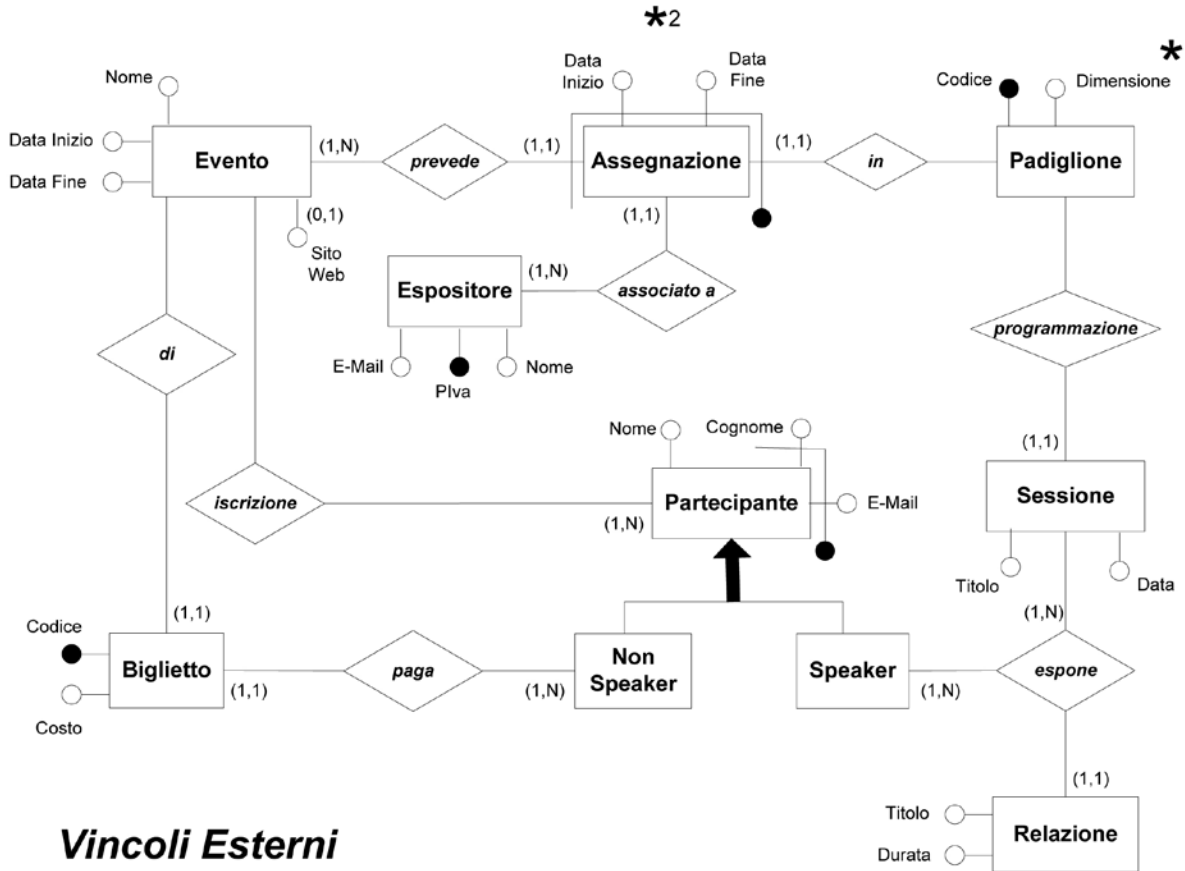


Esame di Basi di Dati, **SOLUZIONE APPELLO 09/06/2009**

1. Si effettui la progettazione concettuale della base di dati secondo la specifica che segue fornendo un diagramma ER. Un'azienda che gestisce gli eventi di uno spazio di fiera vuole progettare una base di dati per la memorizzazione delle informazioni di suo interesse. L'azienda ha il compito di gestire tutti gli eventi che sono organizzati nello spazio di fiera. Questi eventi sono caratterizzati da un nome, dalla durata dell'evento e, se disponibile, dal sito web dell'evento. L'evento si svolge nei padiglioni dello spazio di fiera. E' di interesse memorizzare quali padiglioni sono occupati in quali giorni. Inoltre i padiglioni, oltre ad essere identificati mediante un codice alfanumerico, sono classificabili in padiglioni grandi, medi e piccoli. Ogni padiglione ha associato un espositore, di cui interessa il nome, la partita iva, ed il riferimento ad una persona di contatto (e-mail). Il padiglione ha un programma delle giornate in cui si svolge l'evento che risulta organizzato in sessioni. A ciascuna sessione è associato un titolo, la giornata in cui si svolge e un insieme di speaker, di cui interessa nome, cognome, e-mail, oltre che titolo e durata della relazione che dovranno esporre. Si noti che uno speaker può effettuare più di una relazione all'interno dello stesso evento. I partecipanti all'evento devono iscriversi fornendo i loro dati anagrafici (nome, cognome, e-mail). Anche gli speaker devono iscriversi all'evento. Tutti i partecipanti tranne gli speaker devono poi pagare un biglietto di iscrizione caratterizzato da un codice identificativo ed un costo.

1.1 SCHEMA E-R



Vincoli Esterni

*

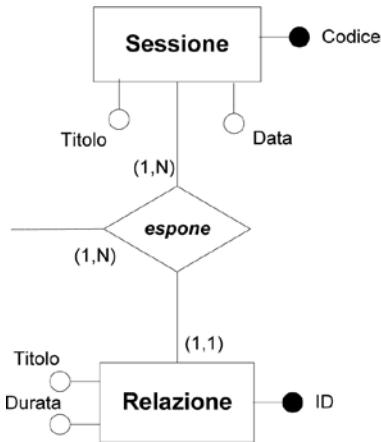
L'attributo *Dimensione* dell'entità **Padiglione** può assumere solo i seguenti tre valori : grande, medio, piccolo

*2

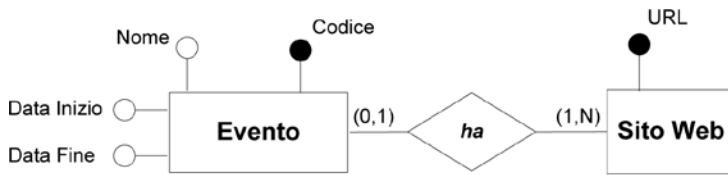
Il periodo in cui un padiglione viene assegnato ad un evento deve essere compatibile con il periodo in cui l'evento stesso si svolge (in termini di data inizio e data fine dell'evento)

2. Si effettui la prog. logica del diagramma ER realizzato con riferimento alla domanda 1. Si motivino le scelte di progetto effettuate con particolare riferimento a scelte connesse alla presenza di valori nulli, fornendo anche una discussione generale sul trattamento dei valori nulli in una base di dati.

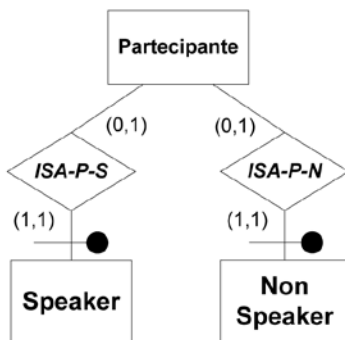
2.1 ESTRATTO dello SCHEMA E-R RISTRUTTURATO



Per ogni entità è necessario individuare un identificatore principale. E' possibile introdurre un ulteriore attributo (un **codice**, i cui valori sono speciali ed hanno l'unico scopo di identificare le istanze dell'entità) per tutte quelle entità che non dispongono di un identificatore principale.



Se si privilegia una scelta senza valori NULL, l'attributo opzionale *Sito Web* può essere trasformato in un'entità.



Vincolo di Generalizzazione :

Ogni istanza di **Partecipante** partecipa ad **ISA-P-S** o ad **ISA-P-N**, ma non ad entrambi.

2.2 SCHEMA RELAZIONALE

Evento(Codice,Nome,DataInizio,DataFine)

inclusione : Evento(Codice) \subseteq Assegnazione(Evento)

ha(Evento,SitoWeb)

foreign key : ha(Evento) \subseteq Evento(Codice)

foreign key : ha(SitoWeb) \subseteq SitoWeb(URL)

chiave : Evento

SitoWeb(URL)

inclusione : SitoWeb(URL) \subseteq ha(SitoWeb)

Assegnazione(Evento,Padiglione,DataInizio,DataFine,Espositore)

foreign key : Assegnazione(Evento) \subseteq Evento(Codice)

foreign key : Assegnazione(Padiglione) \subseteq Padiglione(Codice)

foreign key : Assegnazione(Espositore) \subseteq Espositore(PIva)

Espositore(PIva,Nome,E-mail)

inclusione : Espositore(PIva) \subseteq Assegnazione(Espositore)

Padiglione(Codice,Dimensione)

Sessione(Codice,Padiglione,Titolo,Data)

foreign key : Sessione(Padiglione) \subseteq Padiglione(Codice)

inclusione : Sessione(Codice) \subseteq Relazione(Sessione)

Relazione(ID,Sessione,SpeakerCognome,SpeakerE-Mail,Durata,Titolo)

foreign key : Relazione(Sessione) \subseteq Sessione(Codice)

foreign key : Relazione(SpeakerCognome,SpeakerE-Mail) \subseteq Speaker(Cognome,E-Mail)

Partecipante(Cognome,E-Mail,Nome)

inclusione : Partecipante(Cognome,E-Mail) \subseteq Iscrizione(PartCognome,PartE-Mail)

Iscrizione(Evento,PartCognome,PartE-Mail)

foreign key : Iscrizione(Evento) \subseteq Evento(Codice)

foreign key : Iscrizione(PartCognome,PartE-Mail) \subseteq Partecipante(Cognome, E-Mail)

Speaker(Cognome,E-Mail)

foreign key : Speaker(Cognome,E-Mail) \subseteq Partecipante(Cognome,E-Mail)

inclusione : Speaker(Cognome,E-Mail) \subseteq Relazione(SpeakerCognome,SpeakerE-Mail)

NonSpeaker(Cognome,E-Mail)

foreign key : NonSpeaker(Cognome,E-Mail) \subseteq Partecipante(Cognome,E-Mail)

inclusione : NonSpeaker(Cognome,E-Mail) \subseteq

Biglietto(NonSpeakerCognome,NonSpeakerE-Mail)

Biglietto(Codice,Evento,NonSpeakerCognome,NonSpeakerE-Mail,Costo)

foreign key : Biglietto(NonSpeakerCognome,NonSpeakerE-Mail) \subseteq

NonSpeaker(Cognome,E-Mail)

foreign key : Biglietto(Evento) \subseteq Evento(Codice)

I vincoli di generalizzazione dello schema ristrutturato devono essere espressi in forma insiemistica nello schema relazionale :

- $\text{NonSpeaker}[\text{Cognome}, \text{E-Mail}] \cap \text{Speaker}[\text{Cognome}, \text{E-Mail}] = \emptyset$
- $\text{Partecipante}[\text{Cognome}, \text{E-Mail}] \subseteq \text{NonSpeaker}[\text{Cognome}, \text{E-Mail}] \cup \text{Speaker}[\text{Cognome}, \text{E-Mail}]$

Rimangono ovviamente validi i vincoli esterni espressi in fase di Progettazione Concettuale.

3. Dato il seguente schema:

CONTRIBUENTE(CodFiscale, Nome, Via, Città)

DICHIARAZIONE(CodDichiarazione, Reddito)

PRESENTA(CodFiscale, CodDichiarazione, Data)

3.1 – Si fornisca il DDL della definizione delle tabelle.

```
CREATE TABLE Contribuente (  
CodFiscale CHAR(16) PRIMARY KEY,  
Nome VARCHAR(20),  
Via VARCHAR(20),  
Città VARCHAR(20)  
)
```

```
CREATE TABLE Dichiarazione (  
CodDichiarazione CHAR(5) PRIMARY KEY,  
Reddito REAL  
)
```

```
CREATE TABLE Presenta (  
CodFiscale CHAR(16) REFERENCES Contribuente(CodFiscale),  
CodDichiarazione CHAR(5) REFERENCES Dichiarazione(CodDichiarazione),  
Data DATE,  
PRIMARY KEY (CodFiscale, CodDichiarazione)  
)
```

3.2 – Scrivere un'interrogazione SQL che restituisca il codice fiscale ed il reddito dei contribuenti di Roma dal 1990 in poi.

```
SELECT C.CodFiscale, D.Reddito  
FROM Contribuente C, Dichiarazione D, Presenta P  
WHERE C.CodFiscale=P.CodFiscale  
AND D.CodDichiarazione=P.CodDichiarazione  
AND P.Data>'1/1/1990'
```

3.3 – Scrivere un'interrogazione SQL che restituisca il codice fiscale, il nome e la media dei redditi dichiarati dal 1990 in poi, per i contribuenti tali che il massimo reddito da loro dichiarato, dal 1990 in poi, sia superiore alla media dei redditi calcolata su tutte le dichiarazioni nel database.

```
SELECT C.CodFiscale, C.Nome, AVG(Reddito)
FROM Contribuente C, Dichiarazione D, Presenta P
WHERE C.CodFiscale=P.CodFiscale
      AND D.CodDichiarazione=P.CodDichiarazione AND P.Data>'1/1/1990'
GROUP BY C.CodFiscale, C.Nome
HAVING MAX(D.Reddito)>(SELECT AVG(Reddito) FROM Dichiarazione)
```