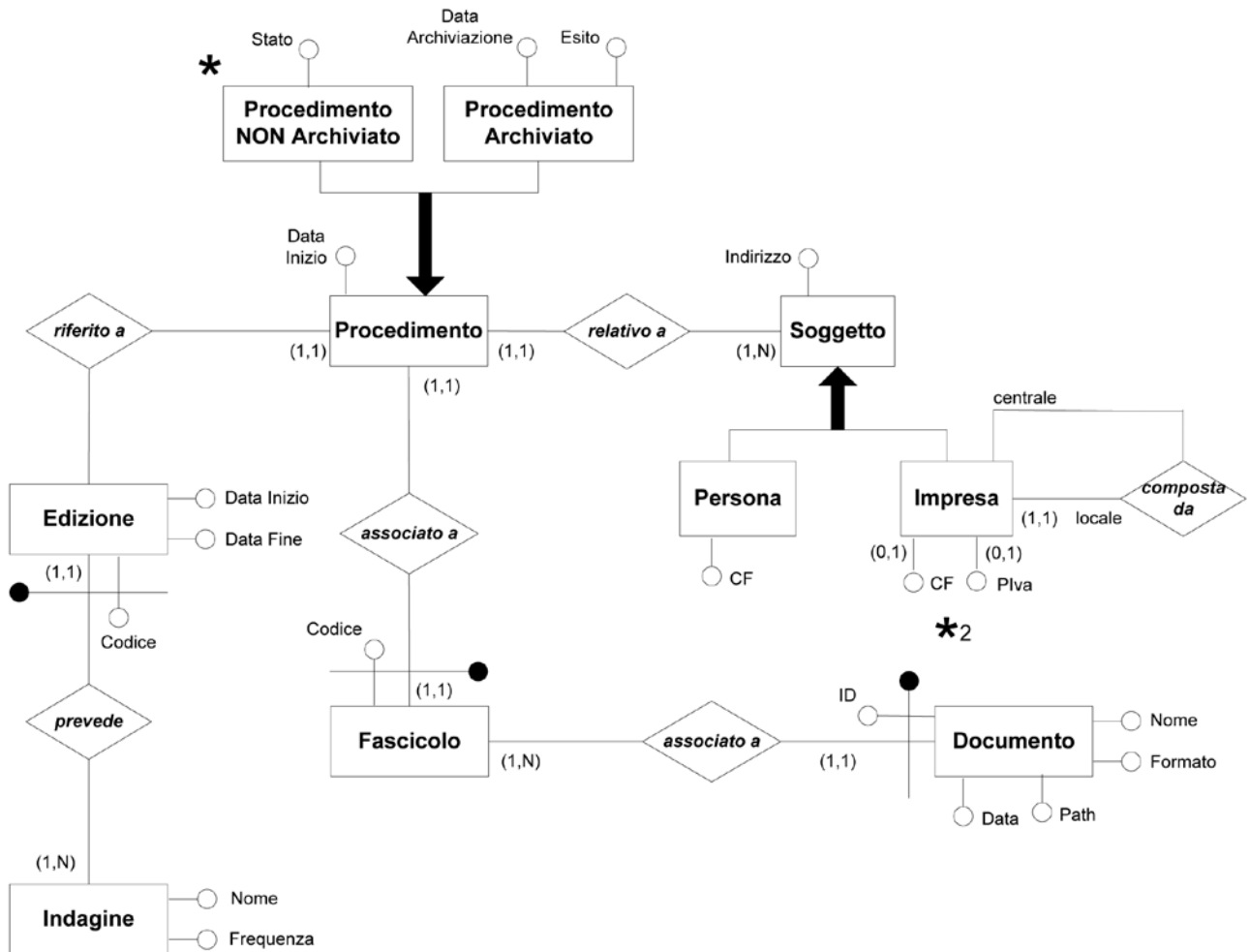


SOLUZIONE APPELLO 14/06/2011

1. Si effettui la progettazione concettuale della base di dati secondo la specifica che segue fornendo un diagramma ER. Si vuole realizzare una base di dati che gestisca procedimenti sanzionatori nel contesto di rilevazioni statistiche ufficiali di carattere nazionale. Per alcune rilevazioni statistiche ufficiali esiste, infatti, l'*obbligo di risposta* da parte dei soggetti contattati per la conduzione delle rilevazioni. Qualora il soggetto contattato non risponda al questionario inviatogli, dopo un prefissato intervallo di tempo, ha inizio un procedimento sanzionatorio che consta di due fasi principali: invio della diffida al soggetto non rispondente e, qualora tale soggetto continui ad essere inadempiente (cioè non risponda al questionario), invio della sanzione che il soggetto stesso dovrà pagare. I soggetti possono essere persone fisiche o imprese. Delle persone fisiche interessa memorizzare il codice fiscale, delle imprese il codice fiscale o la partita iva in maniera alternativa. Inoltre è di interesse l'indirizzo cui il soggetto è contattabile. Si noti che le imprese possono prevedere delle unità locali, ovvero l'impresa si articola secondo una struttura che consiste di un'impresa centrale ed eventualmente di un insieme di imprese "periferiche". Un procedimento viene avviato in relazione alla non-risposta ad una specifica edizione di un'indagine. Ogni indagine è caratterizzata da un nome (es. Forze di lavoro), da una frequenza con cui le sue edizioni occorrono (es. trimestrale) e dalle specifiche edizioni che sono occorse (es. primo trimestre 2011). Le edizioni, che hanno un codice univoco nell'ambito dell'indagine in cui sono svolte, hanno una data di inizio ed una data di fine che caratterizzano l'inizio e la fine della rilevazione sul campo dei dati oggetto dell'indagine. Nell'ambito di un procedimento è prodotto un insieme di documenti che costituisce il fascicolo del procedimento. Un fascicolo ha un codice che lo identifica nell'ambito del procedimento a cui è legato. I documenti, che dispongono di un ID univoco nell'ambito del fascicolo in cui sono redatti, sono rappresentati da un nome, un tipo, una data di produzione e dal *path* relativo al file cui sono associati. Del procedimento, oltre alle informazioni necessarie a desumere il suo avanzamento, interessa memorizzare la data di inizio e, qualora sia stato archiviato, l'esito della archiviazione (ad esempio archiviato perché il soggetto ha risposto) e la data di archiviazione.

1.1 SCHEMA E-R



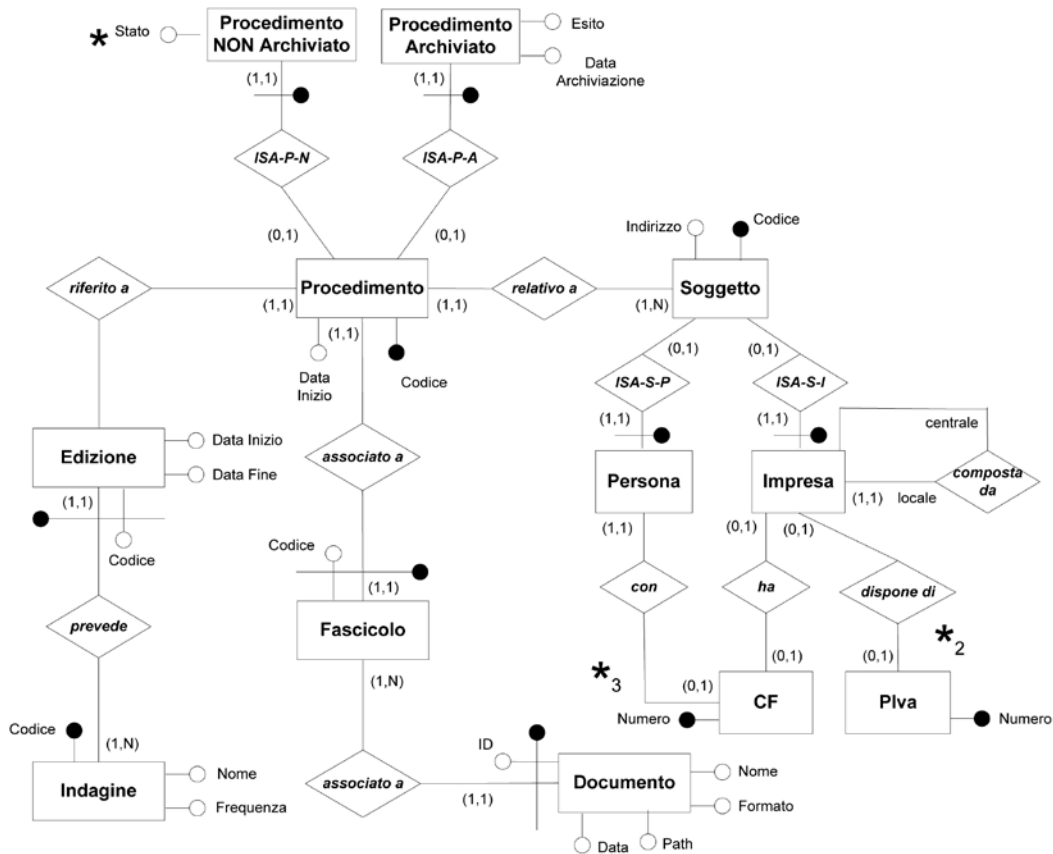
L'attributo *Stato* dell'entità **ProcedimentoNonArchiviato** può assumere solo i seguenti tre valori : non rispondente, diffidato, sanzionato

***2**

Ogni istanza dell'entità **Impresa** dovrà assumere un valore esclusivamente o per l'attributo **CF**, o per l'attributo **Piva**

2. Si effettui la prog. logica del diagramma ER realizzato con riferimento alla domanda 1. Si motivino le scelte di progetto effettuate.

2.1 SCHEMA E-R RISTRUTTURATO



* L'attributo *Stato* dell'entità **ProcedimentoNonArchiviato** può assumere solo i seguenti tre valori : non rispondente, diffidato, sanzionato

*2 Ogni istanza di **Impresa** partecipa alla relazione "ha" oppure alla relazione "dispone di", ma non ad entrambe.

*3 Ogni istanza di **CF** partecipa alla relazione "ha" oppure alla relazione "con", ma non ad entrambe.

Vincoli di Generalizzazione :

- Ogni istanza di **Procedimento** partecipa ad "ISA-P-N" oppure ad "ISA-P-A", ma non ad entrambi.
- Ogni istanza di **Soggetto** partecipa ad "ISA-S-P" oppure ad "ISA-S-I", ma non ad entrambi.

NOTE sulla RISTRUTTURAZIONE

Le scelte effettuate nella fase di ristrutturazione hanno avuto come obiettivo quello di evitare la presenza di valori NULL nella base di dati. Ciò ha comportato una traduzione "ridondante" delle generalizzazioni (in cui sia l'entità padre che le entità figlie sono state mantenute nello schema) e la traduzione degli attributi con cardinalità (0,1) in nuove entità specifiche. Notare che quest'ultimo aspetto ha comportato la modifica di uno dei vincoli esterni (*2) e l'aggiunta di un nuovo vincolo esterno (*3) rispetto a quelli definiti in fase di Progettazione Concettuale.

Inoltre, dato che per ogni entità è necessario individuare un identificatore principale, è stato introdotto 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 ne dispongono.

2.2 SCHEMA RELAZIONALE

Procedimento(Codice,Soggetto,Edizione,Indagine,DataInizio)

foreign key : Procedimento(Soggetto) \subseteq Soggetto(Codice)

foreign key : Procedimento(Edizione,Indagine) \subseteq Edizione(Codice,Indagine)

inclusione : Procedimento(Codice) \subseteq Fascicolo(Procedimento)

ProcedimentoNonArchiviato(Codice,Stato)

foreign key : ProcedimentoNonArchiviato (Codice) \subseteq Procedimento(Codice)

ProcedimentoArchiviato(Codice,Esito,DataArchiviazione)

foreign key : ProcedimentoArchiviato(Codice) \subseteq Procedimento(Codice)

Indagine(Codice,Nome,Frequenza)

inclusione : Indagine(Codice) \subseteq Edizione(Indagine)

Edizione(Codice,Indagine,DataInizio,DataFine)

foreign key : Edizione(Indagine) \subseteq Indagine(Codice)

Fascicolo(Codice,Procedimento)

inclusione : Fascicolo(Codice) \subseteq Documento(Fascicolo)

foreign key : Fascicolo(Procedimento) \subseteq Procedimento(Codice)

chiave : Procedimento

Documento(ID,Fascicolo,Data,Path,Nome,Formato)

foreign key : Documento(Fascicolo) \subseteq Fascicolo(Codice)

Soggetto(Codice,Indirizzo)

inclusione : Soggetto(Codice) \subseteq Procedimento(Soggetto)

Persona(Codice,CF)

foreign key : Persona(Codice) \subseteq Soggetto(Codice)

foreign key : Persona(CF) \subseteq CF(Numero)

chiave : CF

ImpresaLocale(Codice,CodiceCentrale)

foreign key : Impresa(Codice) \subseteq Soggetto(Codice)

foreign key : Impresa(CodiceCentrale) \subseteq ImpresaCentrale(Codice)

ImpresaCentrale(Codice)

foreign key : Impresa(Codice) \subseteq Soggetto(Codice)

CF(Numero)

PIva(Numero)

ha(CF,Impresa)

foreign key : ha(CF) \subseteq CF(Numero)

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

chiavi : CF, Impresa

disponeDi(PIva,Impresa)

foreign key : disponeDi(PIva) \subseteq PIva(Numero)

foreign key : disponeDi (Impresa) \subseteq Impresa(Codice)

chiavi : PIva, Impresa

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

- $\text{ProcedimentoNonArchiviato}[\text{Codice}] \cap \text{ProcedimentoNonArchiviato}[\text{Codice}] = \emptyset$
- $\text{Procedimento}[\text{Codice}] \subseteq \text{ProcedimentoNonArchiviato}[\text{Codice}] \cup \text{ProcedimentoArchiviato}[\text{Codice}]$
- $\text{Persona}[\text{Codice}] \cap \text{Impresa}[\text{Codice}] = \emptyset$
- $\text{Soggetto}[\text{Codice}] \subseteq \text{Persona}[\text{Codice}] \cup \text{Impresa}[\text{Codice}]$

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

3. Dato il seguente schema:

```
Delegato(IdDelegato, Nome, Cognome, Organizzazione)
Partecipa(IdDelegato, IdMeeting)
Meeting(IdMeeting, Citta, Nazione, DataInizio, DataFine)
```

3.1 – Scrivere un'interrogazione SQL che restituisca Nome e Cognome dei delegati che hanno partecipato solo a meeting non italiani.

```
SELECT Nome,Cognome
FROM Delegato
WHERE IdDelegato not in (SELECT IdDelegato
                        FROM Delegato, Partecipa,Meeting
                        WHERE (Delegato.IdDelegato=Partecipa.IdDelegato) AND
                        (Partecipa.IdMeeting=Meeting.IdMeeting) AND Nazione="Italia")
AND IdDelegato in (SELECT IdDelegato
                  FROM Delegato, Partecipa,Meeting
                  WHERE (Delegato.IdDelegato=Partecipa.IdDelegato)
                  AND (Partecipa.IdMeeting=Meeting.IdMeeting) )
```

3.2 – Scrivere un'interrogazione SQL che restituisca, per ogni organizzazione, il numero totale dei delegati che hanno partecipato a tutti i meeting.

```
SELECT Delegato.Organizzazione, count(*)
FROM Delegato, (SELECT IDDelegato
                FROM Delegato
                WHERE NOT EXISTS ((SELECT IdMeeting
                                FROM meeting)
                                EXCEPT
                                (SELECT IdMeeting
                                FROM Partecipa
                                WHERE Partecipa.IdDelegato =
                                Delegato.IdDelegato)) ) AS TEMP
WHERE TEMP.IdDelegato= Delegato.IdDelegato
GROUP BY Delegato.Organizzazione
```