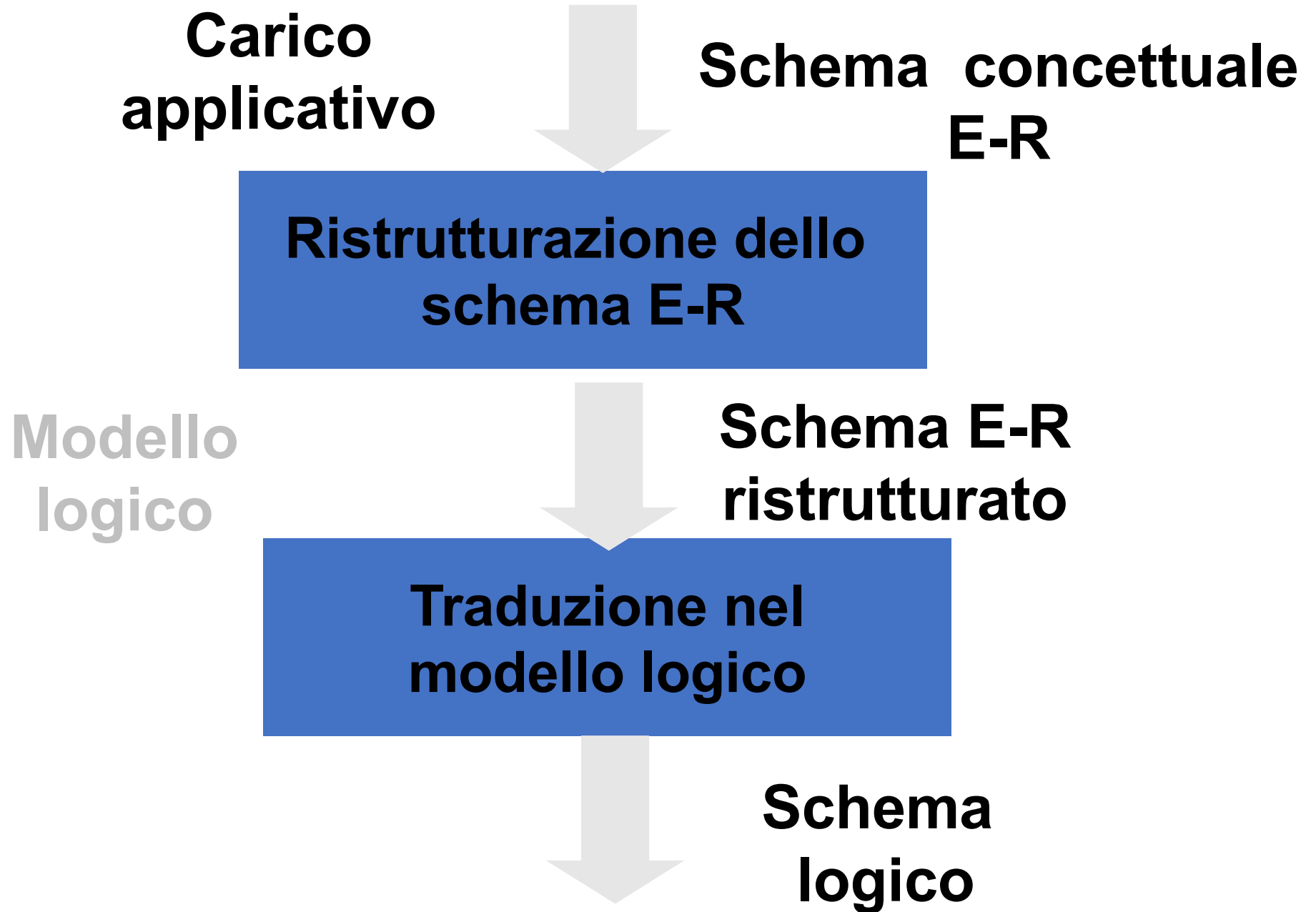
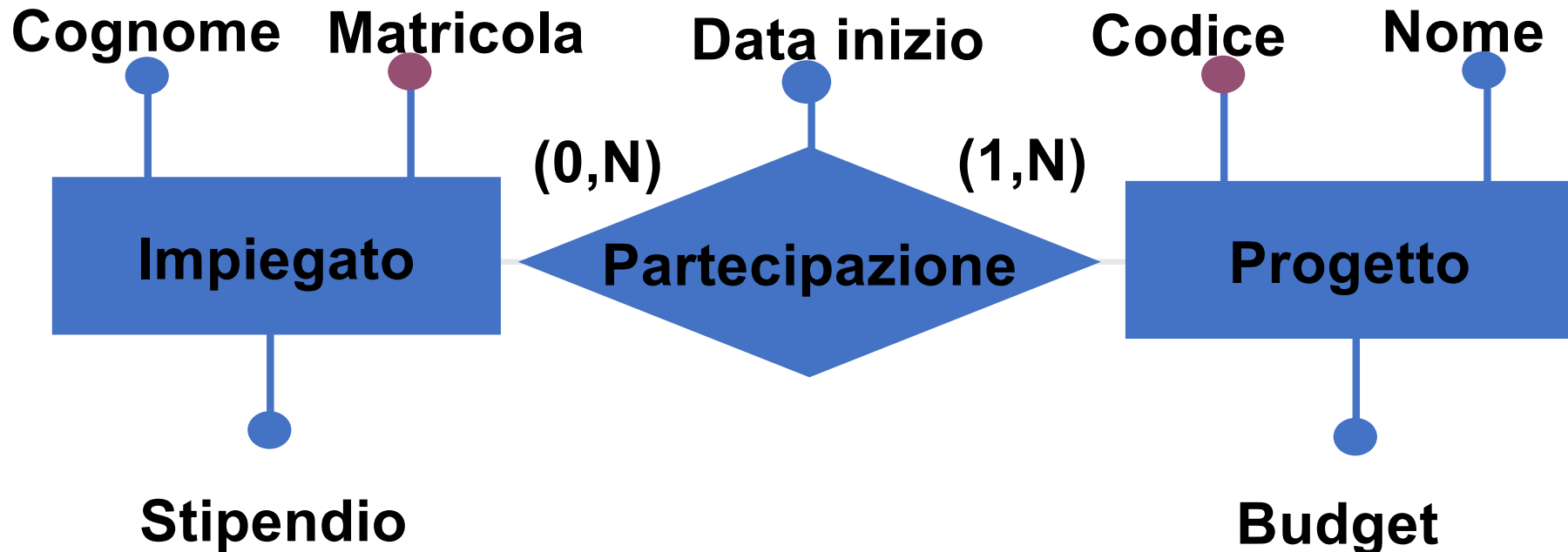




Traduzione verso il modello relazionale

- idea di base:
 - le entità diventano relazioni sugli stessi attributi
 - le relationship diventano relazioni sugli identificatori delle entità coinvolte (più gli attributi propri)

Entità e relationship molti a molti



Impiegato(Matricola, Cognome, Stipendio)

Progetto(Codice, Nome, Budget)

Partecipazione(Matricola, Codice, DataInizio)

Entità e relationship molti a molti

Impiegato(Matricola, Cognome, Stipendio)

Progetto(Codice, Nome, Budget)

Partecipazione(Matricola, Codice, DataInizio)

- con vincoli di integrità referenziale fra
 - **Matricola** in **Partecipazione** e (la chiave di) **Impiegato**
 - **Codice** in **Partecipazione** e (la chiave di) **Progetto**

**Nomi più espressivi per gli attributi
della chiave della relazione che
rappresenta la relationship**

Impiegato(Matricola, Cognome, Stipendio)

Progetto(Codice, Nome, Budget)

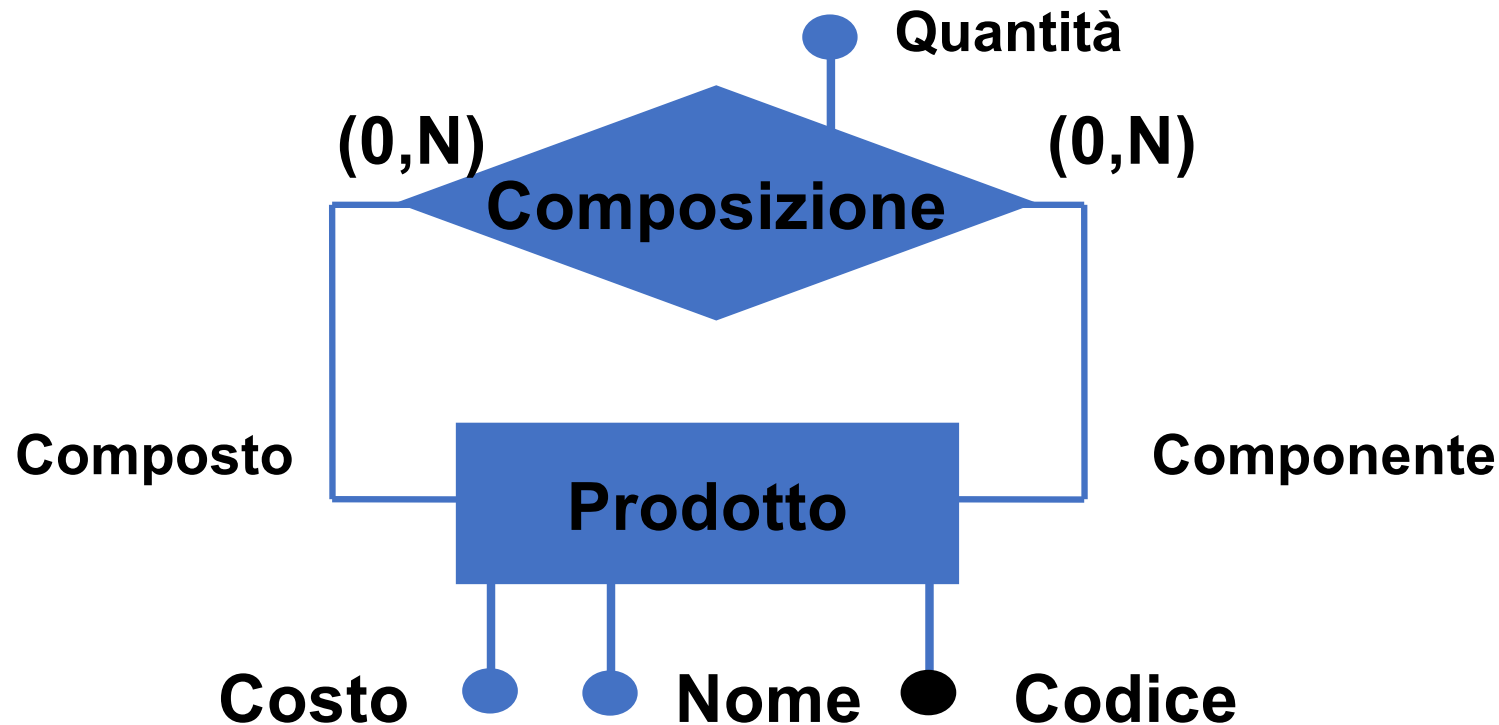
Partecipazione(Matricola, Codice, DataInizio)

Partecipazione(Impiegato, Progetto, DataInizio)

Nota

- La traduzione non riesce a tener conto delle cardinalità minime delle relationship molti a molti (se non con vincoli di CHECK complessi e poco usati)

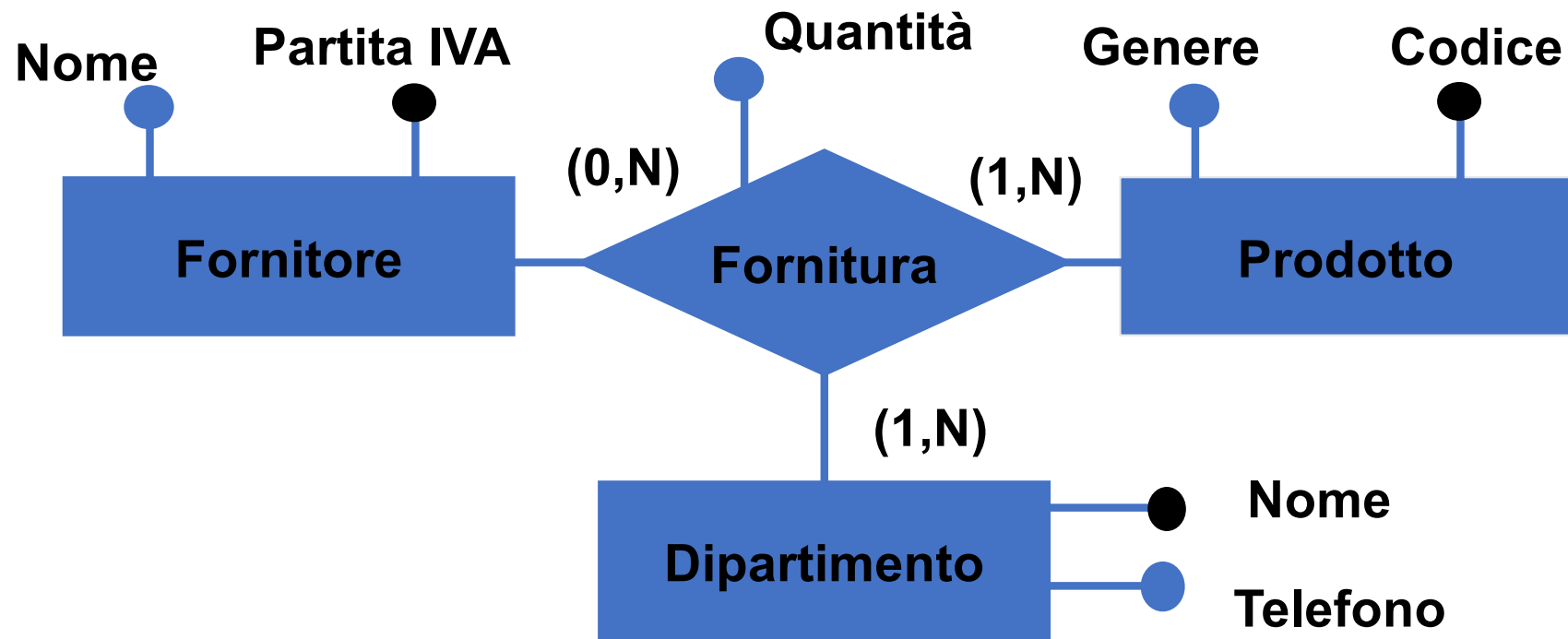
Relationship ricorsive



Prodotto(Codice, Nome, Costo)

Composizione(Composto, Componente, Quantità)

Relationship n-arie



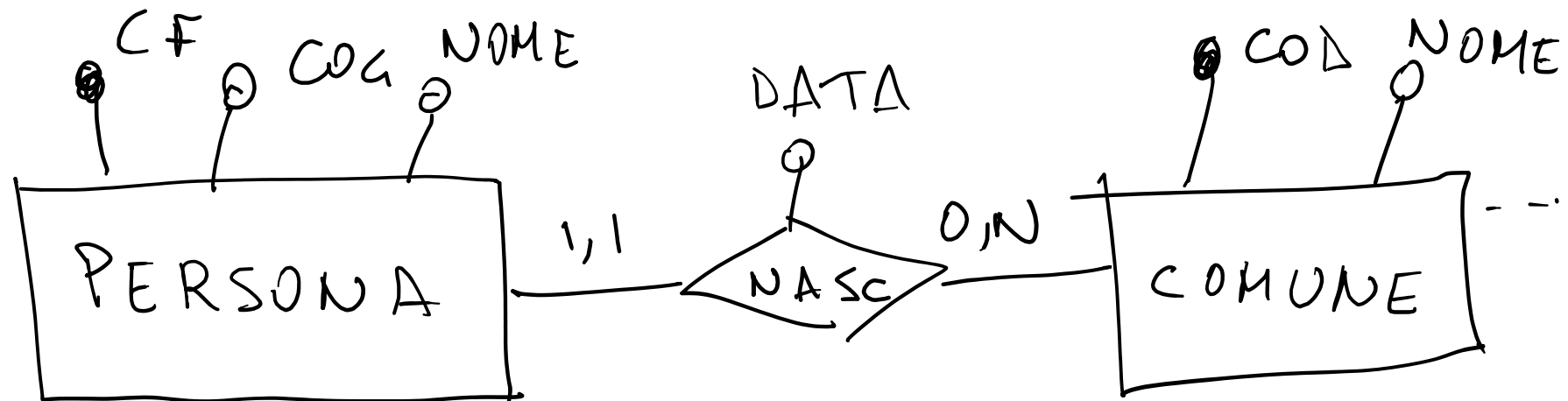
Fornitore(PartitaIVA, Nome)

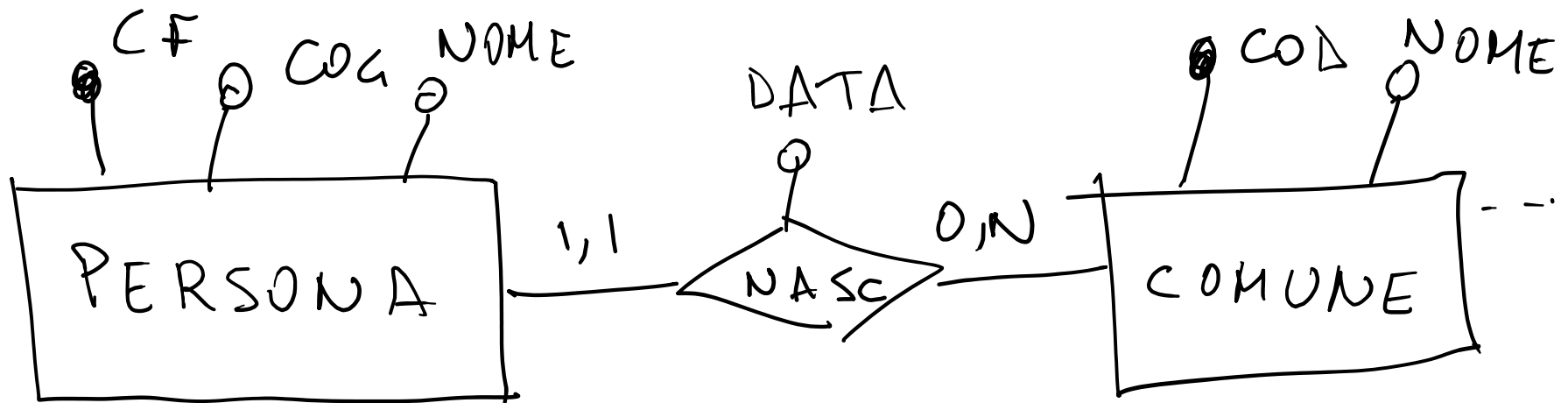
Prodotto(Codice, Genere)

Dipartimento(Nome, Telefono)

Fornitura(Fornitore, Prodotto, Dipartimento, Quantità)

Relationship uno a molti

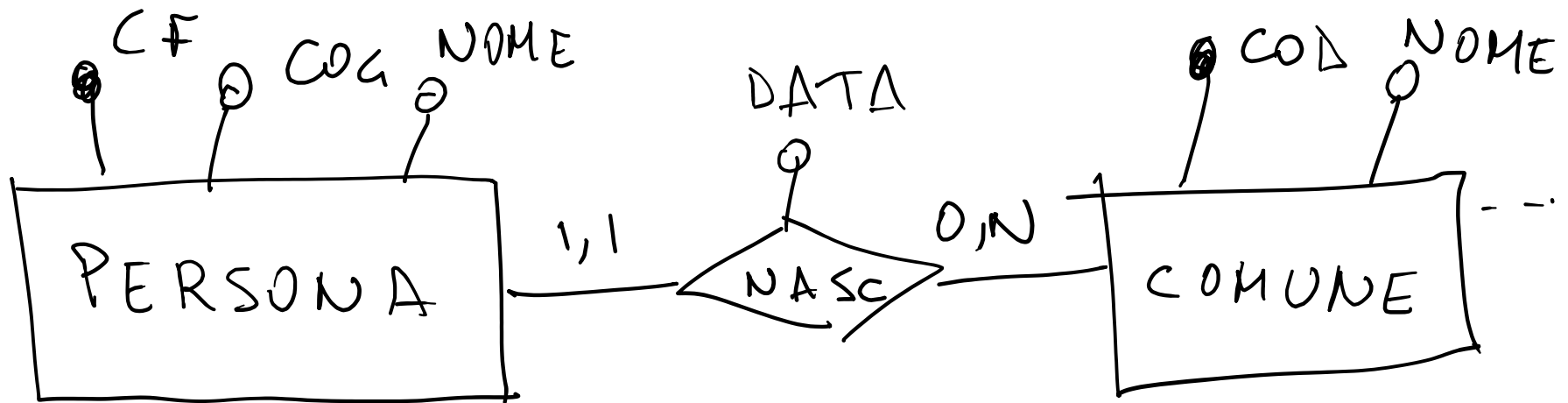




PERSONA (CF, COG, NOME)

COMUNE (COD, NOME, ..)

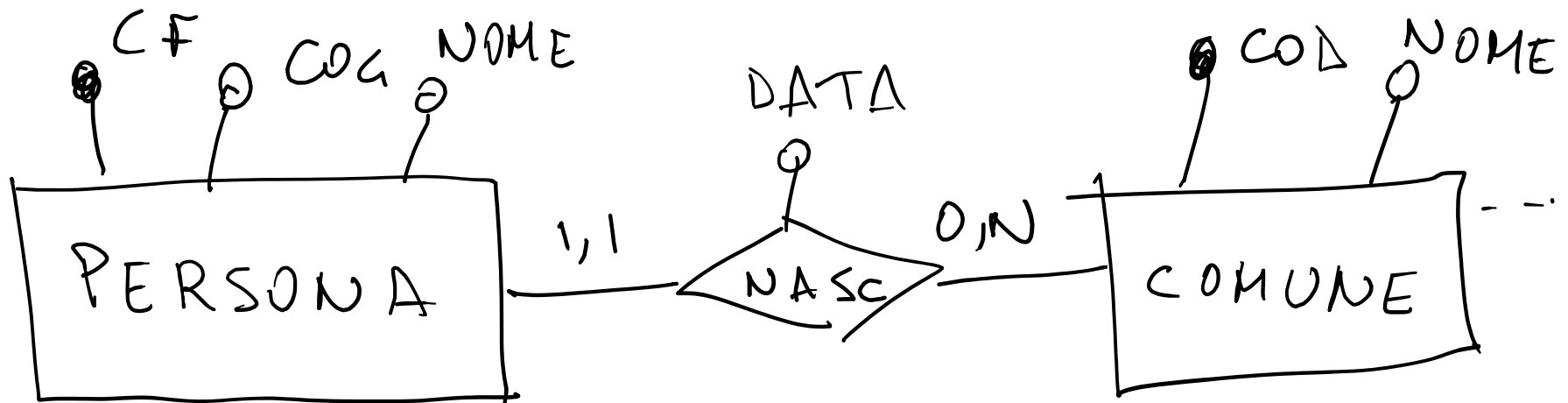
NASCITA (PERSONA, COMUNE, DATA)



PERSONA (CF, COG, NOME)

COMUNE (COD, NOME, ...)

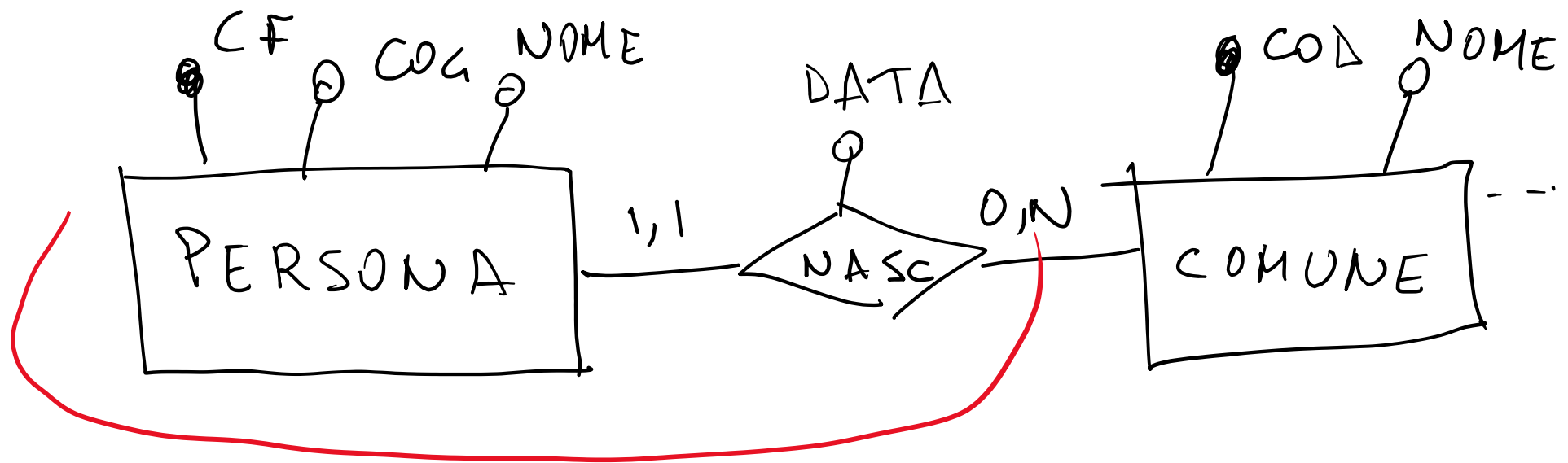
NASCITA (PERSONA, COMUNE, DATA)



PERSONA (CF, COG, NOME)

COMUNE (COD, NOME, ...)

NASCITA (PERSONA, COMUNE, DATA)
CF



PERSONA (CF, COG, NOME, COMUNE, DATA)

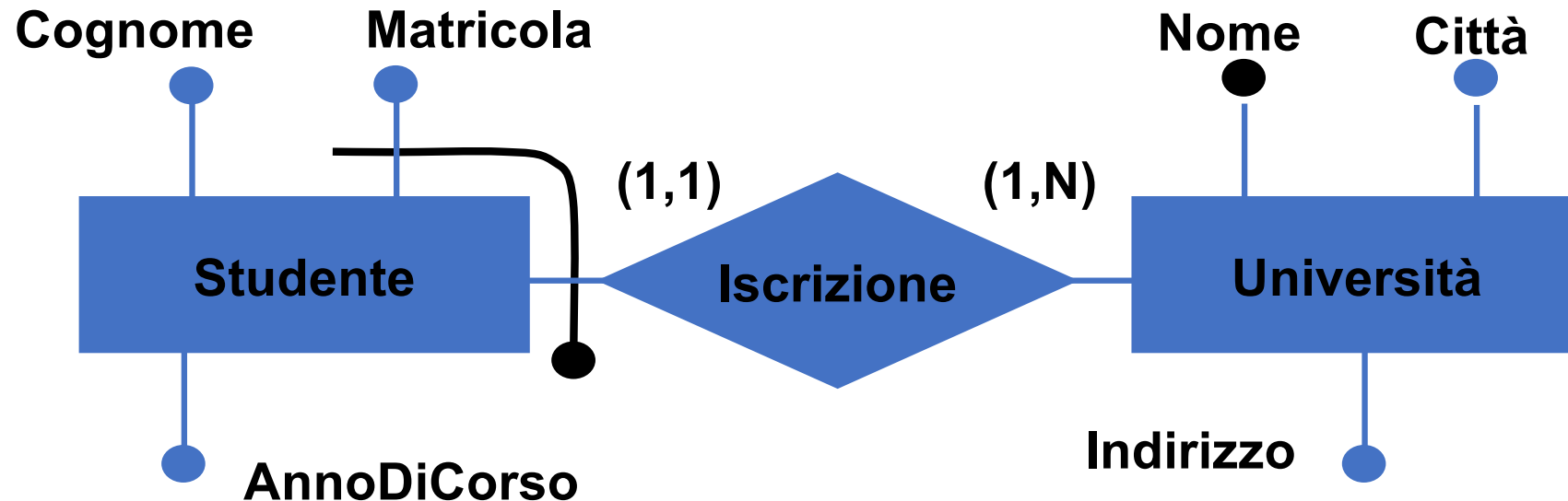
COMUNE (COD, NOME, ...)

~~NASCITA (PERSONA, COMUNE, DATA)~~
~~CF~~

Nota

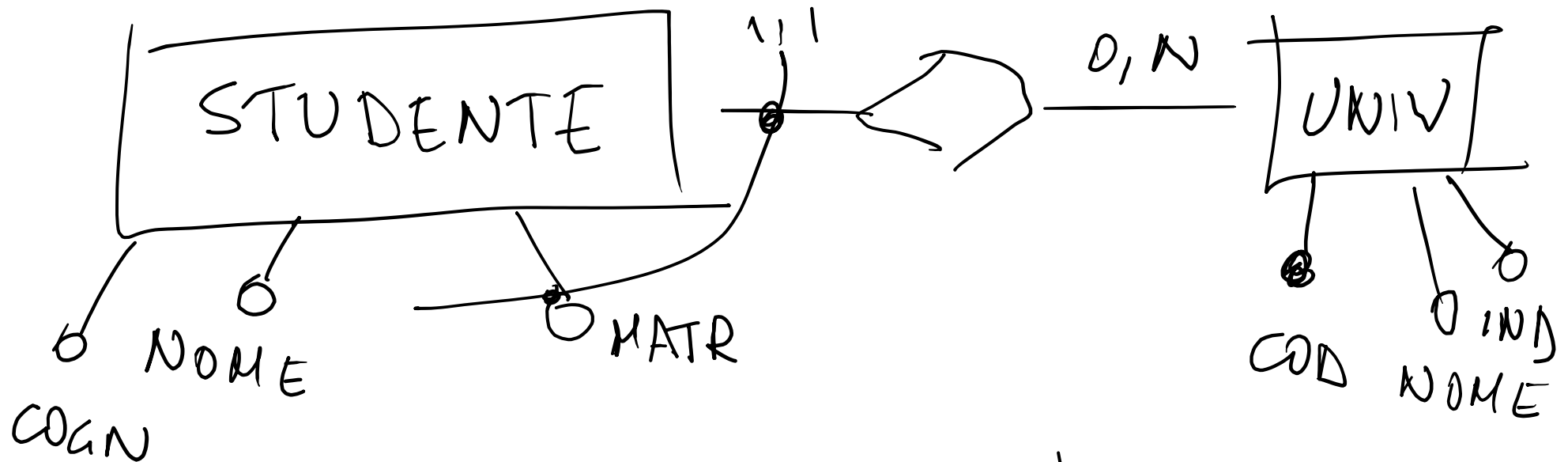
- La traduzione riesce a rappresentare efficacemente la cardinalità minima della partecipazione che ha 1 come cardinalità massima:
 - 0 : valore nullo ammesso
 - 1 : valore nullo non ammesso

Entità con identificazione esterna



Studente(Matricola, Università, Cognome, AnnoDiCorso)
Università(Nome, Città, Indirizzo)

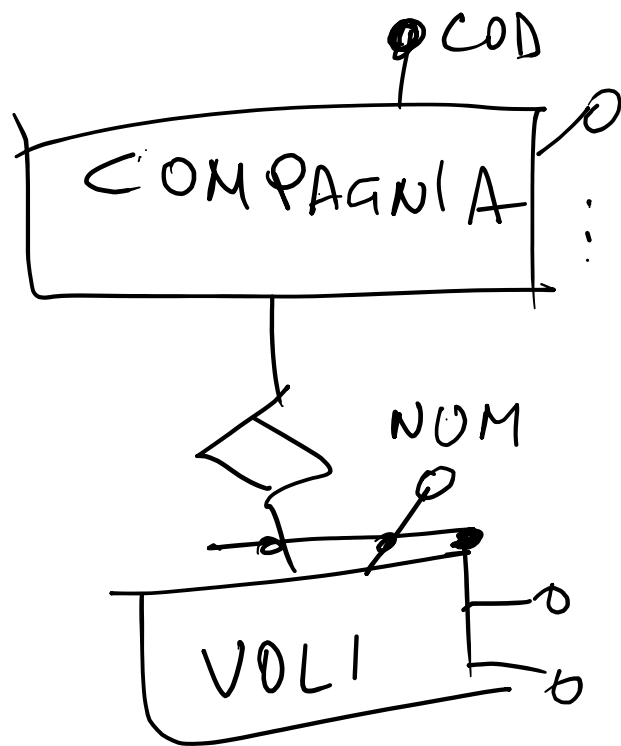
- **con vincolo ...**



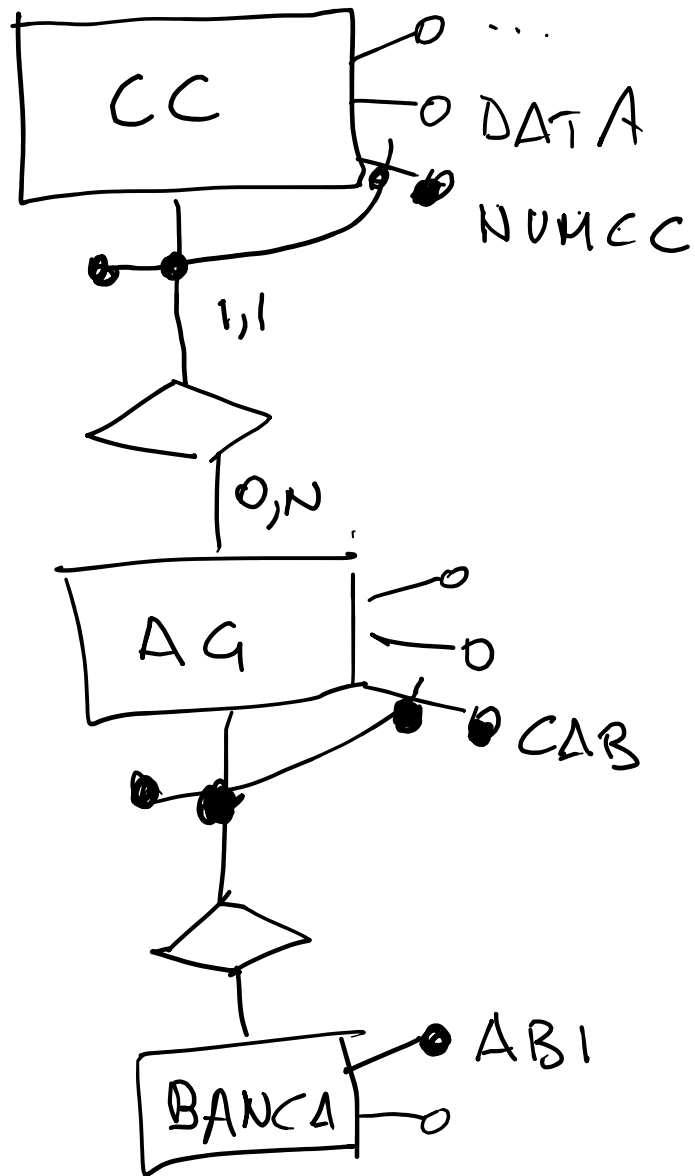
UNIV (COD, NOME, IND)
 STUDENTE (MATR, UNIV, NOME, COGNOME)

Esempi

- Voli e coordinate bancarie, vediamo alla lavagna



$VOL1(\underbrace{COMP}, \underbrace{NUM}, \dots)$



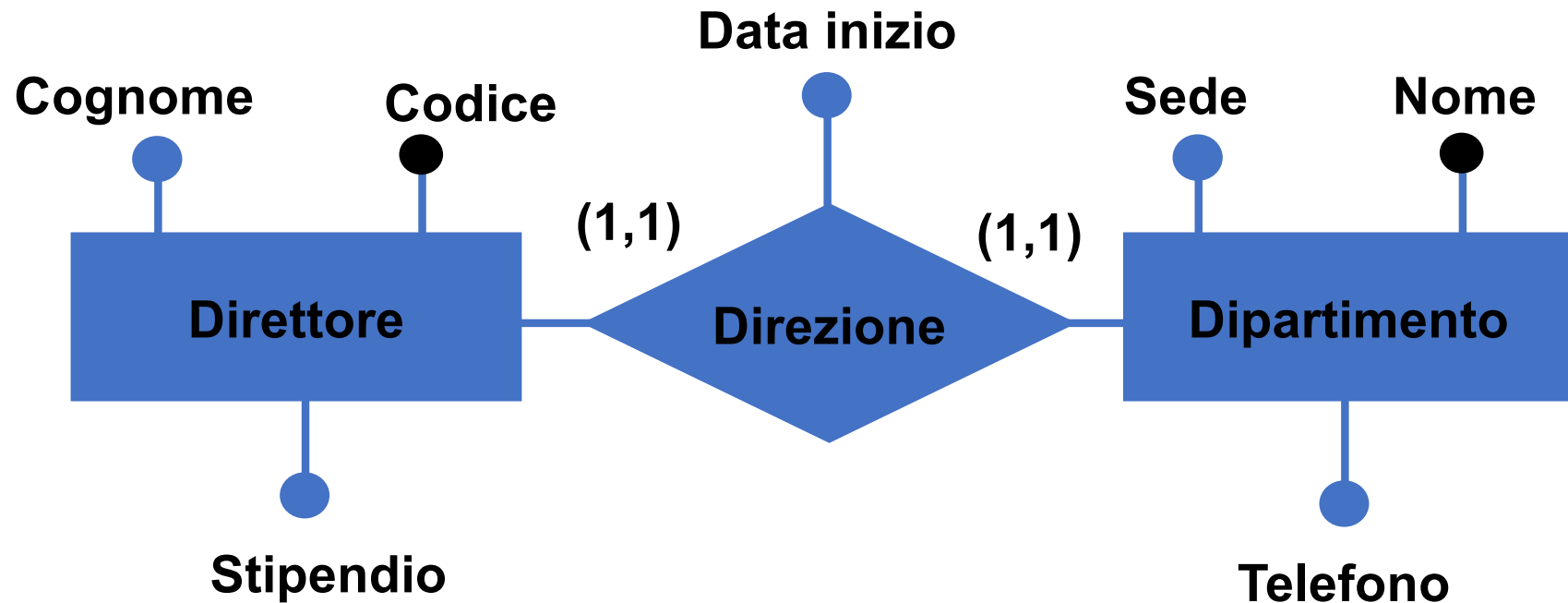
$CC(\underline{ABI}, \underline{CAB}, \underline{NUMCC}, \dots)$



$AGENZE(\underline{1}, \underline{CAB}, \dots)$

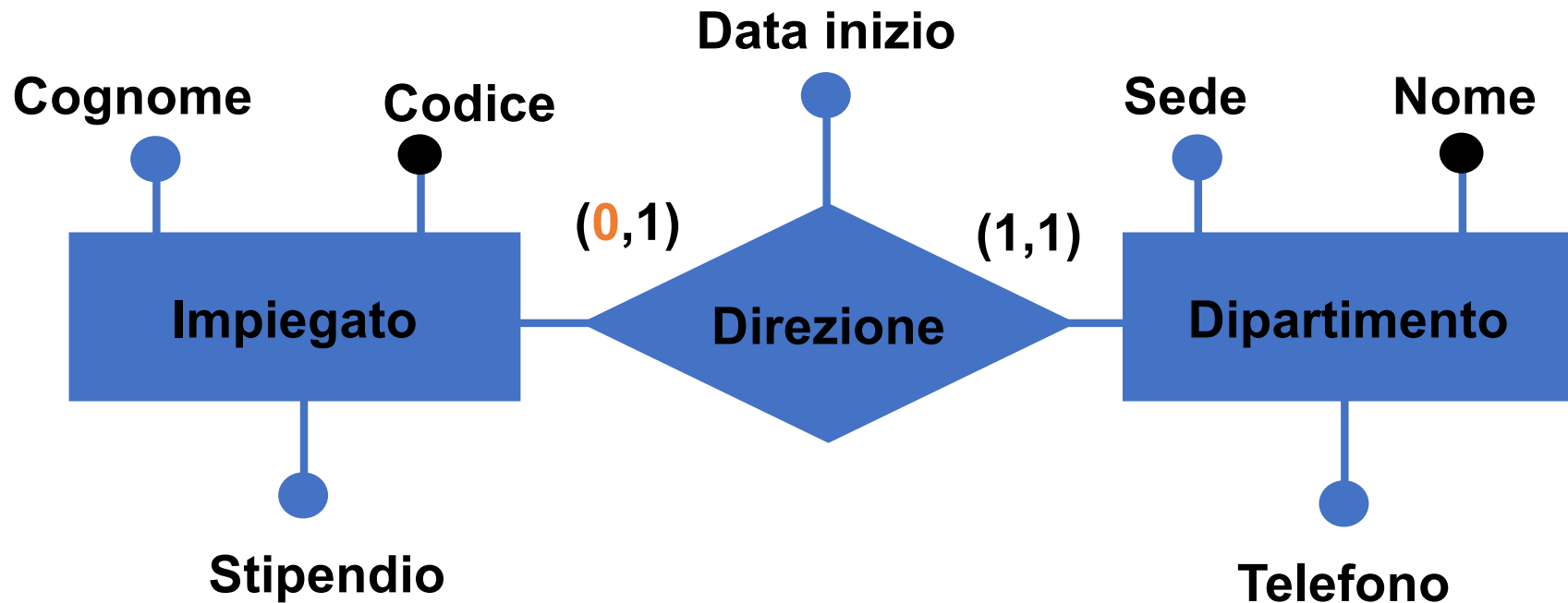
$BANCA(\underline{ABI}, \dots)$

Relationship uno a uno



- **varie possibilità:**
 - **fondere da una parte o dall'altra**
 - **fondere tutto?**

Una possibilità privilegiata

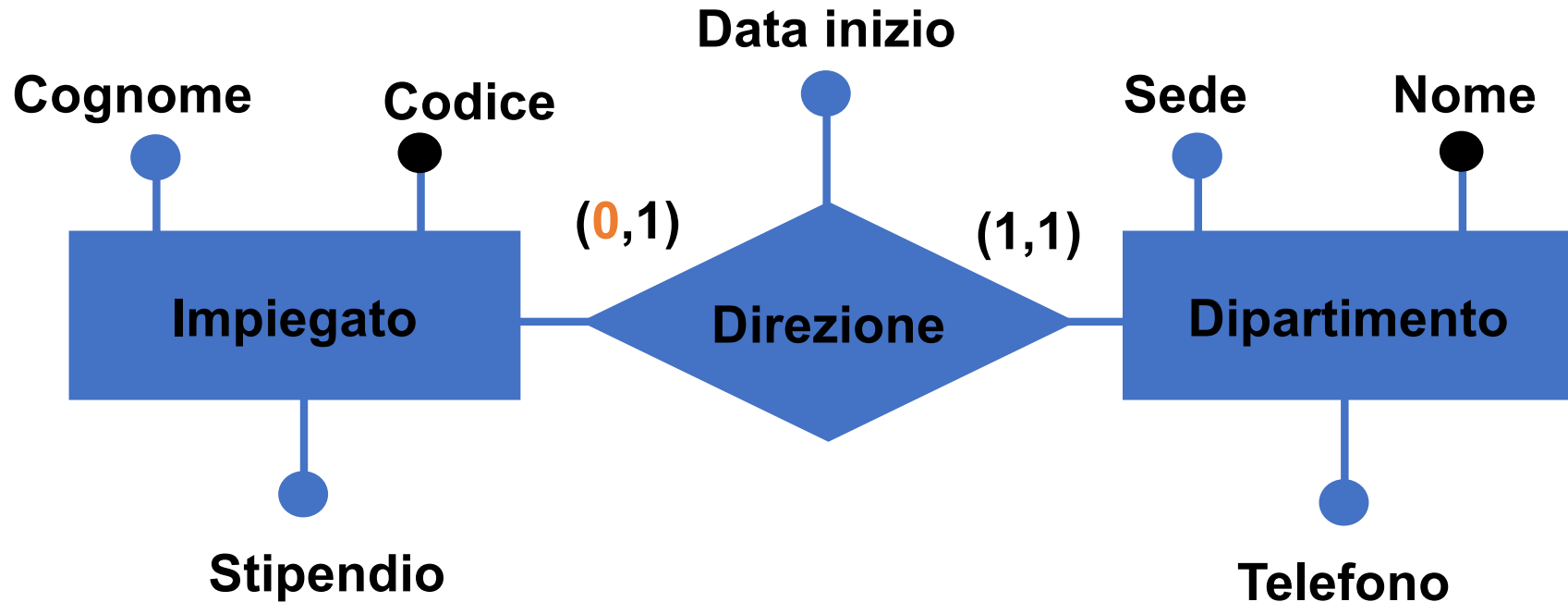


Impiegato (Codice, Cognome, Stipendio)

Dipartimento (Nome, Sede, Telefono, Direttore, DataInizioD)

- **con vincolo di integrità referenziale, senza valori nulli**

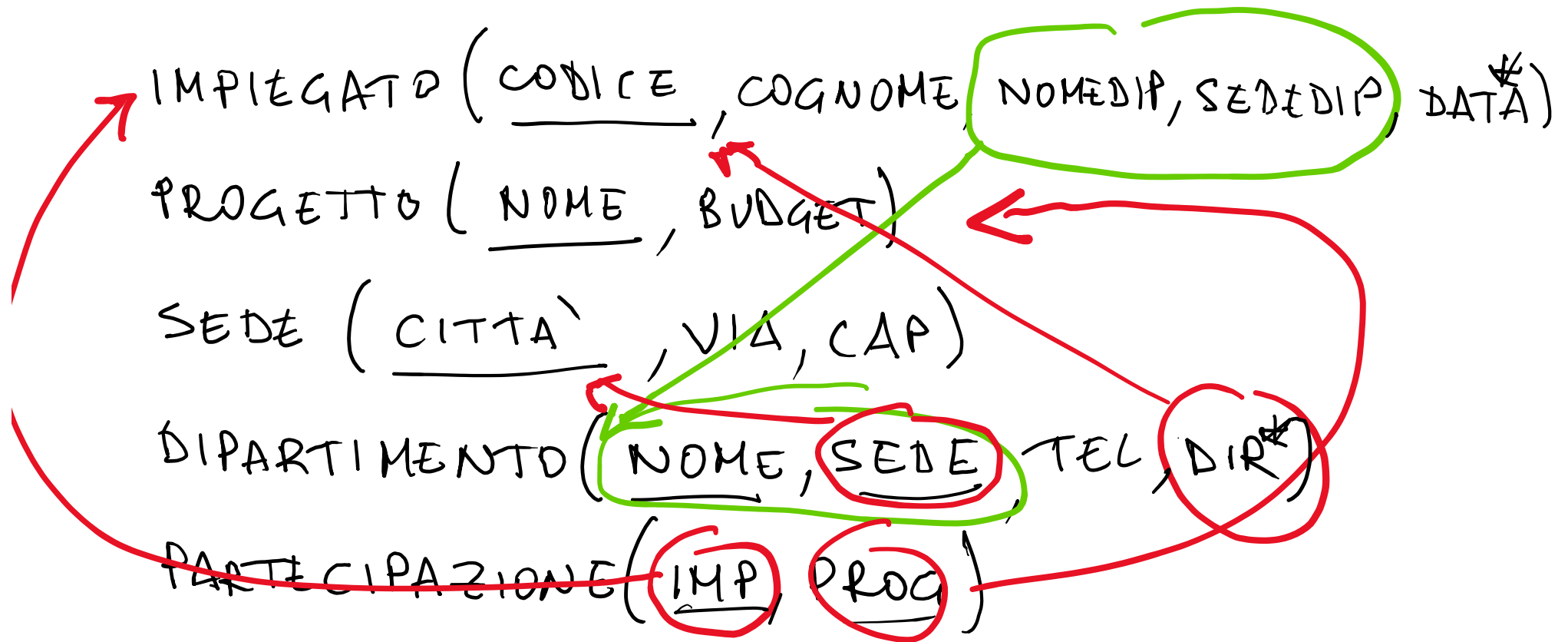
Alternativa, forse non preferibile



Impiegato (Codice, Cognome, Stipendio, Dip, DataInizioD)

Dipartimento (Nome, Sede, Telefono)

- con vincolo di integrità referenziale e valore nullo ammesso su Dip e DataInizioD



SEDE (CITTÀ, VIA, CAP)

PROGETTO(NOME, BUDGET)

IMPIEGATO(COD, COG, NOME DIP, SEDE, DATA ASS^{*})

DIPARTIMENTO(NOME, SEDE, TELEFONO, DIRETTORE^{*})

PARTICIPAZIONE(IMP, PROG)

Schema finale

**Impiegato(Codice, Cognome,
Dipartimento, Sede, Data*)**

Dipartimento(Nome, Città, Telefono, Direttore*)

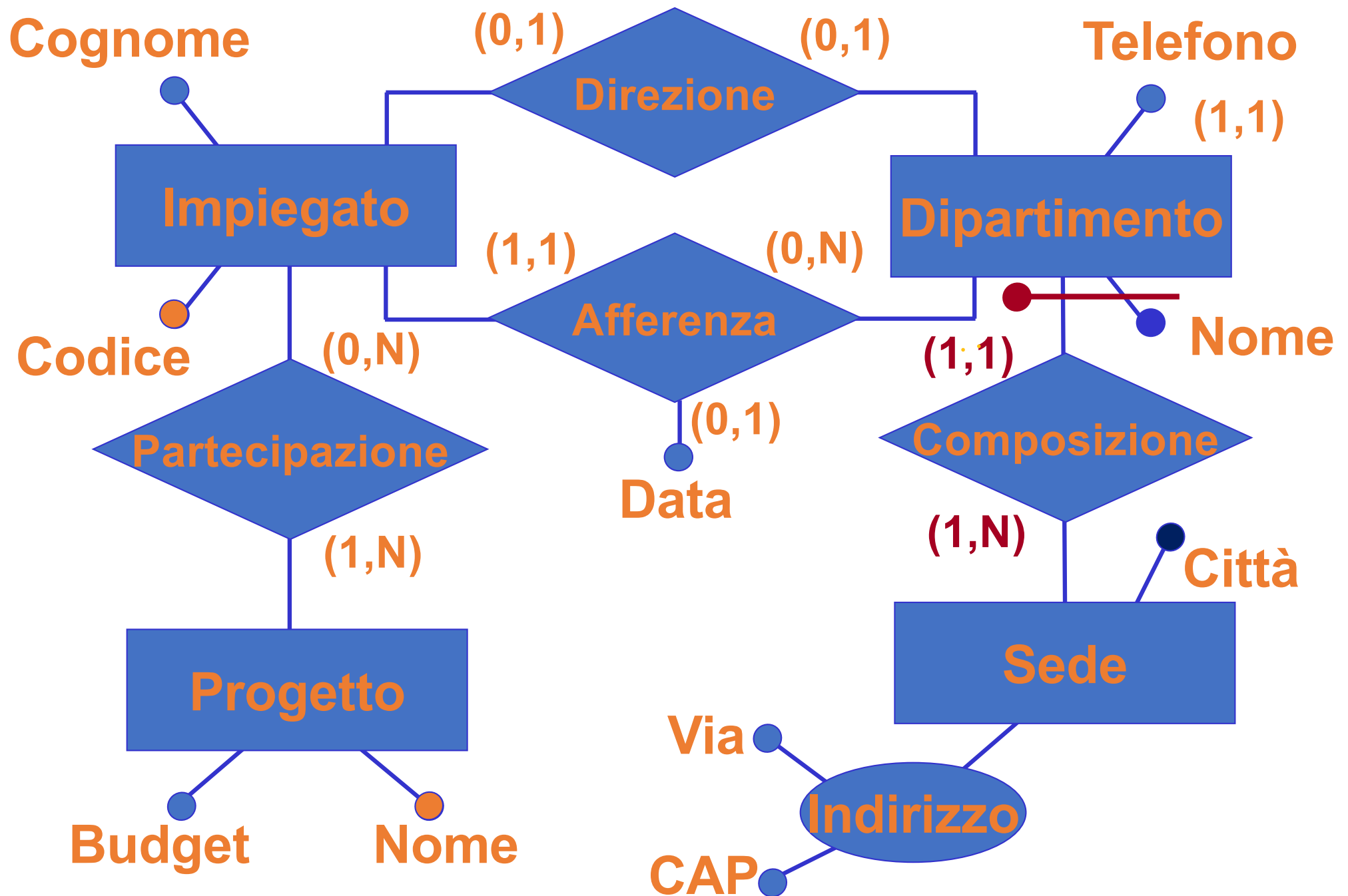
Sede(Città, Via, CAP)

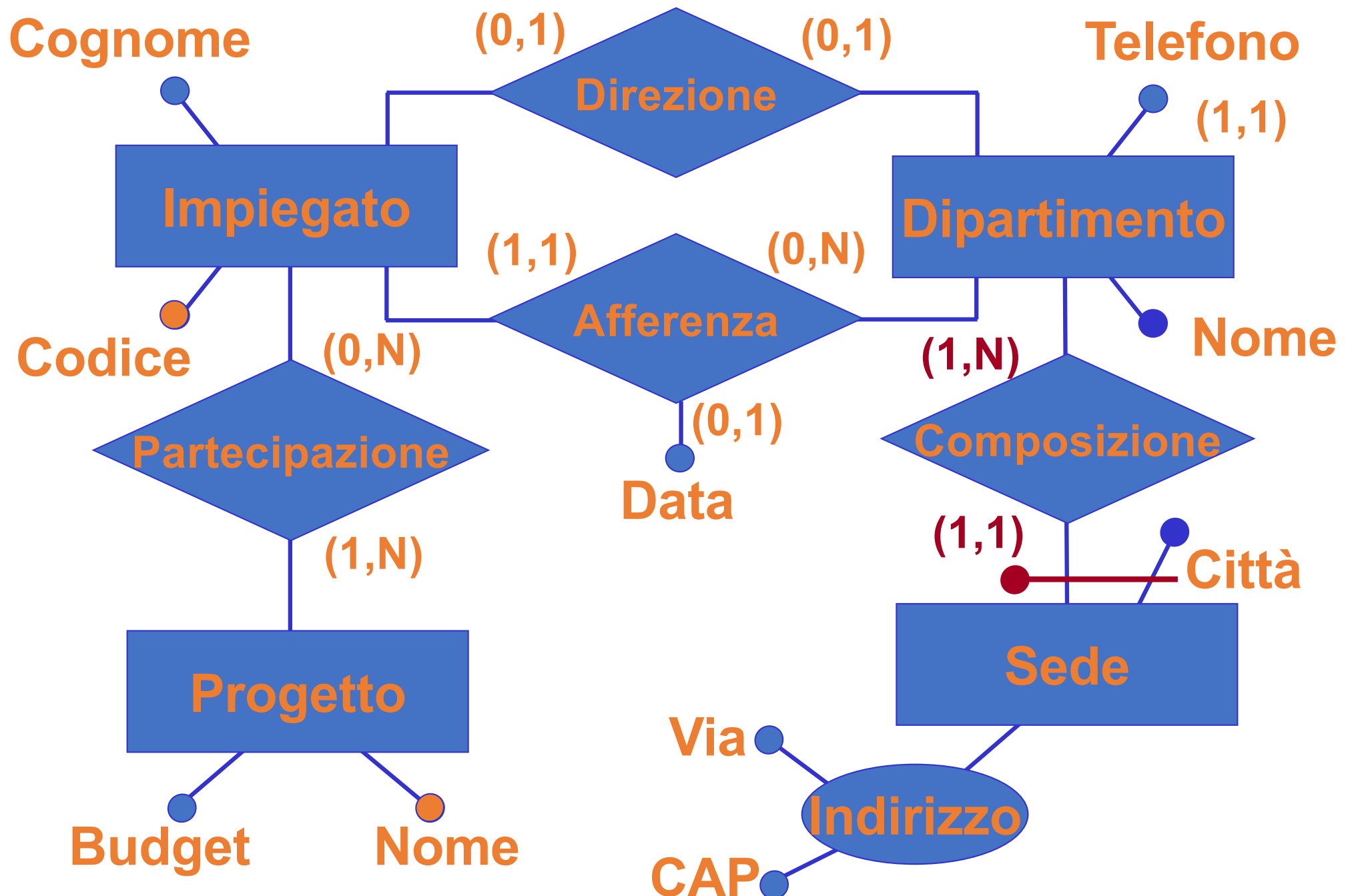
Progetto(Nome, Budget)

Partecipazione(Impiegato, Progetto)

Attenzione

- Differenze apparentemente piccole in cardinalità e identificatori possono cambiare di molto il significato ...





SE

Schema finale

**Impiegato(Codice, Cognome,
Dipartimento, Data*)**

Dipartimento(Nome, Telefono, Direttore*)

Sede(Dipartimento, Città, Via, CAP)

Progetto(Nome, Budget)

Partecipazione(Impiegato, Progetto)