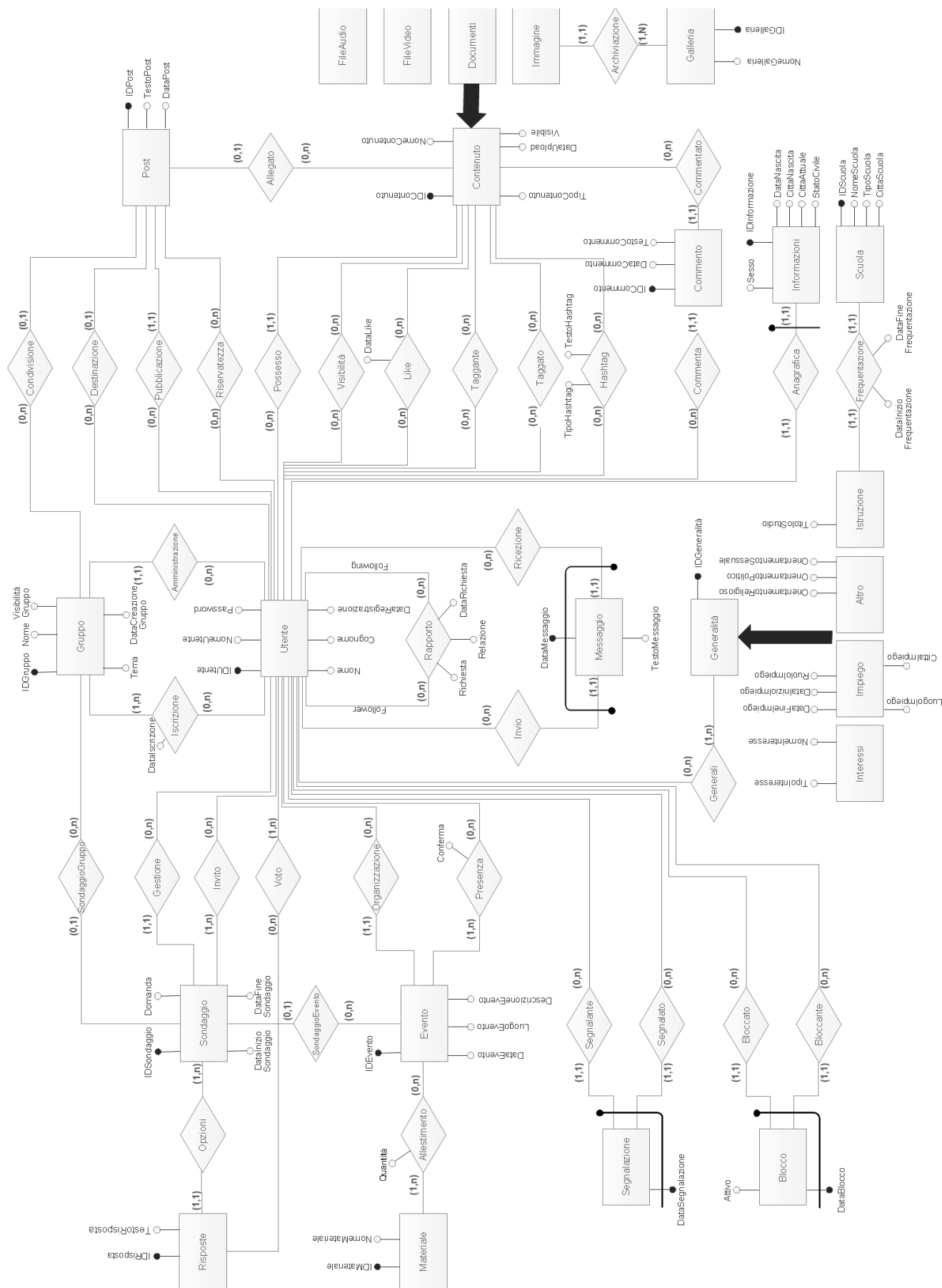
## 2.GLOSSARIO DEI TERMINI

| Termine               | Definizione  | Sinonimi                     | Collegamenti  |
|-----------------------|--|------------------------------|---|
| <b>Utente</b>         | Persona fisica che tramite la registrazione al Social Network può usufruire delle sue funzionalità.                                    | Profilo, Bacheca             | Gruppi, eventi, sondaggi, messaggi, contenuti, generalità (interessi, istruzione, anagrafica, impiego ecc...) |
| <b>Profilo</b>        | Area del Social Network dedicata al singolo utente: contiene i dati che l'utente vuole rendere pubblici.                               | Bacheca, Utente              | Utenti, informazioni, post, visibilità  |
| <b>Registrazione</b>  | Procedura necessaria per diventare un utente del Social Network  | Iscrizione al social network | Utenti  |
| <b>Iscrizione</b>     | Procedura necessaria per diventare membro di un gruppo   |                              | Utenti, gruppi  |
| <b>Invito</b>         | Invito che un utente riceve dal creatore del sondaggio per parteciparvi  |                              | Utenti, sondaggi  |
| <b>Partecipazione</b> | Partecipazione a un sondaggio  | Scelta di una risposta       | Utenti, sondaggi  |
| <b>Presenza</b>       | Presenza a un evento   | Partecipazione all'evento    | Utenti, eventi  |
| <b>Follower</b>       | Dato un utente, un suo follower è quell'utente che decide di seguire le attività sul Social Network del primo                          | Amico                        | Utenti  |
| <b>Following</b>      | Dato un utente, i suoi following sono gli utenti che segue   | Amico                        | Utenti  |
| <b>Post</b>           | Un post è un messaggio pubblico e può contenere testo e contenuti allegati   | Messaggio pubblico           | Utenti, gruppi, contenuti, visibilità   |
| <b>Riservatezza</b>   | Specifica gli utenti che possono interagire con i post pubblicati da un altro utente ai quali quest'ultimo ha modificato la visibilità | Visibile                     | Utente, Post  |
| <b>Like</b>           | Espressione di apprezzamento verso un contenuto, proprio o di altri utenti   | "Mi piace"                   | Utenti, contenuti   |
| <b>Tag</b>            | Collegamento tra un contenuto e un utente  |                              | Utenti, contenuti   |



|                       |   |                          |   |
|-----------------------|---|--------------------------|---|
| <b>Hashtag</b>        | Parola chiave che categorizza i contenuti per argomento. E' caratterizzato da un tipo e da un nome. | Parola chiave            | Contenuti                                   |
| <b>Amministratore</b> | Utente che gestisce un gruppo   | Creatore del gruppo      | Utenti, gruppi                              |
| <b>Gestore</b>        | Utente che gestisce un sondaggio  | Creatore del sondaggio   | Utenti, sondaggi                            |
| <b>Organizzatore</b>  | Utente che gestisce un evento   | Creatore dell'evento     | Utenti, eventi                              |
| <b>Segnalazione</b>   | Procedura che impedisce per 30 giorni l'interazione tra due utenti                                  |                          | Utenti, blocco                              |
| <b>Blocco</b>         | Procedura che impedisce definitivamente l'interazione tra due utenti                                |                          | Utenti, segnalazione                        |
| <b>Pubblicazione</b>  | Specifica i post pubblicati da un singolo utente  |                          | Utenti, post, contenuti, visibilità         |
| <b>Condivisione</b>   | Specifica i post pubblicati all'interno di un gruppo  | Pubblicazione in gruppo  | Utenti, post, contenuti, gruppi, visibilità |
| <b>Destinatario</b>   | Specifica i post pubblicati sulla bacheca di un utente  | Pubblicazione in bacheca | Utenti, post, contenuti, visibilità         |
| <b>Grado</b>          | Specifica il grado di istruzione del singolo utente   | Livello di istruzione    | Utenti, istruzione                          |
| <b>Immagine</b>       | Contenuto di tipo immagine  | Foto                     | Contenuti, Galleria                         |

### 3.SCHEMA INIZIALE



*Figura 1 - Diagramma ER con generalizzazioni*

# II. PROGETTAZIONE LOGICA

## 4.RISTRUTTURAZIONE

Dal precedente diagramma ER non risultano attributi multivalore.

Si assume che le generalizzazioni GENERALI e CONTENUTO siano esclusive: non è quindi possibile inserire informazioni non previste dal social network o interagire con i contenuti in altri modi non esplicitamente implementati.

Si è deciso di:

- Eliminare la generalizzazione relativa a CONTENUTI (Figura 2), accorpendo i figli al genitore. Le entità figlie vengono eliminate e le loro proprietà (attributi e partecipazione alle associazioni) vengono aggiunte al genitore. A quest'ultimo viene inoltre aggiunto un ulteriore attributo (*TipoContenuto*) che serve a distinguerne le occorrenze e capire a quale delle entità figlie si riferiva.
- Eliminare la generalizzazione relativa a GENERALITA' (Figura 3), accorpendo il genitore ai figli: per la proprietà di ereditarietà, l'entità genitore viene eliminata e i suoi attributi vengono aggiunti a tutte le entità figlie.

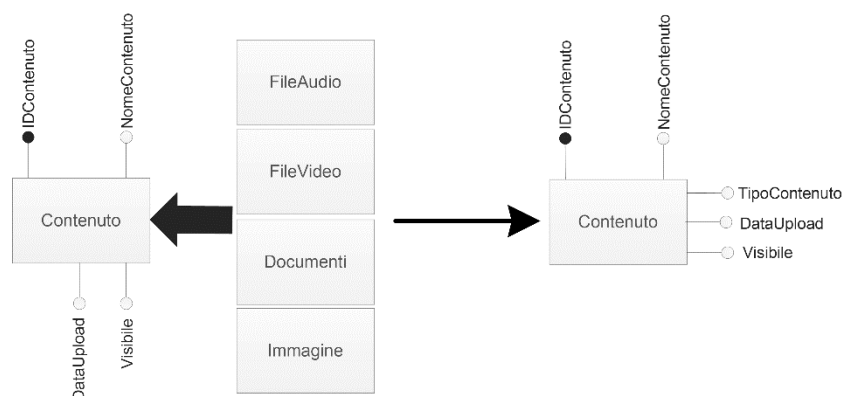


Figura 2 - Generalizzazione CONTENUTO prima e dopo la ristrutturazione

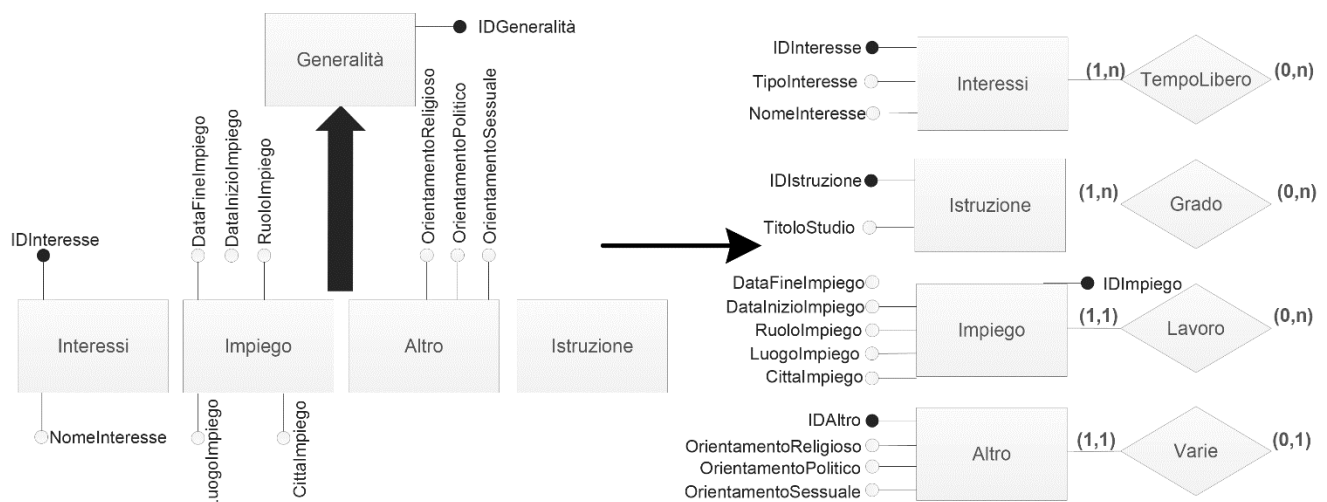


Figura 3 - Generalizzazione GENERALITA' prima e dopo la ristrutturazione

# 5.SCHEMA RISTRUTTURATO

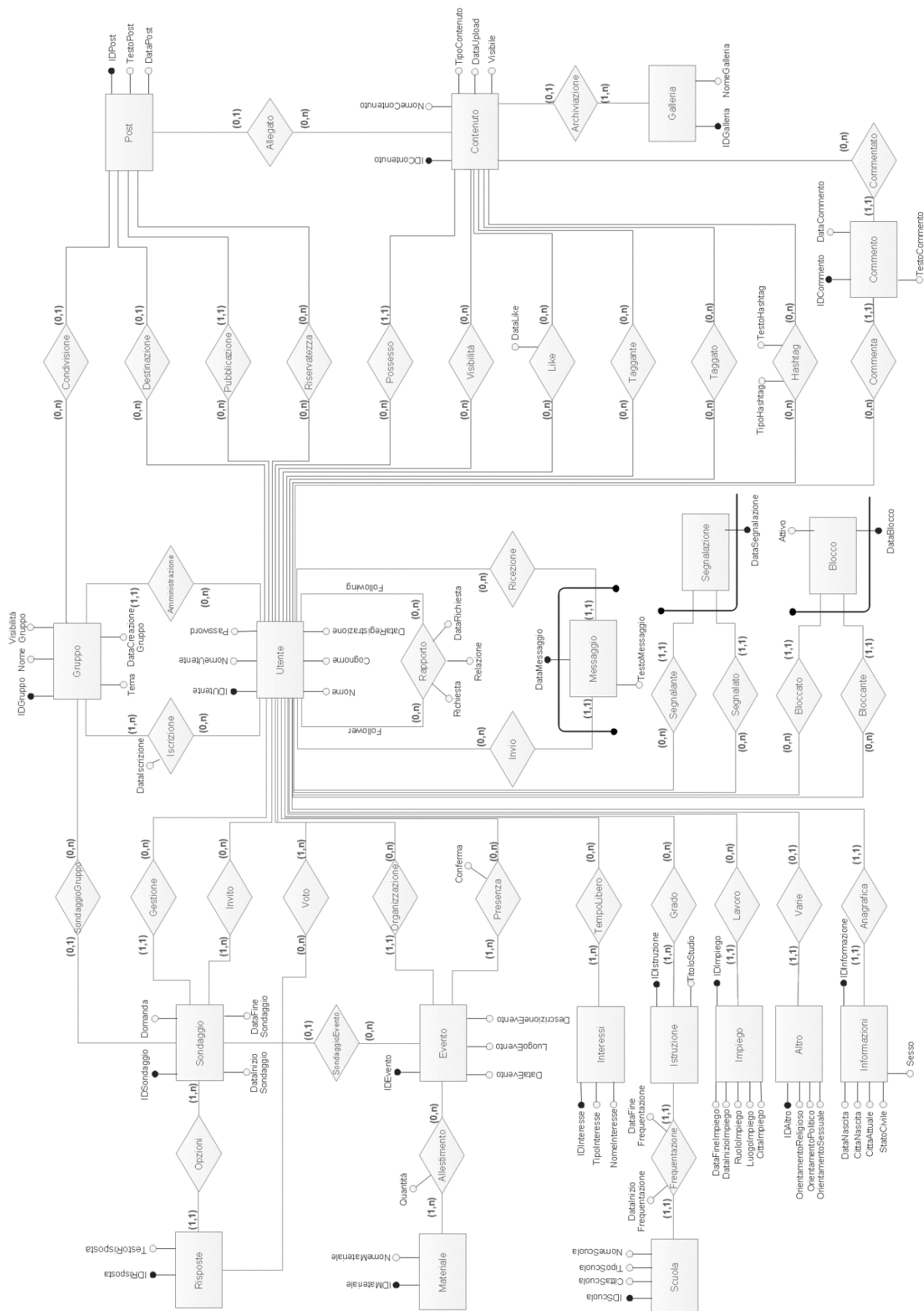


Figura 4 – Diagramma ER senza generalizzazioni

## 6.TABELLA DELLE ENTITA'

| Entità              | Descrizione   | Attributi   | Identificatore                                    |
|---------------------|---|---|---|
| <b>Utente</b>       | Persona fisica che tramite la registrazione al Social Network può usufruire delle sue funzionalità. | IDUtente, NomeUtente, Password, Nome, Cognome, DataRegistrazione                        | IDUtente  |
| <b>Gruppo</b>       | Insieme di utenti, riuniti per interessi comuni.  | IDGruppo, Tema, Nome, DataCreazioneGruppo, VisibilitàGruppo                             | IDGruppo  |
| <b>Messaggio</b>    | Messaggio di testo, scambiato esclusivamente tra due utenti (messaggio privato)                     | TestoMessaggio, DataMessaggio   | Esterno (Mittente, Destinatario), DataMessaggio   |
| <b>Segnalazione</b> | Procedura che impedisce per 30 giorni l'interazione tra due utenti                                  | DataSegnalazione  | Esterno (Segnalante, Segnalato), DataSegnalazione |
| <b>Blocco</b>       | Procedura che impedisce definitivamente l'interazione tra due utenti                                | DataBlocco, Attivo  | Esterno (Bloccante, Bloccato), DataBlocco         |
| <b>Sondaggio</b>    | Metodo per valutare proposte di vario genere  | IDSondaggio, DataInizioSondaggio, DataFineSondaggio, Domanda                            | IDSondaggio                                       |
| <b>Risposte</b>     | Possibili opzioni tra cui scegliere in un sondaggio   | IDRisposta, TestoRisposta   | IDRisposta  |
| <b>Evento</b>       | Evento organizzato sul Social Network   | IDEvento, DataEvento, LuogoEvento, DescrizioneEvento                                    | IDEvento  |
| <b>Materiale</b>    | Oggetto e/o persona che può essere necessario all'organizzazione dell'evento                        | IDMateriale, NomeMateriale  | IDMateriale                                       |
| <b>Informazioni</b> | Informazioni basilari sull'utente   | IDInformazione, DataNascita, CittàNascita, CittàAttuale, StatoCivile, Sesso             | IDInformazione                                    |
| <b>Interesse</b>    | Passatempi con cui l'utente si diletta nella vita fuori dal Social Network                          | IDInteresse, TipoInteresse, NomeInteresse   | IDInteresse                                       |
| <b>Impiego</b>      | Attività lavorativa dell'utente   | IDImpiego, CittàImpiego, LuogoImpiego, RuoloImpiego, DataInizioImpiego, DataFineImpiego | IDImpiego   |
| <b>Istruzione</b>   | Informazioni sul grado di istruzione  | IDIstruzione, Scuola, TitoloStudio  | IDIstruzione                                      |
| <b>Scuola</b>       | Informazioni sull'istituto scolastico frequentato   | IDScuola, TipoScuola, NomeScuola, CittàScuola   | IDScuola  |

|                  |                                       |  |             |
|------------------|---------------------------------------|--|-------------|
| <b>Altro</b>     | Informazioni aggiuntive               | IDAltro, OrientamentoPolitico, OrientamentoReligioso, OrientamentoSessuale | IDAltro     |
| <b>Post</b>      | Messaggi pubblici                     | IDPost, TestoPost, DataPost  | IDPost      |
| <b>Contenuto</b> | File caricati sul social network      | IDContenuto, NomeContenuto, DataUpload, Visibile, TipoContenuto            | IDContenuto |
| <b>Commento</b>  | Commenti dei vari utenti ai contenuti | IDCommento, TestoCommento, DataCommento                                    | IDCommento  |
| <b>Galleria</b>  | Insieme di immagini                   | IDGalleria, NomeGalleria   | IDGalleria  |

## 7.TABELLA DELLE ASSOCIAZIONI

| Associazione           | Descrizione  | Componenti           | Attributi                           | Cardinalità   |  |
|------------------------|--|----------------------|-------------------------------------|---|--|
| <b>Rapporto</b>        | Specifica i legami di amicizia tra utenti                          | Utente, Utente       | Richiesta, DataRichiesta, Relazione | (0, N)<br>Un utente può non seguire altri utenti o seguirne molti                 | (0, N)<br>Un utente può non essere seguito da alcun utente o essere seguito da molti |
| <b>Iscrizione</b>      | Procedura attraverso cui l'utente si iscrive a un gruppo           | Utente, Gruppo       | DataIscrizione                      | (0, N)<br>Un utente può non iscriversi a nessun gruppo o iscriversi a più gruppi  | (1, N)<br>Un gruppo ha almeno l'amministratore come membro                           |
| <b>Amministrazione</b> | Specifica l'amministratore (unico) del gruppo                      | Utente, Gruppo       |                                     | (0, N)<br>Un utente può non amministrare nessun gruppo o amministrarne più di uno | (1, 1)<br>Un gruppo è amministrato da uno e un solo utente                           |
| <b>Invio</b>           | Specifica i messaggi privati inviati da un utente                  | Utente, Messaggio    |                                     | (0, N)<br>Un utente può non inviare nessun messaggio o inviarne molti             | (1, 1)<br>Un messaggio esiste se inviato da un utente                                |
| <b>Ricezione</b>       | Specifica i messaggi privati ricevuti da un utente                 | Utente, Messaggio    |                                     | (0, N)<br>Un utente può non ricevere nessun messaggio o riceverne molti           | (1, 1)<br>Un messaggio esiste se ricevuto da un utente                               |
| <b>Segnalante</b>      | Specifica l'utente che effettua la segnalazione di un altro utente | Utente, Segnalazione |                                     | (0, N)<br>Un utente può non segnalare alcun utente o segnalarne molti             | (1,1)<br>Segnalazioni effettuate in tempi diversi sono diverse                       |
| <b>Segnalato</b>       | Specifica l'utente a cui è stata fatta la segnalazione             | Utente, Segnalazione |                                     | (0, N)<br>Un utente può non essere segnalato da nessun utente o da molti          | (1, 1)<br>Segnalazioni ricevute in tempi diversi dallo stesso utente sono diverse    |
| <b>Bloccante</b>       | Specifica l'utente che effettua il blocco di un altro utente       | Utente, Blocco       |                                     | (0, N)<br>Un utente può non bloccare alcun utente o bloccarne molti               | (1,1)<br>Blocchi effettuati in date diverse allo stesso utente sono diversi          |
| <b>Bloccato</b>        | Specifica l'utente bloccato  | Utente, Blocco       |                                     | (0, N)  | (1, 1)<br>Blocchi ricevuti in date diverse dallo stesso                              |

|                         |  |                     |          |  |  |
|-------------------------|--|---------------------|----------|--|--|
|                         |  |                     |          | Un utente può non essere bloccato da nessun utente o da molti                                      | utente sono diversi  |
| <b>Gestione</b>         | Specifica l'utente che indice un sondaggio   | Utente, Sondaggio   |          | (0, N)<br>Un utente può non indire alcun sondaggio o indirne molti                                 | (1, 1)<br>Un sondaggio può essere indetto da un solo utente                      |
| <b>Invito</b>           | Specifica gli utenti invitati a partecipare a un sondaggio                               | Utente, Sondaggio   |          | (0, N)<br>Un utente può non essere invitato a partecipare a nessun sondaggio o partecipare a molti | (1, N)<br>Un sondaggio ha almeno il gestore come partecipante                    |
| <b>Opzioni</b>          | Specifica le risposte di un sondaggio tra le quali gli utenti invitati possono scegliere | Sondaggio, Risposte |          | (1, N)<br>Un sondaggio deve prevedere almeno una risposta  | (1, 1)<br>Una risposta è opzione di uno e un solo sondaggio                      |
| <b>Voto</b>             | Specifica le preferenze degli utenti che hanno deciso di partecipare al sondaggio        | Risposte, Utente    |          | (1, N)<br>Un utente, se partecipa ad almeno un sondaggio, deve votare almeno una risposta          | (0, N)<br>Una risposta può non essere votata da nessun utente o da molti         |
| <b>Sondaggio-Gruppo</b> | Specifica i sondaggi che vengono indetti all'interno di un gruppo                        | Sondaggio, Gruppo   |          | (0, 1)<br>Un sondaggio può essere presente al più una volta all'interno di un gruppo               | (0, N)<br>Un gruppo può non contenere alcun sondaggio o contenerne molti         |
| <b>Organizzazione</b>   | Specifica l'utente che organizza un evento   | Utente, Evento      |          | (0, N)<br>Un utente può non organizzare nessun evento o organizzarne molti                         | (1, 1)<br>Un evento deve essere organizzato da uno e un solo utente              |
| <b>Presenza</b>         | Specifica la presenza di utente a un dato evento   | Utente, Evento      | Conferma | (0, N)<br>Perché un utente può non partecipare a nessun evento, o partecipare a molti              | (1, N)<br>Un evento ha tra i suoi partecipanti almeno il suo organizzatore       |
| <b>Allestimento</b>     | Specifica l'insieme dei materiali necessari all'organizzazione dell'evento               | Evento, Materiali   | Quantità | (0, N)<br>Un evento può non richiedere alcun materiale o richiederne molti                         | (1, N)<br>Un'istanza di materiale esiste se necessaria in almeno un allestimento |



|                        |   |                      |  |   |  |
|------------------------|---|----------------------|--|---|--|
| <b>SondaggioEvento</b> | Specifica i sondaggi che vengono indetti all'interno di un evento   | Sondaggio, Evento    |  | (0, 1)<br>Un sondaggio può essere presente al più una volta all'interno di un evento        | (0, N)<br>Un evento può non contenere alcun sondaggio o contenerne molti                     |
| <b>Anagrafica</b>      | Specifica le informazioni anagrafiche del singolo utente            | Utente, Informazioni |  | (1, 1)<br>Ogni utente ha un singolo set di informazioni anagrafiche                         | (1, 1)<br>Ogni set di informazioni anagrafiche è relativo a uno e un solo utente             |
| <b>TempoLibero</b>     | Specifica gli interessi di un singolo utente                        | Utente, Interesse    |  | (0, N)<br>Un utente può non inserire interessi o inserirne molti                            | (1, N)<br>Un interesse appartiene ad almeno un utente, ma può essere comune a molti          |
| <b>Grado</b>           | Specifica il grado di istruzione di un singolo utente               | Utente, Istruzione   |  | (0, N)<br>Un utente può non inserire alcun titolo di studio o inserirne molti               | (1, N)<br>Un grado di istruzione appartiene ad almeno un utente ma può essere comune a molti |
| <b>Frequenzazione</b>  | Specifica le scuole frequentate dal singolo utente                  | Istruzione, Scuola   | DataInizioFrequenzazione, DataFineFrequenzazione | (1, 1)<br>Ogni grado di istruzione dell'utente è legato a una e una sola scuola             | (1, 1)<br>Una scuola è univocamente associata al grado di istruzione di un utente            |
| <b>Lavoro</b>          | Specifica l'impiego di un utente                                    | Utente, Impiego      |  | (0, N)<br>Un utente può non inserire alcun lavoro o inserirne molti                         | (1, 1)<br>Un impiego è univocamente associato ad un utente                                   |
| <b>Varie</b>           | Specifica informazioni aggiuntive sull'utente                       | Utente, Altro        |  | (0, 1)<br>Un utente può non inserire un set di informazioni aggiuntive o inserirne uno solo | (1, 1)<br>Ogni set di informazioni aggiuntive è relativo a un solo utente                    |
| <b>Pubblicazione</b>   | Specifica i post pubblicati da un singolo utente                    | Utente, Post         |  | (0, N)<br>Un utente può non pubblicare alcun post o pubblicarne molti                       | (1, 1)<br>Un post può essere pubblicato da un solo utente                                    |
| <b>Destinazione</b>    | Specifica i post pubblicati da un utente sulla bacheca di un utente | Utente, Post         |  | (0, N)<br>Un utente può non ricevere sulla  | (0, 1)<br>Un post può essere pubblicato sulla bacheca di                                     |

|                     |  |                   |          |  |   |
|---------------------|--|-------------------|----------|--|---|
|                     |  |                   |          | sua bacheca alcun post o riceverne molti   | un utente al più una volta  |
| <b>Riservatezza</b> | Specifica i post a cui un utente ha modificato la visibilità                       | Utente, Post      |          | (0, N)<br>Un utente può non visualizzare alcun post o visualizzarne molti                                | (0, N)<br>Un post può non essere visibile ad alcun utente o esserlo a molti                         |
| <b>Condivisione</b> | Specifica i post pubblicati da un utente all'interno di un gruppo                  | Gruppo, Post      |          | (0, N)<br>Un gruppo può non contenere alcun post o contenerne molti                                      | (0, 1)<br>Un post può essere pubblicato in un gruppo al più una volta                               |
| <b>Allegato</b>     | Specifica il contenuto allegato ad un singolo post                                 | Post, Contenuto   |          | (0, 1)<br>Un post può non avere allegati o averne uno solo   | (0, N)<br>Un contenuto può non essere allegato in nessun post o esserlo in molti                    |
| <b>Possesso</b>     | Specifica i contenuti appartenenti al singolo utente                               | Utente, Contenuto |          | (0, N)<br>Un utente può non caricare nessun contenuto o caricarne molti                                  | (1, 1)<br>Un contenuto caricato appartiene a uno e un solo utente                                   |
| <b>Visibilità</b>   | Specifica i contenuti per i quali un utente ha impostato una visibilità intermedia | Utente, Contenuto |          | (0, N)<br>Un utente può scegliere di modificare o meno la politica di visibilità default di un contenuto | (0, N)<br>Un contenuto può non avere visibilità intermedia, o essere visibile solo da alcuni utenti |
| <b>Like</b>         | Specifica gli utenti che hanno espresso un apprezzamento verso un contenuto        | Utente, Contenuto | DataLike | (0, N)<br>Un utente può non apprezzare nessun contenuto o apprezzarne molti                              | (0, N)<br>Un contenuto può non essere apprezzato da nessun utente o esserlo da molti                |
| <b>Taggante</b>     | Specifica l'utente che tagga un altro utente in un contenuto                       | Utente, Contenuto |          | (0, N)<br>Un utente può scegliere di non taggare altri utenti in un contenuto o taggarne molti           | (0, N)<br>Un contenuto può non avere alcun tag o averne molti                                       |
| <b>Taggato</b>      | Specifica l'utente che viene taggato da un altro utente in un contenuto            | Utente, Contenuto |          | (0, N)<br>Un utente può non essere taggato da altri utenti in un contenuto o essere taggato in molti     | (0, N)<br>Un contenuto può non avere alcun tag o averne molti                                       |

|                      |  |                     |                           |   |  |
|----------------------|--|---------------------|---------------------------|---|--|
| <b>Hashtag</b>       | Specifica le parole chiave assegnate a un contenuto dal proprietario o da altri utenti | Utente, Contenuto   | TipoHashtag, TestoHashtag | (0, N)<br>Un utente può non assegnare alcun hashtag a un contenuto o assegnarne a molti                                       | (0, N)<br>Un contenuto può non avere hashtag o averne molti                    |
| <b>Commenta</b>      | Specifica i commenti effettuati dagli utenti   | Utente, Commento    |                           | (0, N)<br>Un utente può non effettuare alcun commento o effettuarne molti   | (1, 1)<br>Un commento appartiene a un solo utente                              |
| <b>Commentato</b>    | Specifica i contenuti che hanno ricevuto commenti                                      | Commento, Contenuto |                           | (1, 1)<br>Un commento è associato a uno e un solo contenuto   | (0, N)<br>Un contenuto può non ricevere alcun commento o riceverne molti       |
| <b>Archiviazione</b> | Specifica le immagini catalogate in una galleria                                       | Contenuto, Galleria |                           | (0, 1)<br>Un contenuto può essere catalogato in gallerie solo se di tipo immagine. In tal caso sarà catalogato una sola volta | (1, N)<br>Una galleria esiste se contiene almeno un contenuto di tipo immagine |

## 8.REGOLE AZIENDALI

Per il codice MYSQL dei vincoli di integrità generici 1, 10 e 17 si rimanda al Capitolo 16.

| Regole di vincolo |  |
|-------------------|--|
| <b>RV1</b>        | Non devono esistere due utenti con lo stesso NomeUtente  |
| <b>RV2</b>        | Due utenti tra i quali non esiste alcuna relazione non devono scambiarsi messaggi né pubblici né privati                                   |
| <b>RV3</b>        | Un follower non deve inviare messaggi privati all'utente che segue, a meno che non sia seguito a sua volta da esso                         |
| <b>RV4</b>        | Una volta scaduta una richiesta di interesse pendente un utente non deve inviarne ulteriori allo stesso utente per i successivi sei mesi   |
| <b>RV5</b>        | Il gruppo deve avere un solo utente come amministratore  |
| <b>RV6</b>        | Un utente che non ha ricevuto l'approvazione dell'amministratore di un gruppo non può inviare messaggi né condividere post sul gruppo      |
| <b>RV7</b>        | Post condivisi in un gruppo non devono essere visibili da utenti che non sono iscritti a tale gruppo se la visibilità del gruppo è privata |
| <b>RV8</b>        | Un utente non deve essere destinatario di un post per il quale l'autore del post non ne abbia permesso la visualizzazione                  |
| <b>RV9</b>        | Un utente segnalato non può interagire con il segnalante per i 30 giorni successivi alla segnalazione                                      |
| <b>RV10</b>       | Un utente non deve essere bloccato da un altro utente prima che quest'ultimo l'abbia segnalato due volte                                   |
| <b>RV11</b>       | Un utente bloccato non può interagire o visualizzare i contenuti del segnalante fino a che quest'ultimo non rimuova il blocco              |
| <b>RV12</b>       | Un utente non deve interagire con contenuti che non appartengano a suoi follower/following   |
| <b>RV13</b>       | Un evento deve avere un solo utente come creatore  |
| <b>RV14</b>       | L'organizzatore di un evento non deve invitare utenti che non siano tra i suoi follower/following  |
| <b>RV15</b>       | Un sondaggio deve avere un solo utente come creatore che deve inserire la domanda, i partecipanti e le possibili risposte                  |
| <b>RV16</b>       | Un utente invitato al sondaggio non deve aggiungere o modificare le risposte imposte dal creatore del sondaggio al momento della creazione |
| <b>RV17</b>       | Un utente che non partecipa ad un evento non deve proporre un sondaggio in tale evento   |
| <b>RV18</b>       | Un utente che non è iscritto ad un gruppo non deve proporre un sondaggio in tale gruppo  |
| <b>RV19</b>       | Un utente non invitato a un sondaggio non deve parteciparvi  |
| <b>RV20</b>       | Un utente non deve inserire più di un hashtag in un contenuto  |
| <b>RV21</b>       | La modifica della partecipazione a un evento non deve essere effettuata se l'evento è già concluso   |
| <b>RV22</b>       | Una galleria deve essere cancellata se non contiene almeno una immagine.   |

## 9.ANALISI DELLE PRESTAZIONI

### 9.1 TAVOLA DEI VOLUMI

Si assume che il database mantenga le informazioni relative agli ultimi 5 anni di attività del social network.

| Concetto        | Tipo | Volume     | Motivazione   |
|-----------------|------|------------|---|
| Utente          | E    | 5.000      | Ipotesi iniziale  |
| Gruppo          | E    | 50.000     | Un utente crea in media due gruppi all'anno   |
| Messaggio       | E    | 90.000.000 | Un utente invia in media 10 messaggi al giorno ( $5.000 \times 10 \times 365 \times 5 = 91.250.000$ )   |
| Segnalazione    | E    | 50.000     | Un utente segnala in media due persone all'anno   |
| Blocco          | E    | 20.000     | Tra gli utenti segnalati, il 40% viene bloccato definitivamente   |
| Sondaggio       | E    | 75.000     | Un utente crea in media 3 sondaggi all'anno   |
| Risposte        | E    | 300.000    | Ogni sondaggio offre in media 4 risposte tra cui scegliere  |
| Evento          | E    | 50.000     | Un utente organizza in media 2 eventi all'anno  |
| Materiale       | E    | 200.000    | Ogni evento richiede in media 4 elementi nel suo allestimento, ma il 50% di essi è ripetuto per vari eventi ( $100.000 \times 4 \times 0.5 = 200.000$ ) |
| Informazioni    | E    | 5.000      | La relazione ANAGRAFICA ha cardinalità (1, 1) ambo le parti   |
| Interesse       | E    | 4.500      | Ogni utente ha in media 3 interessi, ma il 30% di essi è ripetuto per vari utenti ( $3 \times 5.000 \times 0.3 = 4.500$ )                               |
| Impiego         | E    | 5.000      | Un utente inserisce in media un impiego   |
| Istruzione      | E    | 15.000     | Un utente inserisce in media tre gradi di istruzione  |
| Scuola          | E    | 15.000     | La relazione FREQUENTAZIONE ha cardinalità (1, 1) ambo le parti   |
| Altro           | E    | 4.000      | Non tutti gli utenti inseriscono informazioni aggiuntive  |
| Post            | E    | 36.000.000 | Un utente pubblica in media 4 messaggi al giorno ( $5.000 \times 4 \times 365 \times 5 = 36.500.000$ )  |
| Contenuto       | E    | 45.000.000 | Un utente pubblica in media 5 contenuti al giorno ( $5.000 \times 5 \times 365 \times 5 = 45.625.000$ )   |
| Galleria        | E    | 150.000    | Oltre alle gallerie di default, un utente crea in media 6 gallerie all'anno ( $5.000 \times 6 \times 5 + 5.000 = 155.000$ )                             |
| Commento        | E    | 91.000.000 | Un utente commenta in media dieci volte al giorno ( $5.000 \times 10 \times 365 \times 5 = 91.250.000$ )  |
| Iscrizione      | R    | 75.000     | Un utente si iscrive in media a 3 gruppi all'anno   |
| Amministrazione | R    | 50.000     | La relazione ha cardinalità (1, 1) con GRUPPO   |
| Segnalante      | R    | 50.000     | La relazione ha cardinalità (1, 1) con SEGNALAZIONE   |
| Segnalato       | R    | 50.000     | La relazione ha cardinalità (1, 1) con SEGNALAZIONE   |
| Bloccante       | R    | 20.000     | La relazione ha cardinalità (1, 1) con BLOCCO   |
| Bloccato        | R    | 20.000     | La relazione ha cardinalità (1, 1) con BLOCCO   |
| Persistenza     | R    | 20.000     | La relazione ha cardinalità (1, 1) con BLOCCO   |
| Gestione        | R    | 75.000     | La relazione ha cardinalità (1, 1) con SONDAGGIO  |
| SondaggioGruppo | R    | 22.500     | I sondaggi indetti all'interno di gruppi sono il 30% del totale   |
| SondaggioEvento | R    | 37.500     | I sondaggi relativi a eventi sono il 50% del totale   |
| Invito          | R    | 125.000    | Un utente viene invitato in media a 5 sondaggi all'anno tra privati, in eventi o in gruppi.   |
| Opzioni         | R    | 300.000    | La relazione ha cardinalità (1, 1) con RISPOSTE   |
| Voto            | R    | 112.500    | Il 90% degli invitati partecipa attivamente al sondaggio  |

|                       |   |               |   |
|-----------------------|---|---------------|---|
| <b>Organizzazione</b> | R | 100.000       | La relazione ha cardinalità (1, 1) con EVENTO   |
| <b>Presenza</b>       | R | 250.000       | Un utente partecipa in media a 10 eventi all'anno ( $5.000 \times 10 \times 5 = 250.000$ )  |
| <b>Allestimento</b>   | R | 400.000       | Un evento richiede in media 4 materiali nel suo allestimento  |
| <b>Rapporto</b>       | R | 1.500.000     | Ogni mese un utente chiede amicizia in media a tre utenti e riceve una richiesta da due utenti ( $12 \times 5 \times (5.000 \times 3 + 5.000 \times 2) = 1.500.000$ ).  |
| <b>Invio</b>          | R | 90.000.000    | La relazione ha cardinalità (1, 1) con MESSAGGIO  |
| <b>Ricezione</b>      | R | 90.000.000    | La relazione ha cardinalità (1, 1) con MESSAGGIO  |
| <b>Possesso</b>       | R | 45.000.000    | La relazione ha cardinalità (1, 1) con CONTENUTO  |
| <b>Visibilità</b>     | R | 1.300.000.000 | Un utente modifica in media la visibilità del 60% dei suoi contenuti per il 40% dei suoi follower ( $5.000 \times ((45.000.000 / 5.000 \times 0.6) \times (12 \times 5 \times 2 \times 0.4)) = 1.296.000.000$ ) |
| <b>Like</b>           | R | 36.500.000    | Un utente mette in media 4 like al giorno ( $5.000 \times 4 \times 365 \times 5 = 36.500.000$ )   |
| <b>Taggante</b>       | R | 18.000.000    | Un utente mette in media 2 tag al giorno ( $5.000 \times 2 \times 365 \times 5 = 18.250.000$ )  |
| <b>Taggato</b>        | R | 18.000.000    | Un utente riceve in media 2 tag al giorno ( $5.000 \times 2 \times 365 \times 5 = 18.250.000$ )   |
| <b>Hashtag</b>        | R | 27.000.000    | Un utente mette in media 3 hashtag al giorno ( $5.000 \times 3 \times 365 \times 5 = 27.375.000$ )  |
| <b>Commenta</b>       | R | 91.000.000    | La relazione ha cardinalità (1, 1) con COMMENTO   |
| <b>Commentato</b>     | R | 91.000.000    | La relazione ha cardinalità (1, 1) con COMMENTO   |
| <b>Archiviazione</b>  | R | 22.500.000    | Il 50% dei contenuti sono di tipo immagine  |
| <b>Pubblicazione</b>  | R | 36.000.000    | La relazione ha cardinalità (1, 1) con POST   |
| <b>Destinazione</b>   | R | 25.200.000    | Il 70% dei POST è pubblicato sul profilo di un utente   |
| <b>Riservatezza</b>   | R | 548.400.000   | Un utente modifica in media la visibilità del 40% dei suoi post per il 30% dei suoi follower ( $5.000 \times ((36.000.000 / 5.000 \times 0.4) \times (12 \times 5 \times 2 \times 0.3)) = 518.400.000$ )        |
| <b>Condivisione</b>   | R | 10.800.000    | Il 30% dei POST è condiviso all'interno di un gruppo  |
| <b>Allegato</b>       | R | 14.400.000    | Il 40% dei POST include degli allegati  |
| <b>Anagrafica</b>     | R | 5.000         | La relazione ha cardinalità (1, 1) con UTENTE   |
| <b>TempoLibero</b>    | R | 15.000        | Un utente ha in media 3 interessi   |
| <b>Grado</b>          | R | 3.000         | Il 60% degli utenti inserisce informazioni sul titolo di studio   |
| <b>Frequentazione</b> | R | 15.000        | La relazione ha cardinalità (1, 1) con ISTRUZIONE   |
| <b>Lavoro</b>         | R | 5.000         | La relazione ha cardinalità (1, 1) con IMPIEGO  |
| <b>Varie</b>          | R | 4.000         | La relazione ha cardinalità (1, 1) con ALTRO  |

## 9.2 OPERAZIONI

Sono state implementate le seguenti operazioni:

| Tavola delle operazioni |   |                  |
|-------------------------|---|------------------|
| Operazione              | Descrizione   | Frequenza        |
| 1                       | Visualizzare l'esito di un sondaggio  | 50 al giorno     |
| 2                       | Visualizzare quanti utenti partecipano, forse partecipano o non partecipano a un evento | 100 al giorno    |
| 3                       | Visualizzare il contenuto più popolare di un utente, in base al numero di like ricevuti | 500 al giorno    |
| 4                       | Eliminare un'immagine da una galleria   | 20 al giorno     |
| 5                       | Inserire un like in un contenuto  | 20.000 al giorno |
| 6                       | Modifica di una partecipazione potenziale a un evento in partecipazione sicura          | 10 al giorno     |
| 7                       | Visualizzare il numero di follower e following di un utente                             | 800 al giorno    |
| 8                       | Segnalare un utente   | 5 al giorno      |

Per il codice MYSQL delle operazioni, si rimanda al Capitolo 15.

## 9.3 TAVOLE DEGLI ACCESSI

Si studiano le tavole degli accessi delle operazioni descritte nel paragrafo precedente.

### 9.3.1 Operazione 1

Si assume di conoscere il sondaggio di cui si vuole sapere l'esito

| N° operazioni                      | Tipo   | Motivazione  |
|------------------------------------|--|--|
| <b>300.000</b>                     | Lettura  | Compio una scansione in OPZIONI per selezionare le risposte del sondaggio desiderato                                     |
| <b>112.500</b>                     | Lettura  | Compio una scansione in VOTO per selezionare il numero di voti ottenuti da ciascuna risposta trovata al passo precedente |
| <b>300.000 + 112.500 = 412.500</b> | Totale operazioni elementari per la singola operazione |  |
| <b>412.500 x 50 = 20.625.000</b>   | Totale operazioni elementari al giorno                 |  |

### 9.3.2 Operazione 2

Si assume di conoscere l'evento su cui si vuole eseguire l'operazione

| N° operazioni                     | Tipo   | Motivazione   |
|-----------------------------------|--|---|
| <b>250.000</b>                    | Lettura  | Compio una scansione in PRESENZA per selezionare tutti gli invitati all'evento desiderato   |
| <b>5*</b>                         | Lettura  | Compio una scansione sugli invitati trovati al punto precedente per contare quanti di loro sono partecipanti, forse partecipanti e non partecipanti |
| <b>250.000 + 5 = 250.005</b>      | Totale operazioni elementari per la singola operazione |   |
| <b>250.005 x 100 = 25.000.500</b> | Totale operazioni elementari al giorno                 |   |

\* il numero totale degli eventi è 50.000, il numero di invitati a tutti gli eventi è 250.000: il numero medio di invitati ad un evento è ottenuto per divisione.

### 9.3.3 Operazione 3

Si assume di conoscere l'utente di cui si vuole visualizzare il contenuto più popolare.

| N° operazioni   | Tipo   | Motivazione  |
|---|--|--|
| <b>45.000.000</b>   | Lettura  | Compio una scansione in POSSESSO per selezionare i contenuti appartenenti all'utente desiderato        |
| <b>36.500.000</b>   | Lettura  | Compio una scansione in LIKE per trovare tutti i like relativi ai post selezionati al passo precedente |
| <b>7.300*</b>   | Lettura  | Compio una scansione per contare i like dei singoli post dell'utente                                   |
| <b>45.000.000 +<br/>36.500.000 + 7.300 =<br/>81.507.300</b> | Totale operazioni elementari per la singola operazione |  |
| <b>81.507.300 x 500 =<br/>40.753.650.000</b>                | Totale operazioni elementari al giorno                 |  |

\* il numero totale dei contenuti è 45.000.000, il numero totale di like è 36.500.000, il numero di contenuti per ciascun utente è in media 9.000: il numero di like sui contenuti di ciascun utente è ottenuto per proporzione ( $36.500.000 : 45.000.000 = x : 9.000$ ).

### 9.3.4 Operazione 4

Si assume di conoscere l'immagine e la galleria in cui questa è catalogata

| N° operazioni      | Tipo   | Motivazione   |
|--------------------|--|---|
| <b>1 x 2 = 2</b>   | Scrittura  | Compio 1 operazione di scrittura su ARCHIVIAZIONE per eliminare l'immagine dalla galleria |
| <b>2</b>           | Totale operazioni elementari per la singola operazione |   |
| <b>2 x 20 = 40</b> | Totale operazioni elementari al giorno                 |   |

### 9.3.5 Operazione 5

Si assume di conoscere l'utente che inserisce il like e il contenuto sul quale verrà inserito

| N° operazioni              | Tipo   | Motivazione  |
|----------------------------|--|--|
| <b>1 x 2 = 2</b>           | Scrittura  | Compio 1 operazione di scrittura su LIKE per inserire l'associazione tra l'utente e il contenuto |
| <b>2</b>                   | Totale operazioni elementari per la singola operazione |  |
| <b>2 x 20.000 = 40.000</b> | Totale operazioni elementari al giorno                 |  |

### 9.3.6 Operazione 6

Si assume di conoscere l'utente che effettua la modifica della sua partecipazione all'evento e tale evento.

| N° operazioni      | Tipo   | Motivazione  |
|--------------------|--|--|
| <b>1 x 2 = 2</b>   | Scrittura  | Compio 1 operazione di scrittura su Conferma in PRESENZA per modificarne il valore |
| <b>2</b>           | Totale operazioni elementari per la singola operazione |  |
| <b>2 x 10 = 20</b> | Totale operazioni elementari al giorno                 |  |

### 9.3.7 Operazione 7

Si assume di conoscere l'utente di cui si vogliono determinare il numero di follower e di following

| N° operazioni    | Tipo    | Motivazione  |
|------------------|---------|--|
| <b>1.500.000</b> | Lettura | Compio 1 scansione di RAPPORTO per selezionare gli utenti che compaiono come follower  |
| <b>1.500.000</b> | Lettura | Compio 1 scansione di RAPPORTO per selezionare gli utenti che compaiono come following |



|  |  |
|--|--|
| <b>1.500.000 + 1.500.000<br/>= 3.000.000</b> | Totale operazioni elementari per la singola operazione |
| <b>3.000.000 x 800 =<br/>2.400.000.000</b>   | Totale operazioni elementari al giorno                 |

### 9.3.8 Operazione 8

Si assume di conoscere l'utente che effettua la segnalazione e l'utente che la riceve

| N° operazioni        | Tipo   | Motivazione   |
|----------------------|--|---|
| <b>1 x 2 = 2</b>     | Scrittura  | Compio 1 operazione di scrittura su SEGNALANTE per registrare chi ha effettuato la segnalazione |
| <b>1 x 2 = 2</b>     | Scrittura  | Compio 1 operazione di scrittura su SEGNALATO per registrare chi ha ricevuto la segnalazione    |
| <b>1 x 2 = 2</b>     | Scrittura  | Compio 1 operazione di scrittura su SEGNALAZIONE per registrare l'avvenuta segnalazione         |
| <b>2 + 2 + 2 = 6</b> | Totale operazioni elementari per la singola operazione |   |
| <b>6 x 5 = 30</b>    | Totale operazioni elementari al giorno                 |   |

## 9.4 RIDONDANZE

Si valuta la possibilità di migliorare le prestazioni delle operazioni 2, 3, 5 e 6 inserendo le ridondanze:

1. Numero di like su un contenuto
2. Numero di utenti partecipanti, forse partecipanti e non partecipanti a un evento

Si analizzano di seguito le tavole degli accessi delle suddette operazioni con ridondanze.

### 9.4.1 Operazione 3 con ridondanze

| N° operazioni                                | Tipo   | Motivazione   |
|--|--|---|
| <b>45.000.000</b>                            | Lettura  | Compio una scansione in POSSESSO per trovare i contenuti appartenenti all'utente desiderato         |
| <b>7.300</b>                                 | Lettura  | Compio una scansione dei contenuti selezionati al passo precedente per conoscerne il numero di like |
| <b>45.000 + 7.300 =<br/>45.007.300</b>       | Totale operazioni elementari per la singola operazione |   |
| <b>45.007.300 x 500 =<br/>22.503.650.000</b> | Totale operazioni elementari al giorno                 |   |

### 9.4.2 Operazione 5 con ridondanze

| N° operazioni              | Tipo   | Motivazione   |
|----------------------------|--|---|
| <b>1 x 2 = 2</b>           | Scrittura  | Compio 1 operazione di scrittura su LIKE per inserire l'associazione tra l'utente e il contenuto                      |
| <b>1 x 2 = 2</b>           | Scrittura  | Compio un'operazione di scrittura su CONTENUTO per incrementare di 1 l'attributo NumeroLike del contenuto considerato |
| <b>2 + 2 = 4</b>           | Totale operazioni elementari per la singola operazione |   |
| <b>4 x 20.000 = 80.000</b> | Totale operazioni elementari al giorno                 |   |

#### 9.4.3 Operazione 2 con ridondanze

| N° operazioni | Tipo   | Motivazione  |
|---------------|--|--|
| 1             | Lettura  | Compio 1 operazione di lettura su EVENTO per conoscere il numero di partecipanti sicuri, potenziali e non partecipanti all'evento desiderato |
| 1             | Totale operazioni elementari per la singola operazione |  |
| 1 x 100 = 100 | Totale operazioni elementari al giorno                 |  |

#### 9.4.4 Operazione 6 con ridondanze

| N° operazioni | Tipo   | Motivazione   |
|---------------|--|---|
| 1 x 2 = 2     | Scrittura  | Compio 1 operazione di scrittura su Conferma in PRESENZA per modificarne il valore                                    |
| 1 x 2 = 2     | Scrittura  | Compio 1 operazione in scrittura su EVENTO per decrementare NumeroIncerti di 1 e incrementare NumeroPartecipanti di 1 |
| 2 + 2 = 4     | Totale operazioni elementari per la singola operazione |   |
| 4 x 10 = 40   | Totale operazioni elementari al giorno                 |   |

## 9.5 VALUTAZIONE DELLE RIDONDANZE

Si decide ora se mantenere o meno le ridondanze, studiando le seguenti tabelle riepilogative.

#### 9.5.1 Ridondanza 1 – Numero di like su un contenuto

##### *Senza ridondanza*

| Operazione   | Numero operazioni elementari al giorno |
|--------------|--|
| Operazione 3 | 40.753.650.000                         |
| Operazione 5 | 40.000                                 |
| Totale       | 40.753.690.000                         |

##### *Con ridondanza*

| Operazione   | Numero operazioni elementari al giorno |
|--------------|--|
| Operazione 3 | 22.503.650.000                         |
| Operazione 5 | 80.000                                 |
| Totale       | 22.503.730.000                         |

Per cui si può dedurre che la presenza della Ridondanza 1 riduce in media i tempi di esecuzione delle operazioni. Si decide pertanto di mantenere l'attributo NumeroLike nello schema ER finale.

#### 9.5.2 Ridondanza 2 – Numero di partecipanti, forse partecipanti e non partecipanti a evento

##### *Senza ridondanza*

| Operazione   | Numero operazioni elementari al giorno |
|--------------|--|
| Operazione 2 | 25.000.500                             |
| Operazione 6 | 20                                     |
| Totale       | 25.000.520                             |

***Con ridondanza***

| Operazione          | Numero operazioni elementari al giorno |
|---------------------|--|
| <b>Operazione 2</b> | 100                                    |
| <b>Operazione 6</b> | 40                                     |
| <b>Totale</b>       | 140                                    |

Per cui si può dedurre che la presenza della Ridondanza 2 riduce in media i tempi di esecuzione delle operazioni. Si decide pertanto di mantenere gli attributi NumeroPartecipanti, NumeroIncerti, NumeroNonPartecipanti nello schema ER finale.

## 10. SCHEMA DEFINITIVO

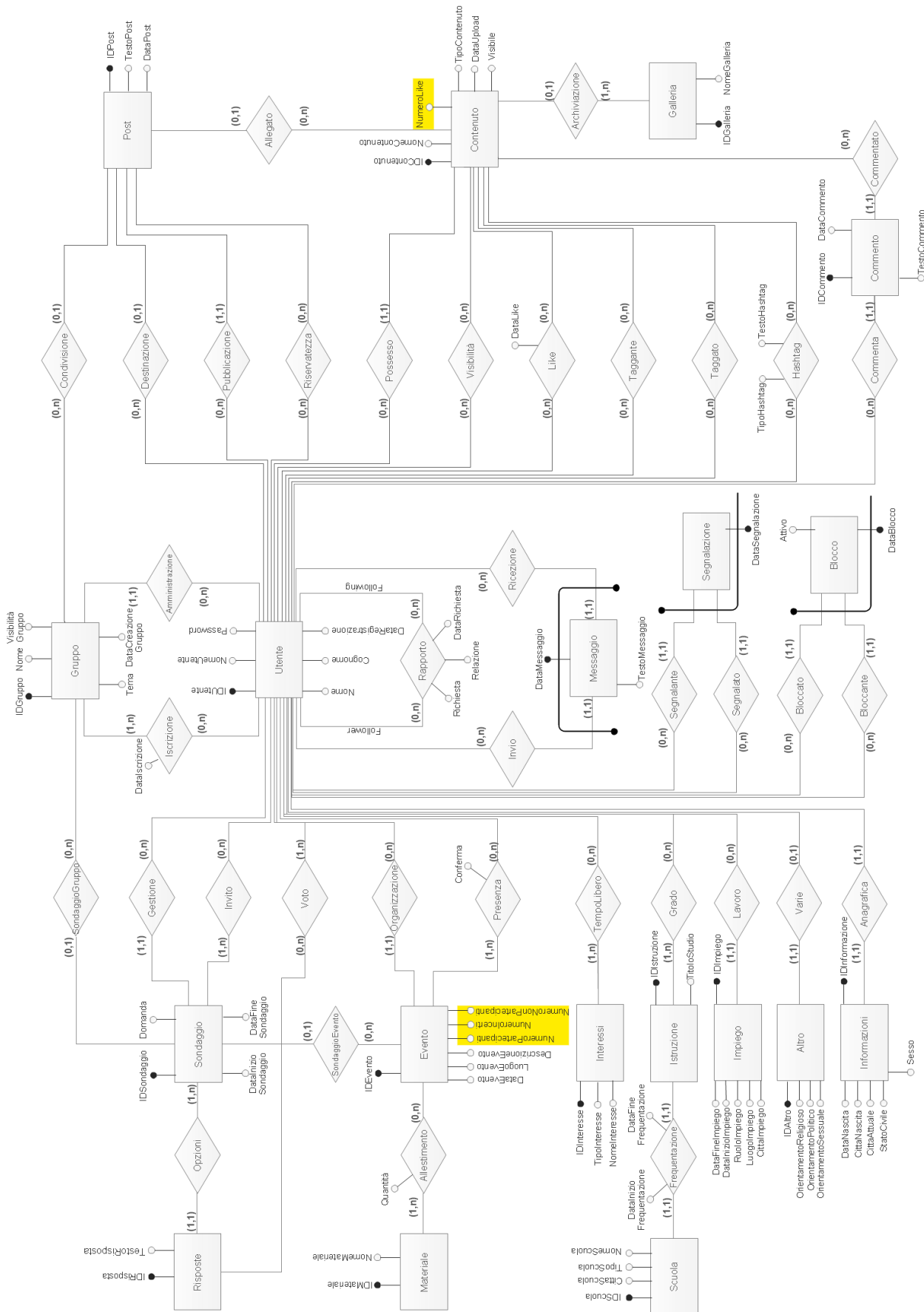


Figura 5 - Diagramma ER definitivo con ridondanze

# 11. TRADUZIONE IN MODELLO RELAZIONALE

---

Si procede alla traduzione del modello ER definitivo in modello relazionale.

Ciascuna entità viene tradotta in una tabella che possiede come campi gli attributi dell'entità stessa, e come chiave l'identificatore dell'entità. Nel caso in cui l'entità abbia un identificatore esterno, anche i campi di entità diverse da essa che fanno parte dell'identificatore vengono inseriti nella tabella e ne costituiscono la chiave.

UTENTE (IDUtente, NomeUtente, Password, Nome, Cognome, DataRegistrazione)

GRUPPO (IDGruppo, Amministratore, Tema, Nome, DataCreazioneGruppo, VisibilitaGruppo)

MESSAGGIO (Mittente, Destinatario, DataMessaggio, TestoMessaggio)

SEGNALAZIONE (Segnalante, Segnalato, DataSegnalazione)

BLOCCO (Bloccante, Bloccato, DataBlocco, Attivo)

SONDAGGIO (IDSondaggio, Gestore, Domanda, DataInizioSondaggio, DataFineSondaggio)

RISPOSTE (IDRisposta, Sondaggio, TestoRisposta)

EVENTO (IDEvento, Organizzatore, DataEvento, LuogoEvento, DescrizioneEvento, NumeroPartecipanti, NumeroIncerti, NumeroNonPartecipanti)

MATERIALE (IDMateriale, NomeMateriale)

INFORMAZIONI (IDInformazione, Utente, DataNascita, CittaNascita, CittaAttuale, StatoCivile, Sesso)

INTERESSE (IDInteresse, TipoInteresse, NomeInteresse)

SCUOLA (IDScuola, TipoScuola, NomeScuola, CittaScuola)

ISTRUZIONE (IDIstruzione, Scuola, TitoloStudio, DataInizioFrequentazione, DataFineFrequentazione)

IMPIEGO (IDImpiego, Utente, CittaImpiego, LuogoImpiego, RuoloImpiego, DataInizioImpiego, DataFineImpiego)

ALTRO (IDAltro, Utente, OrientamentoPolitico, OrientamentoReligioso, OrientamentoSessuale)

POST (IDPost, TestoPost, DataPost)

CONTENUTO (IDContenuto, NomeContenuto, DataUpload, TipoContenuto, Visibile, NumeroLike)

COMMENTO (IDCommento, Utente, Contenuto, DataCommento, TestoCommento)

GALLERIA (IDGalleria, NomeGalleria)

Ciascuna associazione viene tradotta in una tabella che ha come campi gli identificatori di tutte le entità coinvolte, oltre a eventuali attributi propri dell'associazione. La tabella che viene creata ha come chiave l'insieme degli identificatori delle entità connesse all'associazione.

#### RAPPORTO (Follower, Following, Richiesta, DataRichiesta, Relazione)

- Richiesta è implementata come una stringa che può assumere quattro valori: “accettato”, “rifiutato”, “pendente” o “scaduta” (qualora siano trascorsi 30 giorni dalla data di invio della richiesta e non ci sia stata una conferma)

#### ISCRIZIONE (Utente, Gruppo, DataIscrizione)

- DataIscrizione specifica la data in cui l'amministratore ha accettato la richiesta dell'utente a partecipare al gruppo

#### INVITO (Invitato, Sondaggio)

#### VOTO (Utente, RispostaSondaggio)

- Questa tabella contiene le preferenze dei partecipanti al sondaggio

#### SONDAGGIOGRUPPO (Sondaggio, Gruppo)

- Questa tabella contiene esclusivamente i sondaggi effettuati in un gruppo

#### PRESENZA (Utente, Evento, Conferma)

#### ALLESTIMENTO (Evento, Materiale, Quantita)

#### SONDAGGIOEVENTO (Sondaggio, Evento)

- Questa tabella contiene esclusivamente i sondaggi relativi agli eventi

#### TEMPOLIBERO (Utente, Interesse)

#### GRADO (Utente, Istruzione)

#### PUBBLICAZIONE (Utente, PostPubblicato)

- Questa tabella contiene esclusivamente i post pubblicati dagli utenti sulle proprie bacheche

#### DESTINAZIONE (Destinatario, Post)

- Questa tabella contiene esclusivamente i post pubblicati sulle bacheche di utenti diversi dai proprietari, e relativi destinatari

#### RISERVATEZZA (Visualizzante, Post)

- Questa tabella contiene esclusivamente i post a visibilità intermedia, e gli utenti che possono visualizzarli

#### CONDIVISIONE (Post, Gruppo)

- Questa tabella contiene esclusivamente i post pubblicati all'interno di gruppi, e relativi gruppi

#### ALLEGATO (Post, Contenuto)

#### POSSESSO (Utente, Contenuto)

VISIBILITA (Visualizzante, Contenuto)

- Questa tabella contiene esclusivamente i contenuti a visibilità intermedia, e gli utenti che possono visualizzarli

LIKE (Liker, Contenuto, DataLike)

TAGGANTE (Taggante, Contenuto)

TAGGATO (Taggato, Contenuto)

HASHTAG (Hashtagger, Contenuto, TipoHashtag, TestoHashtag)

ARCHIVIAZIONE (Galleria, Immagine)

## 12. VINCOLI DI INTEGRITA' REFERENZIALE

Nel paragrafo precedente sono state ricavate le tabelle della base di dati a partire dallo schema ER (Figura 5). Tali tabelle per poter contenere informazioni plausibili devono verificare una serie di vincoli di integrità referenziale.

La seguente tabella esplica i suddetti vincoli.

| Da                |                 | A            |            |
|-------------------|-----------------|--------------|------------|
| Attributo         | Tabella         | Attributo    | Tabella    |
| Amministratore    | Gruppo          | IDUtente     | Utente     |
| Mittente          | Messaggio       | IDUtente     | Utente     |
| Destinatario      | Messaggio       | IDUtente     | Utente     |
| Segnalante        | Segnalazione    | IDUtente     | Utente     |
| Segnalato         | Segnalazione    | IDUtente     | Utente     |
| Bloccante         | Blocco          | IDUtente     | Utente     |
| Bloccato          | Blocco          | IDUtente     | Utente     |
| Gestore           | Sondaggio       | IDUtente     | Utente     |
| Sondaggio         | Risposte        | IDSondaggio  | Sondaggio  |
| Organizzatore     | Evento          | IDUtente     | Utente     |
| Utente            | Informazioni    | IDUtente     | Utente     |
| Scuola            | Istruzione      | IDScuola     | Scuola     |
| Utente            | Impiego         | IDUtente     | Utente     |
| Utente            | Altro           | IDUtente     | Utente     |
| Utente            | Commento        | IDUtente     | Utente     |
| Contenuto         | Commento        | IDContenuto  | Contenuto  |
| Follower          | Rapporto        | IDUtente     | Utente     |
| Following         | Rapporto        | IDUtente     | Utente     |
| Utente            | Iscrizione      | IDUtente     | Utente     |
| Gruppo            | Iscrizione      | IDGruppo     | Gruppo     |
| Invitato          | Invito          | IDUtente     | Utente     |
| Sondaggio         | Invito          | IDSondaggio  | Sondaggio  |
| Utente            | Voto            | IDUtente     | Utente     |
| RispostaSondaggio | Voto            | IDRisposta   | Risposte   |
| Sondaggio         | SondaggioGruppo | IDSondaggio  | Sondaggio  |
| Gruppo            | SondaggioGruppo | IDGruppo     | Gruppo     |
| Utente            | Presenza        | IDUtente     | Utente     |
| Evento            | Presenza        | IDEvento     | Evento     |
| Evento            | Allestimento    | IDEvento     | Evento     |
| Materiale         | Allestimento    | IDMateriale  | Materiale  |
| Sondaggio         | SondaggioEvento | IDSondaggio  | Sondaggio  |
| Evento            | SondaggioEvento | IDEvento     | Evento     |
| Utente            | TempoLibero     | IDUtente     | Utente     |
| Interesse         | TempoLibero     | IDInteresse  | Interesse  |
| Utente            | Grado           | IDUtente     | Utente     |
| Istruzione        | Grado           | IDIstruzione | Istruzione |
| Utente            | Pubblicazione   | IDUtente     | Utente     |
| PostPubblicato    | Pubblicazione   | IDPost       | Post       |
| Destinatario      | Destinazione    | IDUtente     | Utente     |
| Post              | Destinazione    | IDPost       | Post       |
| Visualizzante     | Riservatezza    | IDUtente     | Utente     |
| Post              | Riservatezza    | IDPost       | Post       |



|               |               |             |           |
|---------------|---------------|-------------|-----------|
| Post          | Condivisione  | IDPost      | Post      |
| Gruppo        | Condivisione  | IDGruppo    | Gruppo    |
| Post          | Allegato      | IDPost      | Post      |
| Contenuto     | Allegato      | IDContenuto | Contenuto |
| Visualizzante | Visibilita    | IDUtente    | Utente    |
| Contenuto     | Visibilita    | IDContenuto | Contenuto |
| Liker         | Like          | IDUtente    | Utente    |
| Contenuto     | Like          | IDContenuto | Contenuto |
| Taggante      | Taggante      | IDUtente    | Utente    |
| Contenuto     | Taggante      | IDContenuto | Contenuto |
| Taggato       | Taggato       | IDUtente    | Utente    |
| Contenuto     | Taggato       | IDContenuto | Contenuto |
| Hashtagger    | Hashtag       | IDUtente    | Utente    |
| Contenuto     | Hashtag       | IDContenuto | Contenuto |
| Galleria      | Archiviazione | IDGalleria  | Galleria  |
| Immagine      | Archiviazione | IDContenuto | Contenuto |

# III. NORMALIZZAZIONE

Si intende provare la correttezza della traduzione dello schema ER nel modello relazionale precedentemente ottenuta attraverso l'analisi delle dipendenze funzionali.

## 13. DIPENDENZE FUNZIONALI

---

UTENTE (IDUtente, NomeUtente, Password, Nome, Cognome, DataRegistrazione)

- IDUtente → NomeUtente, Password, Nome, Cognome, DataRegistrazione
- NomeUtente, Password → IDUtente, Nome, Cognome, DataRegistrazione

*NomeUtente e Password, pur non essendo chiave primaria, sono una superchiave, in quanto non esistono due istanze di UTENTE che abbiano lo stesso valore.*

Poiché la parte sinistra è superchiave, UTENTE è in BCNF.

GRUPPO (IDGruppo, Amministratore, Tema, Nome, DataCreazioneGruppo, VisibilitaGruppo)

- IDGruppo → Amministratore, Tema, Nome, DataCreazioneGruppo, VisibilitaGruppo

Poiché la parte sinistra è superchiave, GRUPPO è in BCNF.

MESSAGGIO (Mittente, Destinatario, DataMessaggio, TestoMessaggio)

Non esistono dipendenze funzionali, quindi MESSAGGIO è in BCNF.

SEGNALAZIONE (Segnalante, Segnalato, DataSegnalazione)

Non esistono dipendenze funzionali, quindi SEGNALAZIONE è in BCNF.

BLOCCO (Bloccante, Bloccato, DataBlocco, Attivo)

- Bloccante, Bloccato, DataBlocco → Attivo

Poiché la parte sinistra è superchiave, BLOCCO è in BCNF.

SONDAGGIO (IDSondaggio, Gestore, Domanda, DataInizioSondaggio, DataFineSondaggio)

- IDSondaggio → Gestore, Domanda, DataInizioSondaggio, DataFineSondaggio

Poiché la parte sinistra è superchiave, SONDAGGIO è in BCNF.

RISPOSTE (IDRisposta, Sondaggio, TestoRisposta)

- IDRisposta → Sondaggio, TestoRisposta

Poiché la parte sinistra è superchiave, RISPOSTE è in BCNF.

EVENTO (IDEvento, Organizzatore, DataEvento, LuogoEvento, DescrizioneEvento, NumeroPartecipanti, NumeroIncerti, NumeroNonPartecipanti)

- IDEvento → Organizzatore, DataEvento, LuogoEvento, DescrizioneEvento, NumeroPartecipanti, NumeroIncerti, NumeroNonPartecipanti

Poiché la parte sinistra è superchiave, EVENTO è in BCNF.

MATERIALE (IDMateriale, NomeMateriale, Quantita)

- IDMateriale → NomeMateriale, Quantita

Poiché la parte sinistra è superchiave, MATERIALE è in BCNF.

INFORMAZIONI (IDInformazione, Utente, DataNascita, CittaNascita, CittaAttuale, StatoCivile, Sesso)

- IDInformazione → Utente, DataNascita, CittaNascita, CittaAttuale, StatoCivile, Sesso
- Utente → IDInformazione, DataNascita, CittaNascita, CittaAttuale, StatoCivile, Sesso

*Utente, pur non essendo chiave primaria, è una superchiave, in quanto non esistono due istanze di INFORMAZIONI che abbia lo stesso valore.*

Poiché la parte sinistra è superchiave, INFORMAZIONI è in BCNF.

INTERESSE (IDInteresse, TipoInteresse, NomeInteresse)

- IDInteresse → TipoInteresse, NomeInteresse

Poiché la parte sinistra è superchiave, INTERESSE è in BCNF.

ISTRUZIONE (IDIstruzione, Scuola, TitoloStudio, DataInizioFrequentazione, DataFineFrequentazione)

- IDIstruzione → Scuola, TitoloStudio, DataInizioFrequentazione, DataFineFrequentazione

Poiché la parte sinistra è superchiave, ISTRUZIONE è in BCNF.

IMPIEGO (IDImpiego, Utente, CittaImpiego, LuogoImpiego, RuoloImpiego,

DataInizioImpiego, DataFineImpiego)

- IDImpiego → Utente, CittaImpiego, LuogoImpiego, RuoloImpiego, DataInizioImpiego, DataFineImpiego
- Utente → IDImpiego, CittaImpiego, LuogoImpiego, RuoloImpiego, DataInizioImpiego, DataFineImpiego

*Utente, pur non essendo chiave primaria, è una superchiave, in quanto non esistono due istanze di IMPIEGO che abbia lo stesso valore.*

Poiché la parte sinistra è superchiave, IMPIEGO è in BCNF.

SCUOLA (IDScuola, TipoScuola, NomeScuola, CittaScuola)

- IDScuola → TipoScuola, NomeScuola, CittaScuola

Poiché la parte sinistra è superchiave, SCUOLA è in BCNF.

ALTRO (IDAltro, Utente, OrientamentoPolitico, OrientamentoReligioso, OrientamentoSessuale)

- IDAltro → Utente, OrientamentoPolitico, OrientamentoReligioso, OrientamentoSessuale
- Utente → IDAltro, OrientamentoPolitico, OrientamentoReligioso, OrientamentoSessuale

*Utente, pur non essendo chiave primaria, è una superchiave, in quanto non esistono due istanze di ALTRO che abbia lo stesso valore.*

Poiché la parte sinistra è superchiave, ALTRO è in BCNF.

POST (IDPost, TestoPost, DataPost)

- IDPost → TestoPost, DataPost

Poiché la parte sinistra è superchiave, POST è in BCNF.

CONTENUTO (IDContenuto, NomeContenuto, DataUpload, TipoContenuto, Visibile, NumeroLike)

- IDContenuto → NomeContenuto, DataUpload, TipoContenuto, Visibile, NumeroLike

Poiché la parte sinistra è superchiave, CONTENUTO è in BCNF.

COMMENTO (IDCommento, Utente, Contenuto, DataCommento, TestoCommento)

- IDCommento → Utente, Contenuto, DataCommento, TestoCommento

Poiché la parte sinistra è superchiave, RAPPORTO è in BCNF.

GALLERIA (IDGalleria, NomeGalleria)

- IDGalleria → NomeGalleria

Poiché la parte sinistra è superchiave, GALLERIA è in BCNF.

RAPPORTO (Follower, Following, Richiesta, DataRichiesta, Relazione)

- Follower, Following → Richiesta, DataRichiesta, Relazione

Poiché la parte sinistra è superchiave, RAPPORTO è in BCNF.

ISCRIZIONE (Utente, Gruppo, DataIscrizione)

- Utente, Gruppo → DataIscrizione

Poiché la parte sinistra è superchiave, ISCRIZIONE è in BCNF.

INVITO (Invitato, Sondaggio)

Non esistono dipendenze funzionali, quindi INVITO è in BCNF.

VOTO (Utente, RispostaSondaggio)

Non esistono dipendenze funzionali, quindi VOTO è in BCNF.

SONDAGGIOGRUPPO (Sondaggio, Gruppo)

Non esistono dipendenze funzionali, quindi SONDAGGIOGRUPPO è in BCNF.

PRESENZA (Utente, Evento, Conferma)

- Utente, Evento → Conferma

Poiché la parte sinistra è superchiave, PRESENZA è in BCNF.

ALLESTIMENTO (Evento, Materiale)

Non esistono dipendenze funzionali, quindi ALLESTIMENTO è in BCNF.

SONDAGGIOEVENTO (Sondaggio, Evento)

Non esistono dipendenze funzionali, quindi SONDAGGIOEVENTO è in BCNF.

### TEMPOLIBERO (Utente, Interesse)

Non esistono dipendenze funzionali, quindi TEMPOLIBERO è in BCNF.

### PUBBLICAZIONE (Utente, PostPubblicato)

Non esistono dipendenze funzionali, quindi PUBBLICAZIONE è in BCNF.

### DESTINAZIONE (Destinatario, Post)

Non esistono dipendenze funzionali, quindi DESTINAZIONE è in BCNF.

### RISERVATEZZA (Visualizzante, Post)

Non esistono dipendenze funzionali, quindi RISERVATEZZA è in BCNF.

### CONDIVISIONE (Post, Gruppo)

Non esistono dipendenze funzionali, quindi CONDIVISIONE è in BCNF.

### ALLEGATO (Post, Contenuto)

Non esistono dipendenze funzionali, quindi ALLEGATO è in BCNF.

### POSSESSO (Utente, Contenuto)

Non esistono dipendenze funzionali, quindi POSSESSO è in BCNF.

### VISIBILITA (Visualizzante, Contenuto)

Non esistono dipendenze funzionali, quindi VISIBILITA è in BCNF.

### LIKE (Liker, Contenuto, DataLike)

- Liker, Contenuto  $\rightarrow$  DataLike

Poiché la parte sinistra è superchiave, LIKE è in BCNF.

### TAGGANTE (Taggante, Contenuto)

Non esistono dipendenze funzionali, quindi TAGGANTE è in BCNF.

### TAGGATO (Taggato, Contenuto)

Non esistono dipendenze funzionali, quindi TAGGATO è in BCNF.

### HASHTAG (Hashtagger, Contenuto, TipoHashtag, TestoHashtag)

- Hashtagger, Contenuto  $\rightarrow$  TipoHashtag, TestoHashtag

Poiché la parte sinistra è superchiave, HASHTAG è in BCNF.

### ARCHIVIAZIONE (Galleria, Immagine)

Non esistono dipendenze funzionali, quindi ARCHIVIAZIONE è in BCNF.

# IV. BUSINESS INTELLIGENCE

## 14. BUSINESS INTELLIGENCE

All'utente che si iscrive al Social Network viene notificato che sulla sua bacheca potranno comparire banner pubblicitari. Il Social Network è finanziato da aziende fornitrici di beni o servizi che investono in questi spazi pubblicitari tanto più capitale quanto più essi possano provocare incrementi alle loro vendite.

E' necessario massimizzare l'efficacia degli spazi pubblicitari mostrandoli solo a chi è veramente interessato ai prodotti o ai servizi promossi dalle aziende che li acquistano.

A questo scopo, si rende necessaria la categorizzazione degli utenti del Social Network, attraverso la progettazione e l'implementazione di operazioni di business intelligence e data mining.

Le strategie di data mining proposte e analizzate sono:

1. Suddivisione degli utenti in base agli interessi e ai temi dei gruppi a cui sono iscritti.
2. Suddivisione degli utenti in base ai like e ai commenti espressi riguardo ai contenuti categorizzati dagli hashtag.

A questo scopo, l'implementazione prevede che il tema di un gruppo, il tipo degli interessi di un utente e il tipo degli hashtag possano essere scelti tra i seguenti:

- Sport
- Arte
- Musica
- Libri
- Cinema
- Viaggi
- Cucina
- Giardinaggio
- Tecnologia
- Videogiochi
- Moda
- Motori
- Collezionismo

L'utente che decide di inserire i propri interessi nel suo profilo deve specificarne sia il tipo che il nome.

| Utente 1      |               | Utente 2      |               |
|---------------|---------------|---------------|---------------|
| TipoInteresse | NomeInteresse | TipoInteresse | NomeInteresse |
| Sport         | Calcio        | Sport         | Tennis        |

In questo caso entrambi gli utenti saranno categorizzati per il solo tipo SPORT, e non per i singoli nomi degli interessi.

Allo stesso modo, molti gruppi possono avere come tema CUCINA anche se con nomi diversi, ma la categorizzazione avviene unicamente sul tema del gruppo.

| Gruppo 1   |                    | Gruppo 2   |                  |
|------------|--------------------|------------|------------------|
| TemaGruppo | NomeGruppo         | TemaGruppo | NomeGruppo       |
| Cucina     | La prova del cuoco | Cucina     | Cotto e mangiato |



Infine, anche gli hashtag sono suddivisi esclusivamente per tipo e non per singolo hashtag, come nell'esempio seguente:

| Hashtag 1   |             | Hashtag 2   |                |
|-------------|-------------|-------------|----------------|
| TipoHashtag | NomeHashtag | TipoHashtag | NomeHashtag    |
| Motori      | #Kawasaki   | Motori      | #DucatiMonster |

## 14.1 OPERAZIONE DATA MINING 1

Per implementare la prima operazione di business intelligence, si utilizza la porzione del database indicata dal seguente frammento di diagramma ER.

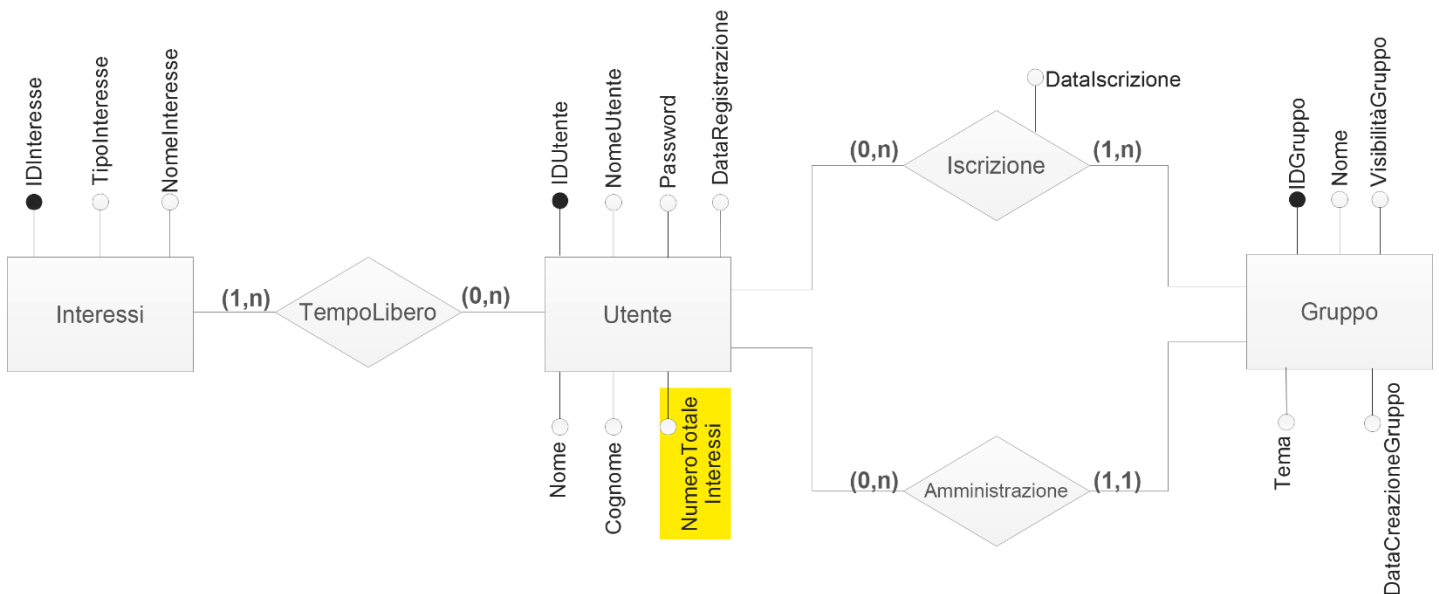


Figura 6 - Porzione del diagramma ER utilizzata per la prima operazione di Business Intelligence

Per ottimizzare la suddetta operazione, si rende necessaria l'aggiunta dell'attributo ridondante *NumeroTotaleInteressi* su *UTENTE*: questo memorizza la somma dei gruppi e degli interessi di ciascun utente. Se l'utente Mario Lasagna si iscrive a due gruppi (anche con tema diverso) e inserisce 3 interessi (anche di tipo diverso), l'attributo segnerà 5.

A seguito di tale aggiunta, la tabella dell'entità *UTENTE* diventa

*UTENTE* (IDUtente, NomeUtente, Password, Nome, Cognome, DataRegistrazione, NumeroTotaleInteressi)

e di conseguenza, le dipendenze funzionali risultano essere

- IDUtente → NomeUtente, Password, Nome, Cognome, DataRegistrazione, NumeroTotaleInteressi
- NomeUtente, Password → IDUtente, Nome, Cognome, DataRegistrazione, NumeroTotaleInteressi

La ridondanza è mantenuta consistente da un trigger, che scatta ogniqualvolta un utente si iscriva a un gruppo o inserisca un nuovo interesse.

L'operazione prevede la creazione, il popolamento e l'aggiornamento di una `MATERIALIZED VIEW` `MV_INTERESSIU-UTENTE` con il seguente schema:

*MV\_InteressiUtente* (IDUtente, Argomento, NumeroIscrizioni, NumeroInteressi)

- *Argomento* contiene i temi dei gruppi e i tipi di interesse, come riportato nel paragrafo precedente.

- *Numeroliscrizioni* mantiene, per ciascun tema di cui in *Argomento*, il numero di gruppi a cui l'utente si è iscritto.
- *NumeroInteressi* mantiene, per ciascun tipo di cui in *Argomento*, il numero di interessi che l'utente ha inserito.

La `MATERIALIZED VIEW` viene popolata inizialmente con una `INSERT` che sfrutta due viste `UTENTEGRUPPO` e `UTENTEINTERESSE` contenenti rispettivamente, per ciascun utente e per ciascun gruppo, il numero di gruppi a cui l'utente è iscritto e il numero di interessi che l'utente ha inserito.

Dopo il popolamento della tabella `MV_INTERESSIUTENTE` essa viene tenuta in uno stato consistente mediante i trigger `NumeroIscrizioni` e `NumeroInteressi`.

- *NumeroIscrizioni*: se un utente non ha mai effettuato iscrizioni ad un gruppo con un dato tema (e quindi non è presente nella tabella) viene aggiunto un nuovo record con l'ID dell'utente, il tema del gruppo come argomento, *Numeroliscrizioni* pari a 1 e *NumeroInteressi* pari a 0; altrimenti il trigger incrementa di 1 il campo *Numeroliscrizioni* relativo al record contenente l'utente e il tema del gruppo lasciando invariato *NumeroInteressi* (infatti l'utente è già presente nella tabella).

Supponiamo che la `MATERIALIZED VIEW` contenga ad un certo istante le seguenti tuple:

| IDUtente | Argomento | Numeroliscrizioni | NumeroInteressi |
|----------|-----------|-------------------|-----------------|
| 1        | Sport     | 3                 | 1               |
| 1        | Arte      | 1                 | 5               |
| 2        | Cucina    | 2                 | 3               |
| 3        | Sport     | 1                 | 0               |

L'utente 1 si iscrive a un gruppo con tema Sport e a uno con tema Viaggi: la `MATERIALIZED VIEW` risulta:

| IDUtente | Argomento | Numeroliscrizioni | NumeroInteressi |
|----------|-----------|-------------------|-----------------|
| 1        | Sport     | $3 + 1 = 4$       | 1               |
| 1        | Arte      | 1                 | 5               |
| 2        | Cucina    | 2                 | 3               |
| 3        | Sport     | 1                 | 0               |
| 1        | Viaggi    | 1                 | 0               |

- *NumeroInteressi*: se un utente non ha mai interessi di un certo tipo (e quindi non è presente nella tabella) viene aggiunto un nuovo record con l'ID dell'utente, il tipo dell'interesse come argomento, *Numeroliscrizioni* pari a 0 e *NumeroInteressi* pari a 1; altrimenti il trigger incrementa di 1 il campo *NumeroInteressi* relativo al record contenente l'utente e il tipo dell'interesse lasciando invariato *Numeroliscrizioni* (infatti l'utente è già presente nella tabella).

Continuando con l'esempio precedente, supponiamo che l'utente 1 inserisca anche 1 interesse con tipo Arte e uno con tipo Giardinaggio: ne risulta

| IDUtente | Argomento    | Numeroliscrizioni | NumeroInteressi |
|----------|--------------|-------------------|-----------------|
| 1        | Sport        | 4                 | 1               |
| 1        | Arte         | 1                 | $5 + 1 = 6$     |
| 2        | Cucina       | 2                 | 3               |
| 3        | Sport        | 1                 | 0               |
| 1        | Viaggi       | 1                 | 0               |
| 1        | Giardinaggio | 0                 | 1               |

Infine la procedura **UtentiTarget()** stampa gli utenti e gli argomenti ai quali essi sono più interessati in modo che possa comparire sulla loro bacheca un banner relativo a quell'interesse.

Per stabilire gli argomenti di maggiore interessi per un utente, si controlla se la somma tra *NumeroIscrizioni* e *NumeroInteressi* relativa a quell'argomento sia maggiore o uguale al 30% del numero totale di interessi di quell'utente (salvato nell'attributo *NumeroTotaleInteressi*).

Per concludere, chiamando la procedura **UtentiTarget()** sulla **MATERIALIZED VIEW** dell'esempio precedente, il risultato sarà:

| Utente | Argomento |
|--------|-----------|
| 1      | Sport     |
| 1      | Arte      |
| 2      | Cucina    |
| 3      | Sport     |

Di seguito si riporta l'implementazione in codice SQL della prima operazione di business intelligence:

```
ALTER TABLE Utente
ADD COLUMN NumeroTotaleInteressi INT(50) DEFAULT 0;

DROP TABLE IF EXISTS MV_InteressiUtente;
CREATE TABLE MV_InteressiUtente
(
    IDUtente INT(50) NOT NULL,
    Argomento VARCHAR(255) NOT NULL,
    NumeroIscrizioni INT(50) default 0,
    NumeroInteressi int(50) default 0,
    PRIMARY KEY (IDUtente, Argomento),
    FOREIGN KEY (IDUtente)
        REFERENCES UTENTE (IDUtente)
)ENGINE = InnoDB DEFAULT CHARSET = latin1;

CREATE OR REPLACE VIEW UtenteGruppo AS
(
    SELECT I.Utente, G.Tema, COUNT(*) AS NumeroIscrizioni
    FROM Iscrizione I INNER JOIN Gruppo G ON I.Gruppo = G.IDGruppo
    GROUP BY I.Utente, G.Tema
);

CREATE OR REPLACE VIEW UtenteInteresse AS
(
    SELECT TL.Utente, I.TipoInteresse, COUNT(*) AS NumeroInteressi
    FROM Interesse I INNER JOIN TempoLibero TL ON I.IDInteresse = TL.Interesse
    GROUP BY TL.Utente, I.TipoInteresse
);

INSERT INTO MV_InteressiUtente
SELECT UG.Utente AS IDUtente, UG.Tema AS Argomento, UG.NumeroIscrizioni, UI.NumeroInteressi
FROM UtenteGruppo UG NATURAL JOIN UtenteInteresse UI
WHERE UG.Tema = UI.TipoInteresse
UNION
SELECT UG.Utente AS IDUtente, UG.Tema AS Argomento, UG.NumeroIscrizioni, 0
FROM UtenteGruppo UG NATURAL LEFT OUTER JOIN UtenteInteresse UI
WHERE UI.TipoInteresse IS NULL
OR UI.Utente IS NULL
UNION
SELECT UI.Utente AS IDUtente, UI.TipoInteresse AS Argomento, 0, UI.NumeroInteressi
FROM UtenteGruppo UG NATURAL RIGHT OUTER JOIN UtenteInteresse UI
WHERE UG.Tema IS NULL
OR UG.Utente IS NULL;
```

```

DROP TRIGGER IF EXISTS NumeroIscrizioni;
DELIMITER $$
CREATE TRIGGER NumeroIscrizioni
AFTER INSERT ON Iscrizione
FOR EACH ROW
BEGIN
    UPDATE Utente
    SET NumeroTotaleInteressi=NumeroTotaleInteressi + 1
    WHERE IDUtente = New.Utente;

    SET @nuovoTema = (SELECT Tema
                      FROM Gruppo
                      WHERE IDGruppo = NEW.Gruppo);

    SET @esiste = (SELECT COUNT(*)
                  FROM MV_InteressiUtente
                  WHERE IDUtente = NEW.Utente
                  AND Argomento = @nuovoTema);

    IF @esiste = 1
    THEN UPDATE MV_InteressiUtente
    SET NumeroIscrizioni=NumeroIscrizioni + 1
    WHERE IDUtente = New.Utente
    AND Argomento = @nuovoTema;
    ELSE INSERT INTO MV_InteressiUtente
    VALUES (NEW.Utente, @nuovoTema, 1, 0);
    END IF;
END $$
DELIMITER ;

DROP TRIGGER IF EXISTS NumeroInteressi;
DELIMITER $$
CREATE TRIGGER NumeroInteressi
AFTER INSERT ON TempoLibero
FOR EACH ROW
BEGIN
    UPDATE Utente
    SET NumeroTotaleInteressi=NumeroTotaleInteressi + 1
    WHERE IDUtente = New.Utente;

    SET @nuovoInteresse = (SELECT TipoInteresse
                           FROM Interesse
                           WHERE IDInteresse = NEW.Interesse);

    SET @esiste = (SELECT COUNT(*)
                  FROM MV_InteressiUtente
                  WHERE IDUtente = NEW.Utente
                  AND Argomento = @nuovoInteresse);

    IF @esiste = 1
    THEN UPDATE MV_InteressiUtente
    SET NumeroInteressi=NumeroInteressi + 1
    WHERE IDUtente = New.Utente
    AND Argomento = @nuovoInteresse;
    ELSE INSERT INTO MV_InteressiUtente
    VALUES (NEW.Utente, @nuovoInteresse, 0, 1);
    END IF;
END $$
DELIMITER ;

```

```

DROP PROCEDURE IF EXISTS UtentiTarget;
DELIMITER $$
CREATE PROCEDURE UtentiTarget ()
BEGIN
    CREATE TABLE IF NOT EXISTS utentiTarget
    (
        Utente INT(50) NOT NULL,
        Argomento VARCHAR(255) NOT NULL,
        PRIMARY KEY (Utente, Argomento)
    );

    TRUNCATE TABLE utentiTarget;

    INSERT INTO utentiTarget
    SELECT IU.IDUtente, IU.Argomento
    FROM MV_InteressiUtente IU
    WHERE IU.NumeroIscrizioni + IU.NumeroInteressi >= 0.3*
        (SELECT U.NumeroTotaleInteressi
        FROM Utente U
        WHERE U.IDUtente = IU.IDUtente);

    SELECT *
    FROM utentiTarget;
END $$
DELIMITER ;

CALL UtentiTarget();

```

## 14.2 OPERAZIONE DATA MINING 2

Per implementare la seconda operazione di business intelligence, si utilizza la porzione del database indicata dal seguente frammento di diagramma ER.

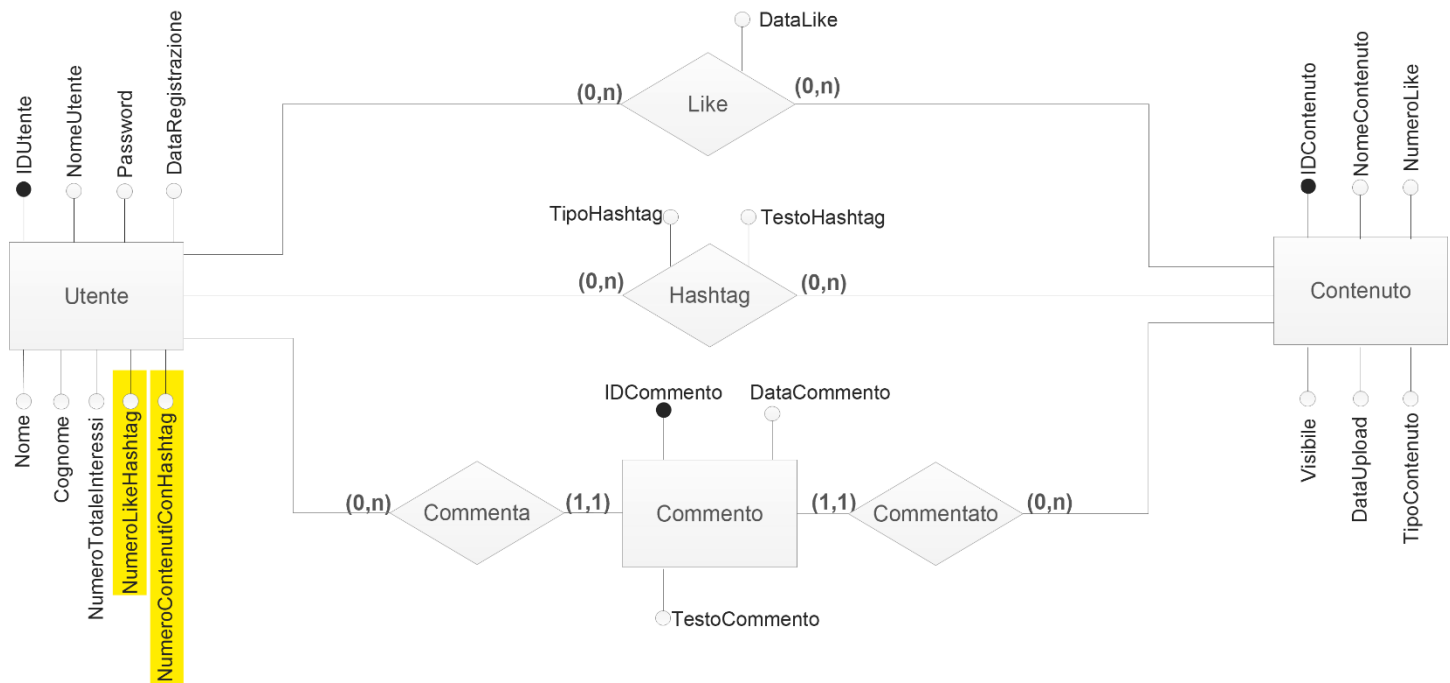


Figura 7 - Porzione del diagramma ER utilizzata per la seconda operazione di Business Intelligence

Per ottimizzare la suddetta operazione, si rende necessaria l'aggiunta degli attributi ridondanti *NumeroLikeHashtag* e *NumeroContenutiConHashtag* su *UTENTE*:

1. *NumeroLikeHashtag* contiene per ciascun utente il numero di like inseriti su contenuti categorizzati per hashtag.
2. *NumeroContenutiConHashtag* contiene per ciascun utente il numero dei suoi contenuti categorizzati per hashtag (l'hashtag può essere inserito dal proprietario del contenuto o da un altro utente).

A seguito di tale aggiunta, la tabella dell'entità *UTENTE* diventa

*UTENTE* (*IDUtente*, *NomeUtente*, *Password*, *Nome*, *Cognome*, *DataRegistrazione*, *NumeroTotaleInteressi*, *NumeroLikeHashtag*, *NumeroContenutiConHashtag*)

e di conseguenza, le dipendenze funzionali risultano essere

- *IDUtente* → *NomeUtente*, *Password*, *Nome*, *Cognome*, *DataRegistrazione*, *NumeroTotaleInteressi*, *NumeroLikeHashtag*, *NumeroContenutiConHashtag*
- *NomeUtente*, *Password* → *IDUtente*, *Nome*, *Cognome*, *DataRegistrazione*, *NumeroTotaleInteressi*, *NumeroLikeHashtag*, *NumeroContenutiConHashtag*

La ridondanze sono mantenute consistenti da due trigger, che scattano ogniqualvolta un utente aggiunga un like a un contenuto a cui è associato almeno un hashtag o pubblichi una risorsa con un hashtag o se un utente aggiunga un hashtag a un contenuto di un altro utente.

L'operazione prevede la creazione, il popolamento e l'aggiornamento di una *MATERIALIZED VIEW* *MV\_LIKECONTENUTIINTERESSI* con il seguente schema:

*MV\_LikeContenutiInteressi* (*IDUtente*, *Argomento*, *NumeroLikeArgomento*, *NumeroContenutiPerArgomento*)

- *Argomento* contiene i tipi degli hashtag (vedi apertura del Capitolo 14)
- *NumeroLikeArgomento* mantiene, per ogni tipo di hashtag di cui in *Argomento*, il numero di like che l'utente ha inserito in contenuti categorizzati con quel tipo di hashtag.
- *NumeroContenutiPerArgomento* mantiene, per ciascun tipo di hashtag di cui in *Argomento*, il numero di contenuti posseduti dall'utente su cui è presente un hashtag di quel tipo.

La *MATERIALIZED VIEW* viene popolata inizialmente con una *INSERT* che sfrutta due *viste* *LIKEUTENTE* e *CONTENUTIUTENTI* contenenti rispettivamente, per ciascun utente e per ciascun tipo di hashtag, il numero di like che l'utente ha inserito e il numero di contenuti posseduti.

Dopo il popolamento della tabella *MV\_LIKECONTENUTIINTERESSI* essa viene tenuta in uno stato consistente mediante i trigger *AggiornaNumeroLikeArgomento* e *AggiornaNumeroContenutiPerArgomento*.

- *AggiornaNumeroLikeArgomento*: se un utente non ha mai inserito un like su un contenuto con un dato tipo di hashtag (e quindi non è presente nella tabella) viene aggiunto un nuovo record con l'ID dell'utente, il tipo di hashtag come argomento, *NumeroLikeArgomento* pari a 1 e *NumeroContenutiPerArgomento* pari a 0; altrimenti il trigger incrementa di 1 il campo *NumeroLikeArgomento* relativo al record contenente l'utente e il tipo di hashtag, lasciando invariato *NumeroContenutiPerArgomento* (infatti l'utente è già presente nella tabella).

Supponiamo che la *MATERIALIZED VIEW* contenga ad un certo istante le seguenti tuple:

| IDUtente | Argomento  | NumeroLikeArgomento | NumeroContenutiPerArgomento |
|----------|------------|---------------------|-----------------------------|
| 4        | Natura     | 2                   | 5                           |
| 4        | Tecnologia | 1                   | 6                           |
| 10       | Cinema     | 3                   | 2                           |
| 6        | Motori     | 0                   | 1                           |

L'utente 10 aggiunge un like a un contenuto caratterizzato da un hashtag con tipo Moda e a uno con tipo Cinema: la `MATERIALIZED VIEW` risulta:

| IDUtente | Argomento  | NumeroLikeArgomento | NumeroContenutiPerArgomento |
|----------|------------|---------------------|-----------------------------|
| 4        | Natura     | 2                   | 5                           |
| 4        | Tecnologia | 1                   | 6                           |
| 10       | Cinema     | 3 + 1 = 4           | 2                           |
| 6        | Motori     | 0                   | 1                           |
| 10       | Moda       | 1                   | 0                           |

- `AggiornaNumeroContenutiPerArgomento`: se un utente non ha mai pubblicato un contenuto sul quale sia stato inserito un dato tipo di hashtag (e quindi non è presente nella tabella) viene aggiunto un nuovo record con l'ID dell'utente, il tipo dell'hashtag come argomento, `NumeroLikeArgomento` pari a 0 e `NumeroContenutiPerArgomento` pari a 1; altrimenti il `trigger` incrementa di 1 il campo `NumeroContenutiPerArgomento` lasciando invariato `NumeroLikeArgomento` (infatti l'utente è già presente nella tabella).

Continuando con l'esempio precedente, supponiamo che a un contenuto caricato dall'utente 10 venga aggiunto un hashtag di tipo Moda e uno di tipo Videogiochi: ne risulta

| IDUtente | Argomento   | NumeroLikeArgomento | NumeroContenutiPerArgomento |
|----------|-------------|---------------------|-----------------------------|
| 4        | Natura      | 2                   | 5                           |
| 4        | Tecnologia  | 1                   | 6                           |
| 10       | Cinema      | 3 + 1 = 4           | 2                           |
| 6        | Motori      | 0                   | 1                           |
| 10       | Moda        | 1                   | 0 + 1 = 1                   |
| 10       | Videogiochi | 0                   | 1                           |

Infine la procedura **UtentiInteressati()** stampa gli utenti i tipi di hashtag ai quali sono più interessati, in modo che possa apparire sulla loro bacheca un banner relativo a quell'argomento.

Per decidere gli argomenti più seguiti da un utente, si controlla se la somma tra `NumeroLikeArgomento` e `NumeroContenutiPerArgomento` relativa a quegli argomenti sia maggiore o uguale al 40% della somma tra `NumeroLikeHashtag` e `NumeroContenutiConHashtag` di quell'utente.

Per concludere, chiamando la procedura **UtentiInteressati()** sulla `MATERIALIZED VIEW` dell'esempio precedente, il risultato sarà:

| Utente | Argomento  |
|--------|------------|
| 4      | Natura     |
| 4      | Tecnologia |
| 10     | Cinema     |
| 6      | Motori     |

Di seguito si riporta l'implementazione in codice SQL della seconda operazione di business intelligence:

```
ALTER TABLE Utente
ADD COLUMN NumeroLikeHashtag INT(50) DEFAULT 0;

ALTER TABLE Utente
ADD COLUMN NumeroContenutiConHashtag INT(50) DEFAULT 0;
```

```

DROP TABLE IF EXISTS MV_LikeContenutiInteressi;
CREATE TABLE MV_LikeContenutiInteressi
(
    IDUtente INT(50) NOT NULL,
    Argomento VARCHAR(255) NOT NULL,
    NumeroLikeArgomento INT(50) DEFAULT 0,
    NumeroContenutiPerArgomento INT(50) DEFAULT 0,
    PRIMARY KEY (IDUtente, Argomento),
    FOREIGN KEY (IDUtente)
        REFERENCES UTENTE(IDUtente)
) ENGINE = InnoDB DEFAULT CHARSET = latin1;

CREATE OR REPLACE VIEW LikeUtente AS
(
    SELECT L.Liker AS IDUtente, H.TipoHashtag AS Argomento,
           COUNT(*) AS NumeroLikeArgomento
    FROM Hashtag H NATURAL JOIN `Like` L
    GROUP BY L.Liker, H.TipoHashtag
);

CREATE OR REPLACE VIEW ContenutiUtente AS
(
    SELECT P.Utente AS IDUtente, H.TipoHashtag AS Argomento,
           COUNT(*) AS NumeroContenutiPerArgomento
    FROM Possesso P NATURAL JOIN Hashtag H
    GROUP BY P.Utente, H.TipoHashtag
);

INSERT INTO MV_LikeContenutiInteressi
SELECT LU.IDUtente, LU.Argomento, LU.NumeroLikeArgomento, CU.NumeroContenutiPerArgomento
FROM LikeUtente LU NATURAL JOIN ContenutiUtente CU
UNION
SELECT LU.IDUtente, LU.Argomento, LU.NumeroLikeArgomento, 0
FROM LikeUtente LU NATURAL LEFT OUTER JOIN ContenutiUtente CU
WHERE CU.Argomento IS NULL
      OR CU.IDUtente IS NULL
UNION
SELECT CU.IDUtente, CU.Argomento, 0, CU.NumeroContenutiPerArgomento
FROM LikeUtente LU NATURAL RIGHT OUTER JOIN ContenutiUtente CU
WHERE LU.Argomento IS NULL
      OR LU.IDUtente IS NULL;

DROP TRIGGER IF EXISTS AggiornaNumeroLikeArgomento;
DELIMITER $$
CREATE TRIGGER AggiornaNumeroLikeArgomento
AFTER INSERT ON `Like`
FOR EACH ROW
BEGIN
    INSERT INTO MV_LikeContenutiInteressi(IDUtente, Argomento, NumeroLikeArgomento)
    SELECT L.Liker, H.TipoHashtag, 1
    FROM `Like` L NATURAL JOIN Hashtag H
    WHERE L.Liker = NEW.Liker
           AND L.Contenuto = NEW.Contenuto
           AND H.TipoHashtag IS NOT NULL
    ON DUPLICATE KEY
    UPDATE NumeroLikeArgomento = NumeroLikeArgomento + 1;
END $$
DELIMITER ;

```



```

DROP TRIGGER IF EXISTS AggiornaNumeroContenutiPerArgomento;
DELIMITER $$
CREATE TRIGGER AggiornaNumeroContenutiPerArgomento
AFTER INSERT ON Hashtag
FOR EACH ROW
BEGIN
    SET @possessore = (SELECT P.Utente
                        FROM Hashtag H NATURAL JOIN Possesso P
                        WHERE H.Hashtagger = NEW.Hashtagger
                        AND H.Contenuto = NEW.Contenuto);

    UPDATE Utente
        SET NumeroContenutiConHashtag = NumeroContenutiConHashtag + 1
        WHERE IDUtente = @possessore;

    INSERT INTO MV_LikeContenutiInteressi(IDUtente,Argomento,NumeroContenutiPerArgomento)
        SELECT P.Utente, NEW.TipoHashtag, 1
        FROM Hashtag H NATURAL JOIN Possesso P
        WHERE H.Hashtagger = NEW.Hashtagger
        ON DUPLICATE KEY
        UPDATE NumeroContenutiPerArgomento = NumeroContenutiPerArgomento + 1;
END $$
DELIMITER ;

DROP PROCEDURE IF EXISTS UtentiInteressati;
DELIMITER $$
CREATE PROCEDURE UtentiInteressati()
BEGIN
    CREATE TABLE IF NOT EXISTS utentiInteressati
    (
        Utente INT(50) NOT NULL,
        Argomento VARCHAR(255) NOT NULL,
        PRIMARY KEY (Utente, Argomento)
    );

    TRUNCATE TABLE utentiInteressati;

    INSERT INTO utentiInteressati
    SELECT LCI.IDUtente, LCI.Argomento
    FROM MV_LikeContenutiInteressi LCI
    WHERE LCI.NumeroLikeArgomento + LCI.NumeroContenutiPerArgomento >= 0.4 *
        (SELECT U.NumeroLikeHashtag + U.NumeroContenutiConHashtag
         FROM Utente U
         WHERE U.IDUtente = LCI.IDUtente);

    SELECT *
    FROM utentiInteressati;
END $$
DELIMITER ;

CALL UtentiInteressati() ;

```

# V. IMPLEMENTAZIONE IN MYSQL

## 15. OPERAZIONI

---

### OPERAZIONE 1

```
CREATE OR REPLACE VIEW OpzioniSondaggio AS
(
    SELECT R.IDRisposta AS Risposta
    FROM Risposte R INNER JOIN Sondaggio S ON (R.Sondaggio = S.IDSondaggio)
    WHERE R.Sondaggio = 2
);

CREATE OR REPLACE VIEW NumeroVotiRisposta AS
(
    SELECT RispostaSondaggio AS Risposta, COUNT(*) AS NumeroVoti
    FROM Voto
    GROUP BY RispostaSondaggio
);

SELECT Risposta, NumeroVoti
FROM OpzioniSondaggio NATURAL JOIN NumeroVotiRisposta;
```

### OPERAZIONE 2

```
CREATE OR REPLACE VIEW Invitati AS
(
    SELECT Utente AS Invitato, Conferma
    FROM Presenza
    WHERE Evento = 3
);

SELECT Conferma, COUNT(*) AS NumeroUtenti
FROM Invitati
GROUP BY Conferma;
```

### OPERAZIONE 2 CON RIDONDANZA

```
SELECT NumeroPartecipanti, NumeroIncerti, NumeroNonPartecipanti
FROM Evento
WHERE IDEvento = 2;
```

### OPERAZIONE 3

```
CREATE OR REPLACE VIEW ContenutiUtente AS
(
    SELECT Contenuto
    FROM Possesso
    WHERE Utente = 5
);
```

```
CREATE OR REPLACE VIEW NumeroLikeContenuto AS
(
    SELECT Contenuto, COUNT(*) AS NumeroLike
    FROM `LIKE`
    GROUP BY Contenuto
);

SELECT Contenuto
FROM ContenutiUtente NATURAL JOIN NumeroLikeContenuto
ORDER BY NumeroLike DESC
LIMIT 1;
```

## OPERAZIONE 3 CON RIDONDANZA

```
CREATE OR REPLACE VIEW ContenutiUtente AS
(
    SELECT Contenuto
    FROM Possesso
    WHERE Utente = 10
);

SELECT CU.Contenuto
FROM Contenuto C INNER JOIN ContenutiUtente CU ON (IDContenuto = CU.Contenuto)
ORDER BY C.NumeroLike
LIMIT 1;
```

## OPERAZIONE 4

```
DELETE FROM Archiviazione
WHERE Galleria = 2
AND Immagine = 13;
```

## OPERAZIONE 5

```
INSERT INTO `LIKE`(Liker, Contenuto, DataLike)
VALUES (5, 12, CURRENT_TIMESTAMP);
```

## OPERAZIONE 5 CON RIDONDANZA

```
INSERT INTO `LIKE`(Liker, Contenuto, DataLike)
VALUES (2, 4, CURRENT_TIMESTAMP);
```

```
DROP TRIGGER IF EXISTS AggiornaNumeroLike;
```

```
DELIMITER $$
CREATE TRIGGER AggiornaNumeroLike
AFTER INSERT ON `LIKE`
FOR EACH ROW
BEGIN
    UPDATE Contenuto
    SET NumeroLike = NumeroLike + 1
    WHERE IDContenuto = OLD.Contenuto;
END $$
DELIMITER ;
```

## OPERAZIONE 6

```
UPDATE Presenza
  SET Conferma = 'Si'
  WHERE Utente = 2
        AND EVENTO = 4;
```

## OPERAZIONE 6 CON RIDONDANZA

```
UPDATE Presenza
  SET Conferma = 'Si'
  WHERE Utente = 2
        AND EVENTO = 4;
```

```
DROP TRIGGER IF EXISTS ModificaPartecipantiEvento;
```

```
DELIMITER $$
CREATE TRIGGER ModificaPartecipantiEvento
AFTER UPDATE ON Presenza
FOR EACH ROW
BEGIN
    UPDATE Evento
      SET NumeroIncerti = NumeroIncerti - 1
      AND NumeroPartecipanti = NumeroPartecipanti + 1
      WHERE IDEvento = OLD.Evento;
END $$
DELIMITER ;
```

## OPERAZIONE 7

```
CREATE OR REPLACE VIEW NumeroFollower AS
(
    SELECT COUNT(*) AS NumeroFollower
    FROM Rapporto
    WHERE Following = 10
          AND Richiesta = 'Accettata'
);

CREATE OR REPLACE VIEW NumeroFollowing AS
(
    SELECT COUNT(*) AS NumeroFollowing
    FROM Rapporto
    WHERE Follower = 10
          AND Richiesta = 'Accettata'
);

SELECT NumeroFollower, NumeroFollowing
FROM NumeroFollower, NumeroFollowing;
```

## OPERAZIONE 8

```
INSERT INTO Segnalazione(Segnalante, Segnalato, DataSegnalazione)
VALUES (2, 4, CURRENT_TIMESTAMP);
```

# 16. REGOLE AZIENDALI

---

## VINCOLO 1

```
DROP TRIGGER IF EXISTS ContolloNomeUtente;
DELIMITER $$
CREATE TRIGGER ContolloNomeUtente BEFORE INSERT ON Utente
FOR EACH ROW
BEGIN
    SET @quanti = (SELECT COUNT(*)
                    FROM Utente
                    WHERE IDUtente = NEW.IDUtente);

    IF @quanti > 0
    THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Utente già esistente';
    END IF;
END $$
DELIMITER ;
```

## VINCOLO 10

```
DROP TRIGGER IF EXISTS ContolloBlocco;
DELIMITER $$
CREATE TRIGGER ContolloBlocco BEFORE INSERT ON Blocco
FOR EACH ROW
BEGIN
    SET @quanti = (SELECT COUNT(*)
                    FROM Segnalazione
                    WHERE Segnalante = NEW.Bloccante
                    AND Segnalato = NEW.Bloccato);

    IF @quanti < 2
    THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Non è stata fatta ancora la seconda segnalazione';
    END IF;
END $$
DELIMITER ;
```

## VINCOLO 17

```
DROP TRIGGER IF EXISTS ContolloSondaggioEvento;
DELIMITER $$
CREATE TRIGGER ContolloSondaggioEvento BEFORE INSERT ON Sondaggio
FOR EACH ROW
BEGIN
    SET @quanti = (SELECT COUNT(*)
                    FROM Presenza P
                    WHERE P.Utente = NEW.Gestore
                    AND P.Evento = (SELECT SE.Evento
                                    FROM SondaggioEvento SE
                                    WHERE SE.Sondaggio = NEW.IDSondaggio)
                    );

    IF @quanti < 1
    THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Utente non iscritto a evento';
    END IF;
END $$
DELIMITER ;
```