

Progettazione e implementazione di una base di dati relazionale per la gestione di conferenze scientifiche

Caporaso Antonio Di Fusco Giorgio

N86003458

N86004389

24 giugno 2023

Indice

1	Traccia	2
1.1	Output attesi dal committente	2
2	Progettazione concettuale	2
2.1	Analisi dei dati	2
3	Schema concettuale	2
4	Ristrutturazione dello schema concettuale	3
4.1	Rimozione degli attributi multivalore	3
4.2	Rimozione classi di associazione	4
4.3	Rimozione generalizzazioni	4
4.4	Scelta degli identificatori principali	4
5	Progettazione logica	4
5.1	Traduzione delle classi	4
5.2	Traduzione delle associazioni	5
5.2.1	Traduzione delle associazioni molti a molti	5
5.2.2	Traduzione delle associazioni uno a molti	6
5.2.3	Traduzione delle associazioni uno a uno	7
5.3	Schema logico	7
6	Implementazione fisica	7
6.1	Definizione delle tabelle	7
6.2	Popolamento	11
6.3	Trigger	11
6.4	Funzioni e procedure	11
A	Dizionario dei dati	11
B	Dizionario delle associazioni	14
C	Dizionario dei vincoli	16

1 Traccia

Si sviluppi un sistema informativo, composto da una base di dati relazionale e da un applicativo Java dotato di GUI (Swing o JavaFX), per la gestione di **conferenze scientifiche**.

Ogni conferenza ha una data di inizio e di fine, una collocazione (sede, indirizzo), uno o più enti che la organizzano, degli sponsor (che coprono in parte le spese), una descrizione, ed un gruppo di organizzatori, che può essere distinto in comitato scientifico e comitato locale (che si occupa cioè della logistica). Di ognuno degli organizzatori, così come di tutti i partecipanti, si riportano titolo, nome, cognome, email ed istituzione di appartenenza.

Ogni conferenza può avere una o più sessioni, anche in parallelo fra loro. Ogni sessione ha una locazione all'interno della sede. Per ogni sessione c'è un programma, che prevede la presenza di un coordinatore (chair) che gestisce la sessione, ed eventualmente di un keynote speaker (un partecipante di particolare rilievo invitato dagli organizzatori). Ogni sessione avrà quindi una successione di interventi ad orari predefiniti e di specifici partecipanti. Per ogni intervento si conserva un abstract (un breve testo in cui viene spiegato il contenuto del lavoro presentato).

Si deve poter considerare la presenza di spazi di intervallo (coffee breaks, pranzo) ma anche la presenza di eventi sociali (cene, gite, etc).

1.1 Output attesi dal committente

1. Documento di Design della base di dati:
 - (a) Class Diagram della base di dati.
 - (b) Dizionario delle Classi, delle Associazioni e dei Vincoli.
 - (c) Schema Logico con descrizione di Trigger e Procedure individuate.
2. File SQL contenenti:
 - (a) Creazione della struttura della base di dati.
 - (b) Popolamento del DB.
 - (c) (Facoltativo, ma apprezzato) README contenente i commenti all'SQL.

2 Progettazione concettuale

2.1 Analisi dei dati

Le entità che possono essere individuate nel problema sono elencate all'interno della Tabella 1.

3 Schema concettuale

Nella Figura 1 è presente lo schema concettuale della base di dati descritta nella sezione 1.

Entità	Descrizione
Conferenza	Per le conferenze delle quali si vuole poter gestire le informazioni. Di ogni conferenza si conservano il <i>nome</i> , l' <i>inizio</i> e la <i>fine</i> e una <i>descrizione</i> .
Ente	Per gli enti che organizzano le conferenze scientifiche. Di ogni ente si conserva il <i>nome</i> e la <i>sigla</i> .
Sponsor	Per gli sponsor che coprono le spese della conferenza. Di ogni sponsor si conserva il <i>nome</i> .
Comitato	Per i gruppi di organizzatori che si occupano della gestione della conferenza. Si distinguono in comitati <i>scientifici</i> e <i>locali</i> .
Organizzatore	Per i membri dei comitati. Di ogni organizzatore si riportano <i>titolo</i> , <i>nome</i> , <i>cognome</i> , <i>email</i> ed <i>istituzione di afferenza</i> .
Sede	Per descrivere il luogo dove si tengono le varie conferenze. Di ogni sede si conservano il <i>nome</i> , l' <i>indirizzo</i> e la <i>città</i> .
Sala	Per tenere traccia dell'ubicazione delle varie sessioni. Di ogni sala si conserva il <i>nome della sala</i> e la sua <i>capacità</i> .
Sessione	Per rappresentare le sessioni di una conferenza. Per ogni sessione si riporta il <i>titolo</i> , un <i>coordinatore</i> , data e orario d' <i>inizio</i> e di <i>fine</i> .
Programma	Per il programma di ciascuna sessione. Ogni programma specifica la presenza di un <i>keynote speaker</i> , ovvero un partecipante di rilievo.
Intervento	Per i vari interventi di una sessione. Per ogni intervento si conserva un <i>abstract</i> , il partecipante (<i>speaker</i>) che effettua l'intervento e l' <i>orario</i> dello stesso.
Partecipante	Per i partecipanti delle varie sessioni. Ogni partecipante ha gli stessi attributi degli organizzatori.
Intervallo	Per descrivere i vari intervalli presenti all'interno di una sessione. Questi possono essere di due tipologie: <i>coffee break</i> oppure dei <i>pranzi</i> . Per ogni intervallo si riporta l' <i>orario</i> .
Evento sociale	Per i vari eventi sociali previsti all'interno di una sessione. Questi possono essere di varia natura. Come per gli intervalli se ne riporta l' <i>orario</i> .

Tabella 1: Entità del problema

4 Ristrutturazione dello schema concettuale

4.1 Rimozione degli attributi multivalore

All'interno del diagramma delle classi mostrato in Figura 1 sono presenti vari attributi multivalore. Per ciascuno di essi sono state fatte le seguenti valutazioni:

1. Si partiziona l'attributo *Indirizzo* presente in SEDE suddividendolo in vari campi *Via*, *Civico*, *Cap*, *City*, *Provincia* e *Nazione* e creando una nuova entità chiamata INDIRIZZO.
2. Si è deciso di partizionare l'attributo *Valuta* presente nella classe di associazione SPONSORIZZAZIONE creando una nuova classe chiamata VALUTA.

4.2 Rimozione classi di associazione

All'interno dello schema concettuale è presente la classe di associazione SPONSORIZZAZIONE all'interno dell'associazione [...] tra CONFERENZA e SPONSOR. Nello schema ristrutturato questa è stata rimossa reificandola e scindendo l'associazione in due associazioni di tipo [1..*].

4.3 Rimozione generalizzazioni

Per quanto riguarda la rimozione delle generalizzazioni presenti nello schema concettuale:

1. Nel caso delle entità COMITATO SCIENTIFICO e COMITATO LOCALE che specializzano la classe COMITATO si è optato per l'accorpamento delle classi figlie all'interno della super-classe attraverso la specifica di una enumerazione chiamata COMITATO_ST composta dai campi *Scientifico* e *Locale*;
2. Nel caso delle entità PRANZO e COFFEE BREAK che specializzano la classe INTERVALLO si è adottato la stessa politica.

4.4 Scelta degli identificatori principali

Risulta conveniente ai fini di una migliore traduzione delle associazioni l'introduzione di chiavi surrogate per ogni entità. Tali chiavi altro non saranno che identificativi numerici interi del tipo *Id_NomeEntità*, eccezion fatta per l'entità VALUTA la quale viene identificata univocamente da una stringa di tre caratteri stando allo standard ISO 4217¹.

5 Progettazione logica

Una volta aver ristrutturato lo schema concettuale mostrato in Figura 1 si procede traducendo le varie associazioni descritte in Figura 2. Iniziamo col tradurre direttamente tutte le classi. Man mano che si andranno a tradurre le varie associazioni andremo a modificare la struttura dei vari schemi relazionali laddove necessario.

5.1 Traduzione delle classi

Si ha quindi:

Indirizzo

<u>Id_Indirizzo</u>	Via	Civico	CAP	City	Provincia	Nazione
---------------------	-----	--------	-----	------	-----------	---------

ENTE

<u>id_ente</u>	nome	sigla
----------------	------	-------

SEDE

<u>id_sede</u>	nome
----------------	------

SPONSOR

<u>id_Sponsor</u>	Nome
-------------------	------

COMITATO

<u>id_Comitato</u>	Tipologia
--------------------	-----------

¹ISO 4217 è uno standard internazionale che descrive codici di tre lettere per definire i nomi delle valute, stabilito dall'Organizzazione internazionale per la normazione (ISO), che viene usato comunemente nel sistema bancario e nel mondo economico, nonché nella stampa specializzata.

ORGANIZZATORE

<u>id_Organizzatore</u>	Nome	Cognome	Titolo	Email
-------------------------	------	---------	--------	-------

SALA

<u>id_Sala</u>	Nome	Capienza
----------------	------	----------

CONFERENZA

<u>id_Conferenza</u>	Nome	Descrizione	Inizio	Fine
----------------------	------	-------------	--------	------

PARTECIPANTE

<u>id_Partecipante</u>	Nome	Cognome	Titolo	Email
------------------------	------	---------	--------	-------

SESSIONE

<u>id_Sessione</u>	Nome	Inizio	Fine
--------------------	------	--------	------

VALUTA

<u>Iso</u>	Nome	Simbolo
------------	------	---------

SPEAKER

<u>id_Speaker</u>	Nome	Cognome	Titolo	Email
-------------------	------	---------	--------	-------

PROGRAMMA

<u>Id_Programma</u>

INTERVALLO

<u>id_Intervallo</u>	Tipologia	Inizio	Fine
----------------------	-----------	--------	------

INTERVENTO

<u>id_Intervento</u>	Titolo	Abstract	Inizio	Fine
----------------------	--------	----------	--------	------

EVENTO

<u>id_Evento</u>	Tipologia	Inizio	Fine
------------------	-----------	--------	------

5.2 Traduzione delle associazioni

5.2.1 Traduzione delle associazioni molti a molti

Traduciamo le associazioni *.* mediante la realizzazioni di apposite tabelle ponte. Si ha allora:

1. L'associazione ENTECONFERENZA tra ENTE e CONFERENZA:

ENTECONFERENZA

<u>id_ente</u>	<u>id_conferenza</u>
----------------	----------------------

2. L'associazione ORGANIZZATORECOMITATO tra ORGANIZZATORE e COMITATO:

ORGANIZZATORECOMITATO

<u>id_organizzatore</u>	<u>id_comitato</u>
-------------------------	--------------------

3. L'associazione PARTECIPANTESESSIONE tra PARTECIPANTE e SESSIONE:

PARTECIPANTESESSIONE

<u>id_Partecipante</u>	<u>id_Sessione</u>
------------------------	--------------------

5.2.2 Traduzione delle associazioni uno a molti

Per ciascuna delle associazioni binarie di tipo uno a molti si identificano le entità deboli e quelle forti che partecipano all'associazione. Per tradurre l'associazione in relazioni basterà includere la chiave surrogata dell'entità forte all'interno della relazione dell'entità debole. Avremo quindi:

1. Associazioni di composizione:

- (a) Una sede è composta da più sale quindi:

SALA

<u>id_Sala</u>	Nome	Capienza	<u>id_sede</u>
----------------	------	----------	----------------

- (b) Una conferenza è composta da più sessioni:

SESSIONE

<u>id_Sessione</u>	Nome	Inizio	Fine	<u>id_conferenza</u>
--------------------	------	--------	------	----------------------

- (c) Un programma è composto da interventi, intervalli ed eventi:

INTERVALLO

<u>id_Intervallo</u>	Tipologia	Inizio	Fine	<u>id_programma</u>
----------------------	-----------	--------	------	---------------------

INTERVENTO

<u>id_Intervento</u>	Titolo	Abstract	Inizio	Fine	<u>id_programma</u>
----------------------	--------	----------	--------	------	---------------------

EVENTO

<u>id_Evento</u>	Tipologia	Inizio	Fine	<u>id_programma</u>
------------------	-----------	--------	------	---------------------

2. Un partecipante, uno speaker ed un organizzatore appartengono ad una istituzione, ovvero un ENTE:

SPEAKER

<u>id_Speaker</u>	Nome	Cognome	Titolo	Email	<u>id_ente</u>
-------------------	------	---------	--------	-------	----------------

PARTECIPANTE

<u>id_Speaker</u>	Nome	Cognome	Titolo	Email	<u>id_ente</u>
-------------------	------	---------	--------	-------	----------------

ORGANIZZATORE

<u>id_Speaker</u>	Nome	Cognome	Titolo	Email	<u>id_ente</u>
-------------------	------	---------	--------	-------	----------------

3. Ogni intervento ha uno speaker che lo effettua:

INTERVENTO

<u>id_Intervento</u>	<u>id_speaker</u>	Titolo	Abstract	Inizio	Fine	<u>id_programma</u>
----------------------	-------------------	--------	----------	--------	------	---------------------

4. Una sala può ospitare più sessioni:

SESSIONE

<u>id_Sessione</u>	Nome	Inizio	Fine	<u>id_sala</u>	<u>id_conferenza</u>
--------------------	------	--------	------	----------------	----------------------

5. Una sede può ospitare più conferenze:

CONFERENZA

<u>id_Conferenza</u>	Nome	Descrizione	Inizio	Fine	<u>id_sede</u>
----------------------	------	-------------	--------	------	----------------

6. Una conferenza ha due comitati, uno scientifico ed uno locale:

CONFERENZA

<u>id_Conferenza</u>	Nome	Descrizione	Inizio	Fine	<u>id_sede</u>	<u>id_com_locale</u>	<u>id_com_scientifico</u>
----------------------	------	-------------	--------	------	----------------	----------------------	---------------------------

5.2.3 Traduzione delle associazioni uno a uno

Si ha:

1. Ogni sessione ha un coordinatore:

SESSIONE

<u>id_Sessione</u>	Nome	Inizio	Fine	<u>id_sala</u>	<u>id_conferenza</u>	<u>id_coordinatore</u>
--------------------	------	--------	------	----------------	----------------------	------------------------

2. Ogni programma si riferisce ad una sessione e ad un keynote speaker:

PROGRAMMA

<u>Id_Programma</u>	<u>id_Sessione</u>	<u>id_keynote</u>
---------------------	--------------------	-------------------

3. Ogni sede ha un indirizzo:

SEDE

<u>id_sede</u>	nome	indirizzo
----------------	------	-----------

5.3 Schema logico

Nella Figura 3 è raffigurato lo schema logico risultante.

6 Implementazione fisica

6.1 Definizione delle tabelle

```
1 CREATE SCHEMA conference;
2 set search_path to conference;
3
4 --Definizione tipi
5 create type comitato_st as enum ('locale','scientifico');
6 create type intervallo_st as enum ('pranzo','coffee break');
7 create type titolo_st as enum ('Dottore','Dottorressa','Professore','
    Professoressa','Assistente','Ricercatore','Ricercatrice','Ingegnere');
8
9 --Creazione tabelle
10 create table ente(
11 id_ente serial primary key,
12 nome text not null unique,
13 sigla varchar(7) not null,
14 unique (nome,sigla)
15 );
16
17 -- TABLE: indirizzo
18 create table indirizzo(
19 id_indirizzo serial primary key,
20 via text not null,
21 civico varchar(5) not null,
22 cap varchar(5) ,
```

```

23     city text not null,
24     provincia varchar(2) not null,
25     nazione text
26 );
27
28 -- TABLE: sede
29 create table sede(
30     id_sede serial primary key,
31     nome text ,
32     indirizzo integer references indirizzo(id_indirizzo) on delete set null
33 );
34
35 -- TABLE: sponsor
36 create table sponsor(
37     id_sponsor serial primary key,
38     nome text not null
39 );
40
41 -- TABLE: comitato
42 create table comitato(
43     id_comitato serial primary key,
44     tipologia comitato_st not null
45 );
46
47 -- TABLE: organizzatore
48 create table organizzatore(
49     id_organizzatore serial primary key,
50     nome text not null,
51     cognome text not null,
52     titolo titolo_st,
53     email text not null unique,
54     id_ente integer references ente(id_ente) on delete cascade -- Istituzione
    di appartenenza
55 );
56
57 -- TABLE: sala
58 create table sala(
59     id_sala serial primary key,
60     nome text not null,
61     capienza integer not null,
62     id_sede integer references sede(id_sede) on delete cascade
63 );
64
65 -- TABLE: conferenza
66 create table conferenza(
67     id_conferenza serial primary key,
68     titolo text not null,
69     descrizione text not null,
70     inizio timestamp not null,
71     fine timestamp not null,
72     id_sede integer references sede(id_sede) on delete set null,
73     id_comitato_scientifico integer references comitato(id_comitato) on delete
    set null,
74     id_comitato_locale integer references comitato(id_comitato) on delete set
    null,
75     check (inizio <= fine), -- Vincolo di integrita': la conferenza deve
    iniziare prima di finire
76     check (inizio >= now()) -- Vincolo di integrita': la conferenza deve
    iniziare in futuro
77 );

```



```

78
79 -- TABLE: partecipante
80 create table partecipante(
81     id_partecipante serial primary key,
82     nome text not null,
83     cognome text not null,
84     titolo titolo_st,
85     email text not null unique,
86     id_ente integer references ente(id_ente) on delete set null -- Istituzione
      di appartenenza
87 );
88
89 -- TABLE: sessione
90 create table sessione(
91     id_sessione serial primary key,
92     titolo text not null,
93     inizio timestamp not null,
94     fine timestamp not null,
95     id_coordinatore integer references organizzatore(id_organizzatore) on
      delete set null,
96     id_conferenza integer references conferenza(id_conferenza) on delete
      cascade,
97     id_sala integer references sala(id_sala) on delete set null,
98     check (inizio <= fine) -- Vincolo di integrita': la sessione deve iniziare
      prima di finire
99 );
100
101 -- TABLE: partecipazione
102 create table partecipazione(
103     id_partecipazione serial primary key,
104     id_partecipante integer references partecipante(id_partecipante) on delete
      cascade,
105     id_sessione integer references sessione(id_sessione) on delete cascade,
106     unique (id_partecipante,id_sessione) -- Vincolo di integrita': un
      partecipante puo' partecipare ad una sessione una sola volta
107 );
108
109 -- TABLE: ente_conferenza
110 create table ente_conferenza(
111     id_ente_conferenza serial primary key,
112     id_ente integer references ente(id_ente) on delete cascade,
113     id_conferenza integer references conferenza(id_conferenza) on delete
      cascade,
114     unique (id_ente,id_conferenza) -- Vincolo di integrita': un ente puo'
      organizzare una conferenza una sola volta
115 );
116
117 -- TABLE: valuta
118 create table valuta(
119     iso char(3) primary key,
120     nome text not null,
121     simbolo text not null
122 );
123
124 -- TABLE: sponsor_conferenza
125 create table sponsor_conferenza(
126     id_sponsor_conferenza serial primary key,
127     id_sponsor integer references sponsor(id_sponsor) on delete cascade,
128     contributo numeric(1000,2) not null,
129     valuta char(3) references valuta(iso) not null,

```

```

130     id_conferenza integer references conferenza(id_conferenza) on delete
        cascade,
131     unique (id_sponsor,id_conferenza) -- Vincolo di integrita': uno sponsor puo
        ' sponsorizzare una conferenza una sola volta
132 );
133
134 -- TABLE: speaker
135 create table speaker(
136     id_speaker serial primary key,
137     nome text not null,
138     cognome text not null,
139     titolo titolo_st,
140     email text not null unique,
141     id_ente integer references ente(id_ente) on delete set null
142 );
143
144 -- TABLE: programma
145 create table programma(
146     id_programma serial primary key,
147     id_sessione integer references sessione(id_sessione) on delete cascade,
148     id_keynote integer references speaker(id_speaker) on delete set null,
149     unique (id_programma, id_sessione)
150 );
151
152 -- TABLE: intervento
153 create table intervento(
154     id_intervento serial primary key,
155     titolo text not null,
156     abstract text not null,
157     inizio timestamp not null,
158     fine timestamp not null,
159     id_speaker integer references speaker(id_speaker) on delete cascade,
160     id_programma integer references programma(id_programma) on delete cascade,
161     unique (id_speaker,id_programma), -- Vincolo di integrita': uno speaker puo
        ' intervenire in una sessione una sola volta
162     check (inizio <= fine) -- Vincolo di integrita': l'intervento deve iniziare
        prima di finire
163 );
164
165 -- TABLE: intervallo
166 create table intervallo(
167     id_intervallo serial primary key,
168     tipologia intervallo_st not null,
169     inizio timestamp not null,
170     fine timestamp not null,
171     check (inizio <= fine), -- Vincolo di integrita': l'intervallo deve
        iniziare prima di finire
172     id_programma integer references programma(id_programma) on delete cascade
173 );
174
175 -- TABLE: evento
176 create table evento(
177     id_evento serial primary key,
178     tipologia text not null,
179     inizio timestamp not null,
180     fine timestamp not null,
181     check (inizio <= fine), -- Vincolo di integrita': l'evento deve iniziare
        prima di finire
182     id_programma integer references programma(id_programma) on delete cascade
183 );

```

```

184
185 -- TABLE: organizzatore_comitato
186 create table organizzatore_comitato(
187     id_organizzatore integer references organizzatore(id_organizzatore) on
delete cascade,
188     id_comitato integer references comitato(id_comitato) on delete cascade,
189     unique (id_organizzatore,id_comitato) -- Vincolo di integrita': un
organizzatore puo' far parte di un comitato una sola volta
190 );

```

6.2 Popolamento

6.3 Trigger

6.4 Funzioni e procedure

A Dizionario dei dati

Classe	Descrizione	Attributi
Comitato	Tabella che descrive i comitati che si occupano della logistica e della pianificazione delle conferenze scientifiche.	<p>id_comitato (<i>serial</i>)(<i>totale</i>): Identificatore univoco per un comitato.</p> <p>tipologia(<i>comitato_st</i>)(<i>totale</i>): Specifica il tipo di comitato (scientifico o locale).</p>
Conferenza	Tabella che descrive le conferenze scientifiche.	<p>Id_Conferenza(<i>serial</i>)(<i>totale</i>): Chiave primaria per una conferenza.</p> <p>Titolo(<i>Text</i>)(<i>totale</i>): Specifica il titolo della conferenza scientifica.</p> <p>Descrizione(<i>Text</i>)(<i>parziale</i>): Fornisce una descrizione della conferenza scientifica.</p> <p>Inizio(<i>Timestamp</i>)(<i>totale</i>): Indica l'inizio della conferenza.</p> <p>Fine (<i>Timestamp</i>)(<i>totale</i>) : Indica la fine della conferenza.</p>
Ente	Tabella delle istituzioni	<p>Id_Ente(<i>serial</i>)(<i>totale</i>): Identificatore primario di una istituzione.</p> <p>Nome(<i>Text</i>)(<i>totale</i>): Nome dell'istituzione.</p> <p>Sigla(<i>Varchar(7)</i>)(<i>totale</i>) : Sigla dell'istituzione.</p>
Evento	Descrittore degli eventi sociali presenti all'interno di una conferenza scientifica.	<p>Id_Evento(<i>Serial</i>)(<i>Totale</i>): Identificatore primario per un evento.</p> <p>Tipologia(<i>text</i>)(<i>totale</i>): Stringa descrittiva della tipologia dell'evento.</p>

Continua nella prossima pagina

Continua dalla pagina precedente

Classe	Descrizione	Attributi
		<p>Inizio(<i>Timestamp</i>)(<i>totale</i>): Indica l'inizio dell'evento.</p> <p>Fine (<i>Timestamp</i>)(<i>totale</i>) : Indica la fine dell'evento.</p>
Indirizzo	Tabella degli indirizzi per ogni sede	<p>Id_Indirizzo(<i>serial</i>)(<i>totale</i>): Chiave primaria.</p> <p>Via(<i>text</i>)(<i>parziale</i>): nome della via.</p> <p>Civico(<i>text</i>)(<i>parziale</i>): civico della sede.</p> <p>Cap(<i>char</i>(5))(<i>parziale</i>): codice di avviamento postale</p> <p>Città(<i>text</i>)(<i>parziale</i>): città della sede.</p> <p>Provincia(<i>varchar</i>(2)): provincia della città.</p> <p>Stato(<i>text</i>)(<i>parziale</i>): stato della sede.</p>
Intervallo	Descrittore degli intervalli presenti all'interno di una conferenza scientifica.	<p>Id_Intervallo(<i>Serial</i>)(<i>Totale</i>): Identificatore primario per un evento.</p> <p>Tipologia(<i>Intervallo_ST</i>)(<i>totale</i>): Specifica il tipo di intervallo (pranzo o coffee break).</p> <p>Inizio(<i>Timestamp</i>)(<i>totale</i>): Indica l'inizio dell'intervallo.</p> <p>Fine (<i>Timestamp</i>)(<i>totale</i>) : Indica la fine dell'intervallo.</p>
Intervento	Descrittore degli interventi che si tengono all'interno delle sessioni.	<p>Id_Intervento(<i>Serial</i>)(<i>totale</i>): Identificatore primario di un intervento.</p> <p>Titolo(<i>Text</i>) (<i>totale</i>): Specifica il titolo dell'intervento.</p> <p>Abstract(<i>Text</i>)(<i>parziale</i>): Fornisce una descrizione dell'intervento.</p> <p>Inizio(<i>Timestamp</i>)(<i>totale</i>): Indica l'inizio dell'intervento.</p> <p>Fine (<i>Timestamp</i>)(<i>totale</i>) : Indica la fine dell'intervento.</p>
Organizzatore	Descrittore dei membri dei comitati.	<p>Id_Organizzatore(<i>serial</i>)(<i>Totale</i>): Identificatore principale di un organizzatore.</p> <p>Nome(<i>text</i>)(<i>totale</i>): nome dell'organizzatore.</p> <p>Cognome(<i>text</i>)(<i>totale</i>): cognome dell'organizzatore.</p>

Continua nella prossima pagina

Continua dalla pagina precedente

Classe	Descrizione	Attributi
		<p>Titolo(<i>Titolo_ST</i>)(<i>parziale</i>): Titolo accademico dell'organizzatore</p> <p>Email(<i>Text</i>)(<i>Parziale</i>): Email dell'organizzatore</p>
Partecipante	Descrittore dei partecipanti delle sessioni.	<p>Id_Partecipante(<i>serial</i>)(<i>Totale</i>): Identificatore principale di un partecipante.</p> <p>Nome(<i>text</i>)(<i>totale</i>): nome dell'organizzatore.</p> <p>Cognome(<i>text</i>)(<i>totale</i>): cognome dell'organizzatore.</p> <p>Titolo(<i>Titolo_ST</i>)(<i>parziale</i>): Titolo accademico del partecipante.</p> <p>Email(<i>Text</i>)(<i>Parziale</i>): Email del partecipante.</p>
Programma	Tabella dei programmi delle sessioni.	<p>Id_Programma(<i>serial</i>)(<i>totale</i>): Identificatore principale dei programmi.</p>
Sala	Tabella delle sale di ciascuna sede.	<p>Id_sala(<i>serial</i>)(<i>totale</i>): identificatore principale di ciascuna sala.</p> <p>Nome(<i>Text</i>)(<i>totale</i>): nome della sala.</p> <p>Capienza(<i>int</i>)(<i>totale</i>): capienza della sala.</p>
Sede	Descrizione delle sedi che ospitano le conferenze	<p>Id_Sede(<i>Serial</i>)(<i>totale</i>) : Identificatore principale delle sedi.</p> <p>Nome(<i>Text</i>)(<i>totale</i>): nome della sede.</p>
Sessione	Tabella delle sessioni di ciascuna conferenza.	<p>Id_Sessione(<i>Serial</i>)(<i>total</i>): Identificatore primario di una sessione.</p> <p>Titolo(<i>Text</i>) (<i>totale</i>): Specifica il titolo della sessione.</p> <p>Inizio(<i>Timestamp</i>)(<i>totale</i>): Indica l'inizio della sessione.</p> <p>Fine (<i>Timestamp</i>)(<i>totale</i>) : Indica la fine della sessione.</p>
Speaker	Descrittore dei vari speaker delle sessioni.	<p>Id_Speaker(<i>serial</i>)(<i>Totale</i>): Identificatore principale di uno speaker.</p> <p>Nome(<i>text</i>)(<i>totale</i>): nome dello speaker.</p> <p>Cognome(<i>text</i>)(<i>totale</i>): cognome dello speaker.</p> <p>Titolo(<i>Titolo_ST</i>)(<i>parziale</i>): Titolo accademico dello speaker.</p> <p>Email(<i>Text</i>)(<i>Parziale</i>): Email dello speaker.</p>
Sponsor	Tabella degli sponsor	<p>Id_Sponsor(<i>serial</i>)(<i>totale</i>): Identificatore primario di uno sponsor.</p>

Continua nella prossima pagina

Continua dalla pagina precedente

Classe	Descrizione	Attributi
		Nome(<i>Text</i>)(<i>totale</i>): Nome dello sponsor.
Valuta	Tabella delle valute	Iso (<i>Char</i> (3))(<i>totale</i>): codice univoco internazionale delle valute. Nome(<i>text</i>)(<i>totale</i>): nome della valuta. Simbolo(<i>char</i> (1))(<i>totale</i>): simbolo della valuta.

B Dizionario delle associazioni

Associazione	Descrizione	Classi coinvolte
Appartiene_A	Rappresenta l'appartenenza di un organizzatore ad una precisa istituzione.	Organizzatore [0..*] : indica l'organizzatore che appartiene all'ente. Ente [0..1] ruolo in : indica l'ente al quale appartiene un organizzatore.
Appartiene_A	Rappresenta l'appartenenza di un partecipante ad una precisa istituzione.	Organizzatore [0..*] : indica il partecipante che appartiene ad un ente. Ente [0..1] ruolo istituzione : indica l'ente al quale appartiene un partecipante.
Appartiene_A	Rappresenta l'appartenenza di uno speaker ad una precisa istituzione.	Organizzatore [0..*] : indica lo speaker che appartiene all'ente. Ente [0..1] ruolo istituzione : indica l'ente al quale appartiene uno speaker.
Comitato_Conferenza	Ogni conferenza è legata ai comitati che ne gestiscono l'organizzazione.	Comitati [2..2] : indica i due comitati nominati per la conferenza. Conferenza [1..1] ruolo di : ogni comitato appartiene ad una sola conferenza.
Sponsorizzazione_Conferenza	Ogni conferenza ha varie sponsorizzazioni da parte degli Sponsor che contribuiscono alle spese generali.	Sponsor [0..*] Conferenza [0..*]

Continua nella pagina successiva

Continua dalla pagina precedente

Associazione	Descrizione	Classi coinvolte
Svolta_In	Specifica l'ubicazione di una conferenza in una sede.	Conferenza [0..*] Sede [1..1]
Svolta_In	Specifica l'ubicazione di una sessione in una sala.	Sessione [0..*] Sala [1..1]
Coordina	Ogni sessione ha un coordinatore.	Sessione [0..1] Organizzatore [1..1]
Sessioni_Conferenza	Ogni conferenza è composta da una o più sessioni.	Conferenza [1..1] Sessioni [0..*]
Sale_Sede	Ogni sede è composta da una o più sedi.	Sede [1..1] Sala [1..*]
Programma_Sessione	Ogni sessione ha un programma	Sessione[1..1] Programma [1..1]
Programma_Intervento	Ogni programma è un composto di vari interventi	Programma [1..1] Intervento [0..*]
Programma_Intervallo	Ogni programma è un composto di vari intervalli	Programma [1..1] Intervallo [0..*]
Programma_Evento	Ogni programma è un composto di vari eventi sociali	Programma [1..*] Evento [0..*]
Partecipante_Sessione	Ogni sessione ha vari partecipanti che partecipano a varie sessioni	Sessione [0..*] Partecipante [0..*]
Speaker_Intervento	Ogni intervento ha un suo speaker che può effettuare vari interventi	Intervento [0..*] Speaker [1..1]
Membro_Comitato	Ogni comitato è composto da vari organizzatori che appartengono a vari comitati	Organizzatore [0..*]

Continua nella pagina successiva

Continua dalla pagina precedente

Associazione	Descrizione	Classi coinvolte
		Comitato [0..*]

C Dizionario dei vincoli

Vincolo	Tipo	Descrizione
---------	------	-------------

Elenco delle figure

1	Schema concettuale del problema	17
2	Ristrutturazione dello schema concettuale	18
3	Schema logico	19

Elenco delle tabelle

1	Entità del problema	3
---	-------------------------------	---

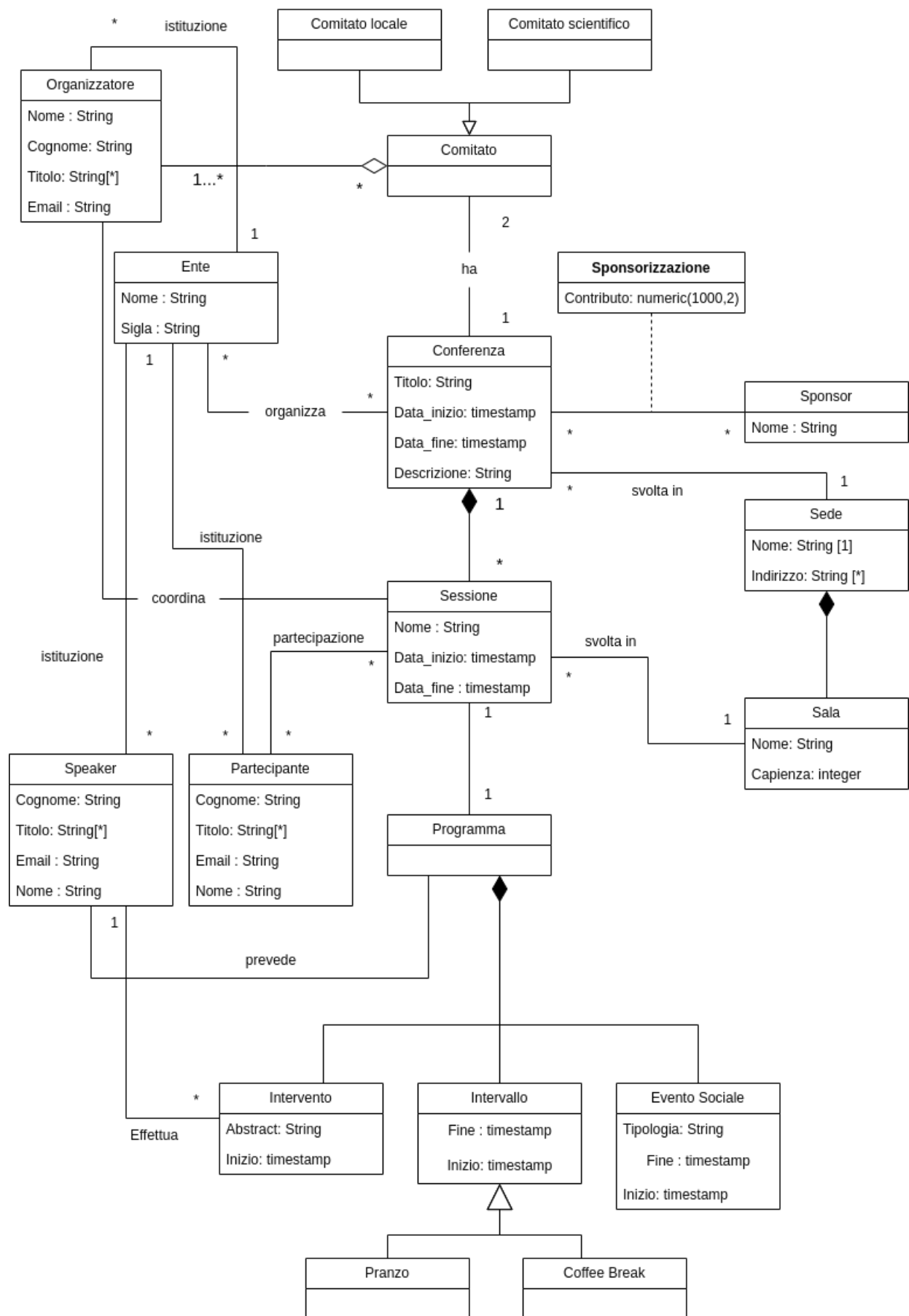


Figura 1: Schema concettuale del problema

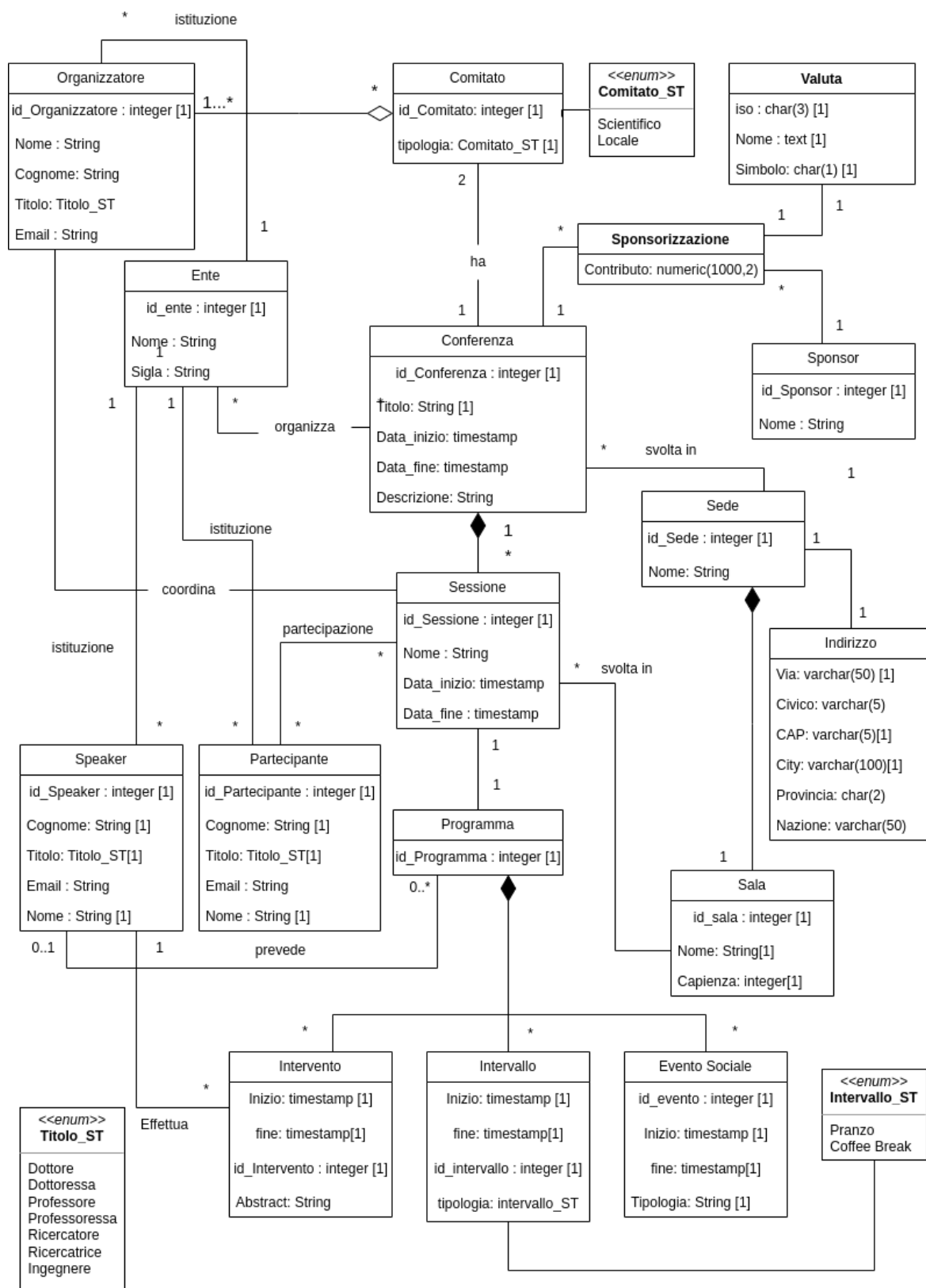


Figura 2: Ristrutturazione dello schema concettuale

Figura 3: Schema logico

