GENERALIZZAZIONE E SPECIALIZZAZIONE

Le gerarchie

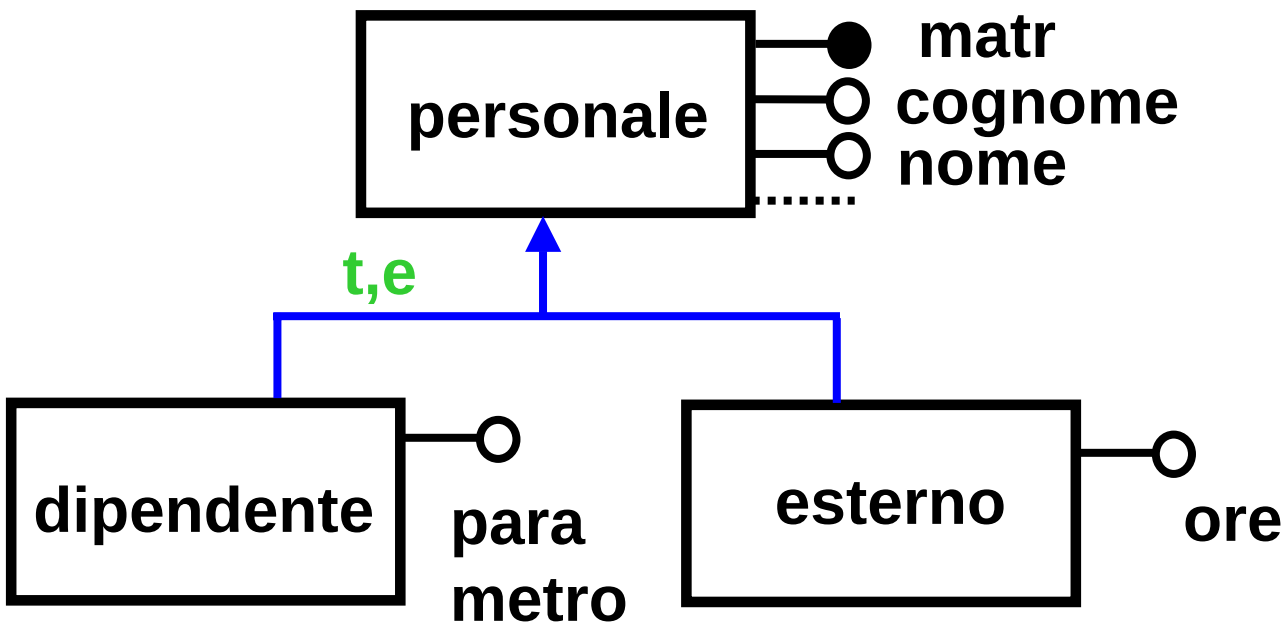
- spesso nella analisi di un settore aziendale può risultare che più entità risultino **simili o casi particolari** l'una dell'altra, derivanti da "**viste**" diverse da parte dell'utenza
- emerge quindi la necessità di evidenziare sottoclassi di alcune classi
- si definisce pertanto **gerarchia di specializzazione** il legame logico che esiste tra classi e sottoclassi

le gerarchie

- la **gerarchia concettuale** è il legame logico tra un'entità padre E ed alcune entità figlie E_1 E_2 .. E_n dove:
- E è la **generalizzazione** di E_1 E_2 .. E_n
- E_1 E_2 .. E_n sono **specializzazioni** di E
- una istanza di E_k **è** anche istanza di E (e di tutte la sue generalizzazioni)
- una istanza di E **può** essere una istanza di E_k

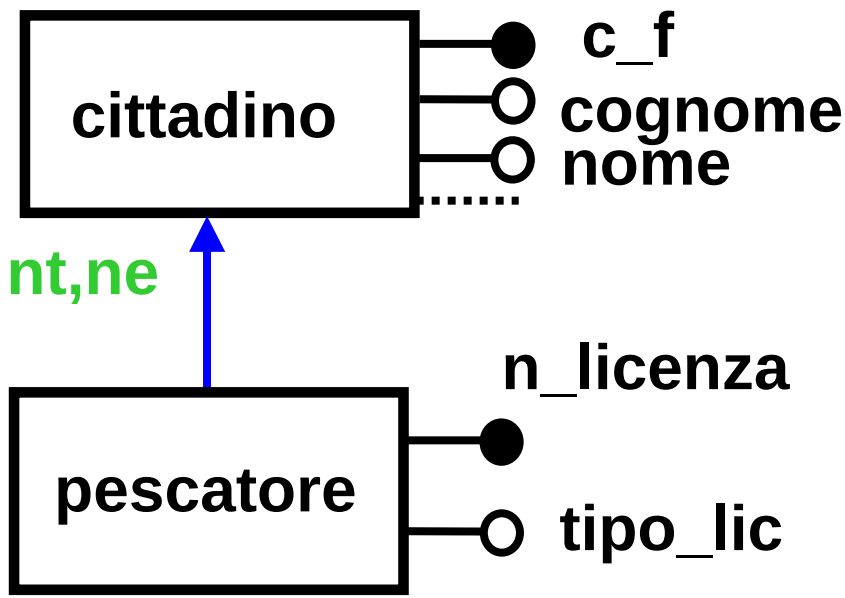
un esempio di gerarchia

un’azienda si avvale dell’opera di professionisti esterni, quindi il suo personale si suddivide in esterni e dipendenti:



un esempio di gerarchia

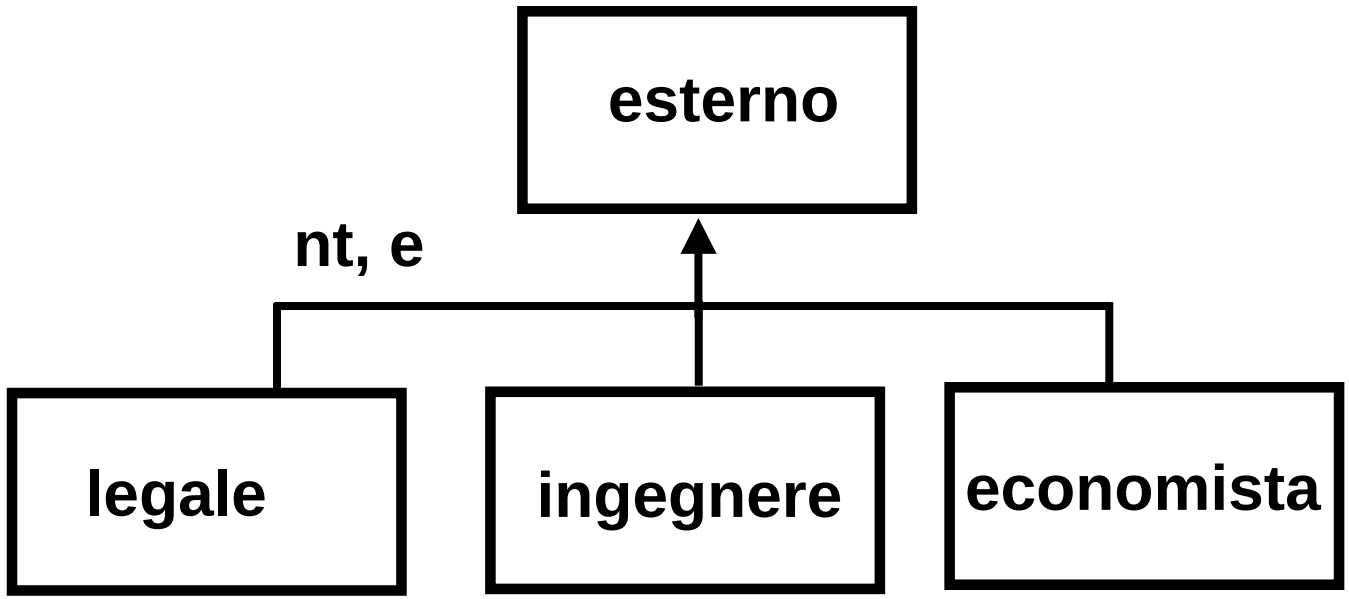
un comune gestisce l'anagrafe ed i servizi per i suoi cittadini alcuni di questi richiedono la licenza di pesca :



gerarchie: definizioni

- ***t*** **sta per totale: ogni istanza dell'entità padre deve far parte di una delle entità figlie**
  - **nell'esempio il personale si divide (completamente) in esterni e dipendenti**
- ***nt*** **sta per non totale: le istanze dell'entità padre possono far parte di una delle entità figlie**
  - **nell'esempio i pescatori sono un sottoinsieme dei cittadini**

un’ulteriore specializzazione

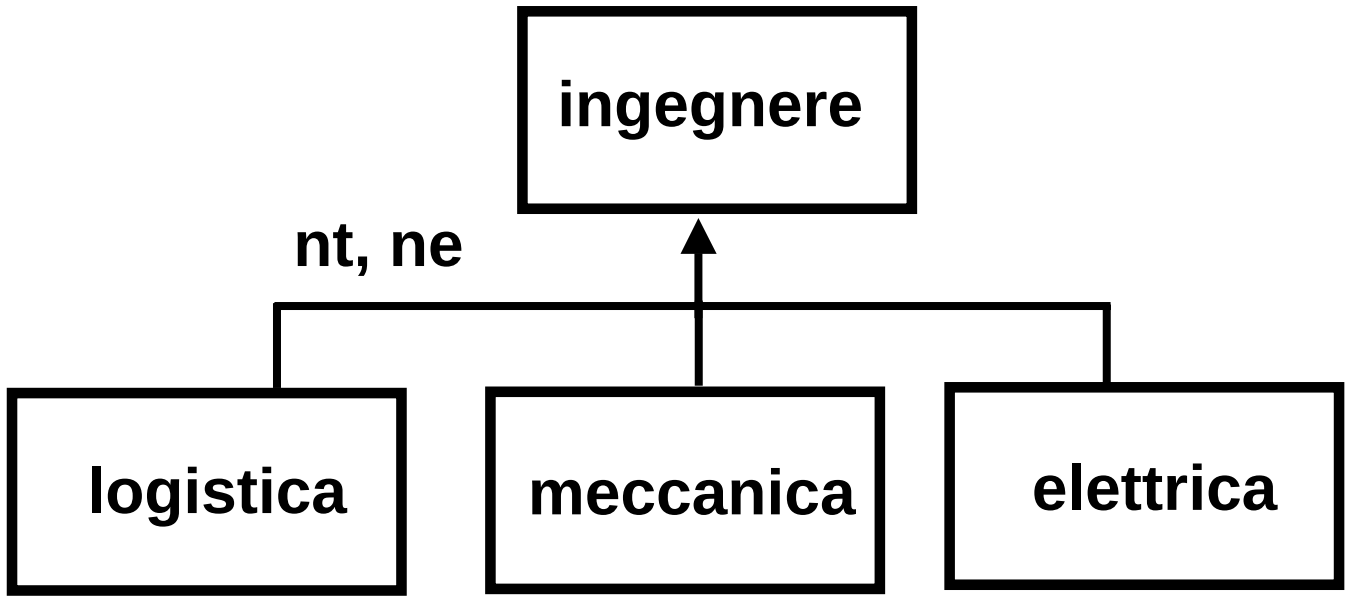


nt **: possono esistere esterni generici che non sono né legali, né ingegneri, né economisti ma non interessa stabilire una sottoclasse ad hoc**

gerarchie: definizioni

- ***e*** **sta per esclusiva: ogni istanza dell'entità padre deve far parte di una sola delle entità figlie**
  - **nell'esempio si esclude che una istanza di personale possa appartenere ad entrambe le sottoclassi**
- ***ne*** **sta per non esclusiva: ogni istanza dell'entità padre può far parte di una o più entità figlie**

un’ulteriore specializzazione



ne **: possono esistere ingegneri sia meccanici, sia elettrici, sia della logistica**
le tre qualifiche non si escludono

ereditarietà delle proprietà

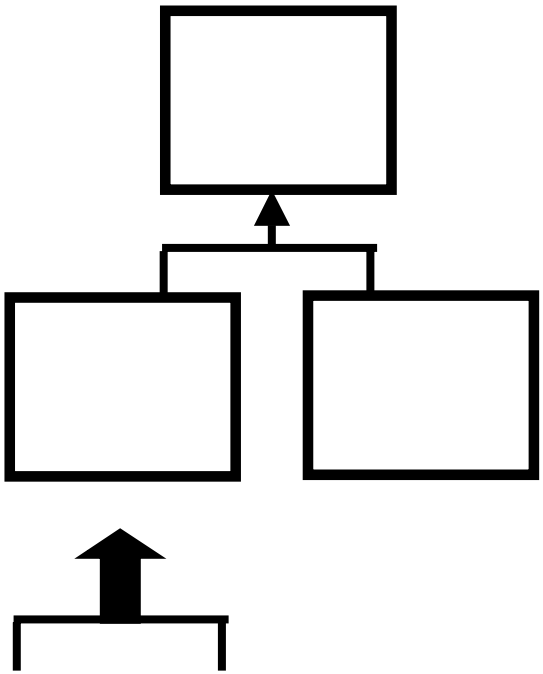
- le **proprietà** dell'entità padre non devono essere replicate sull'entità figlia in quanto questa le **eredita** cioè:
- le proprietà dell'entità padre fanno parte del **tipo** dell'entità figlia
- non è vero il viceversa
  - il tipo di personale è: (**matricola, cognome, nome, indirizzo, data_nascita**)

le gerarchie

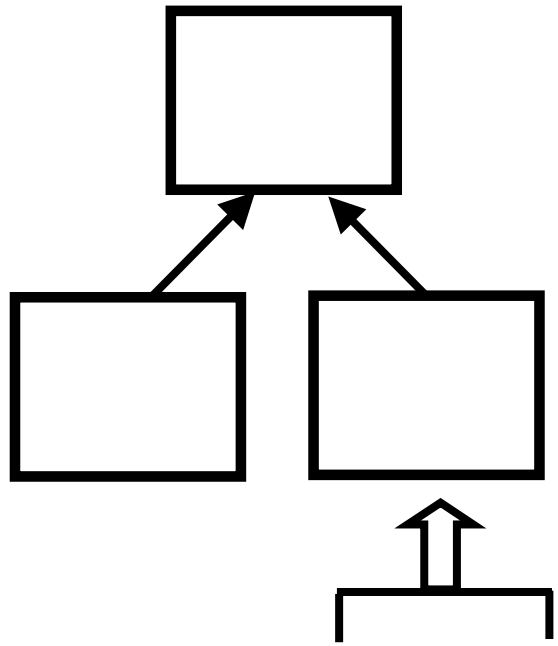
- il tipo di dipendente è: (matricola, cognome, nome, indirizzo, data_nascita, parametro)
- il tipo di esterno è: (matricola, cognome, nome, indirizzo, data_nascita, ore)
- dipendente ed esterno hanno lo stesso tipo

  se considerati insieme come personale
- le gerarchie concettuali sono anche denominate gerarchie ISA
- dipendente è un (is a) personale
- esterno è un (is a) personale

altri simboli grafici

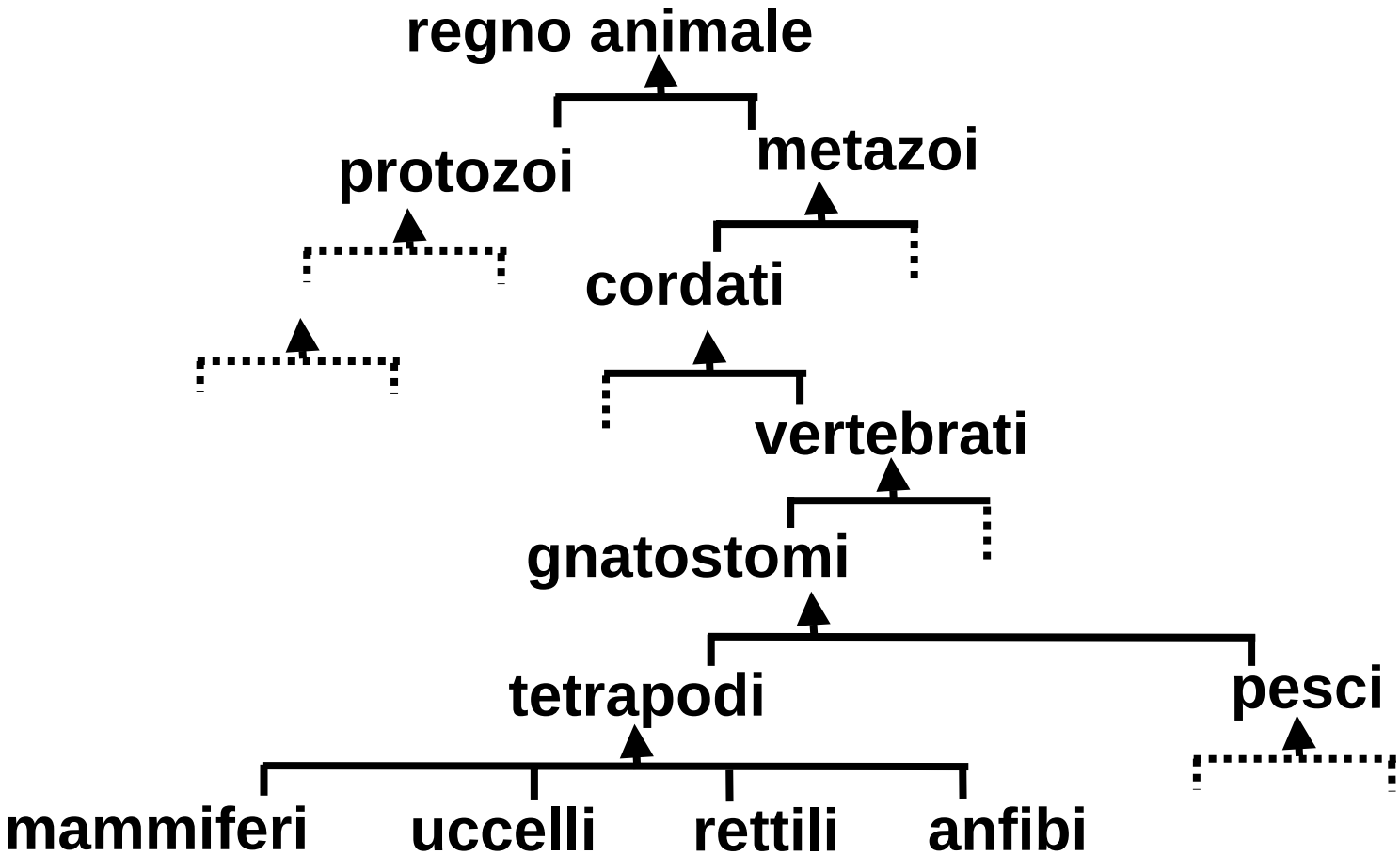


sottoclassi disgiunte

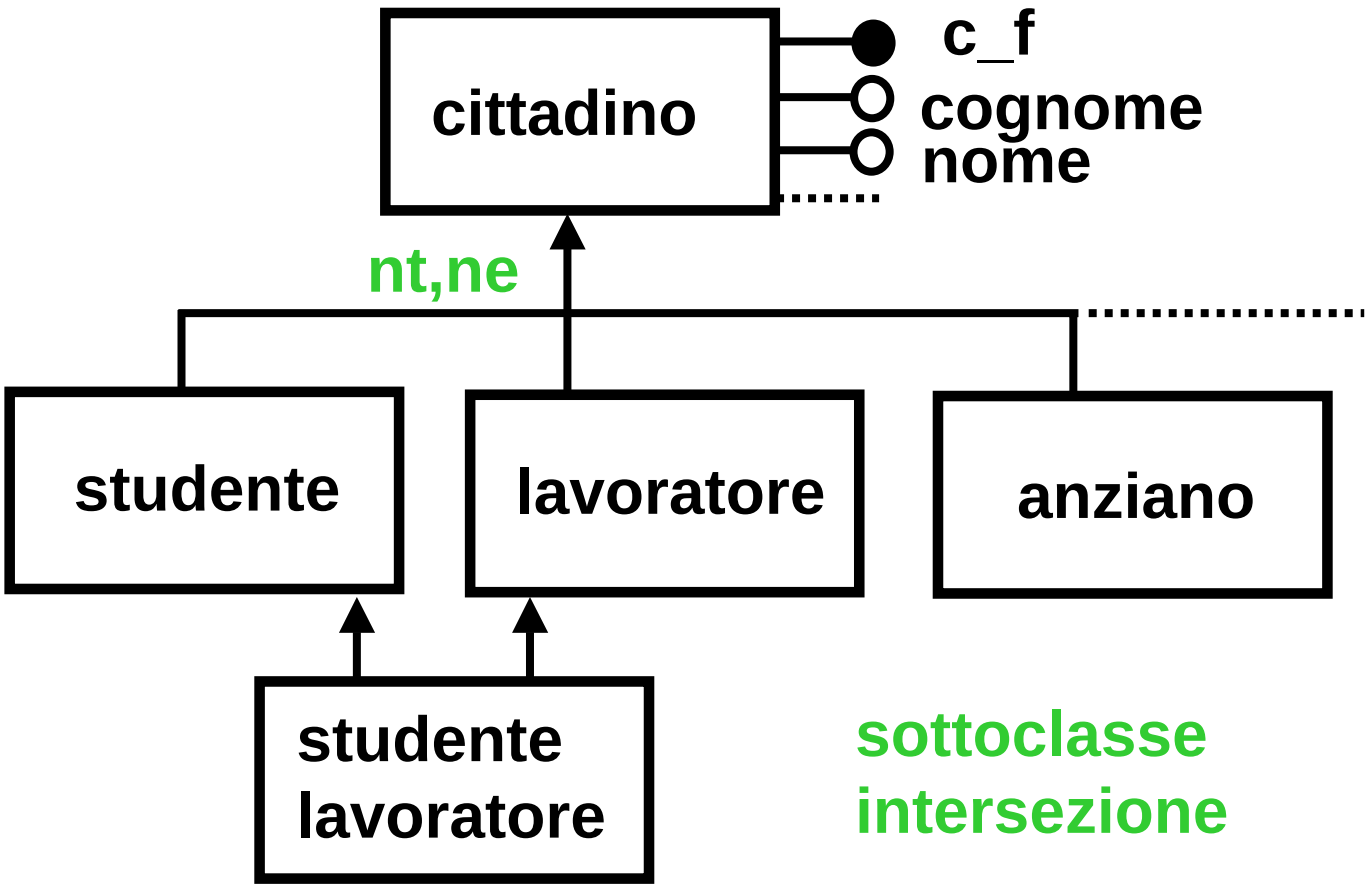


sottoclassi non disgiunte

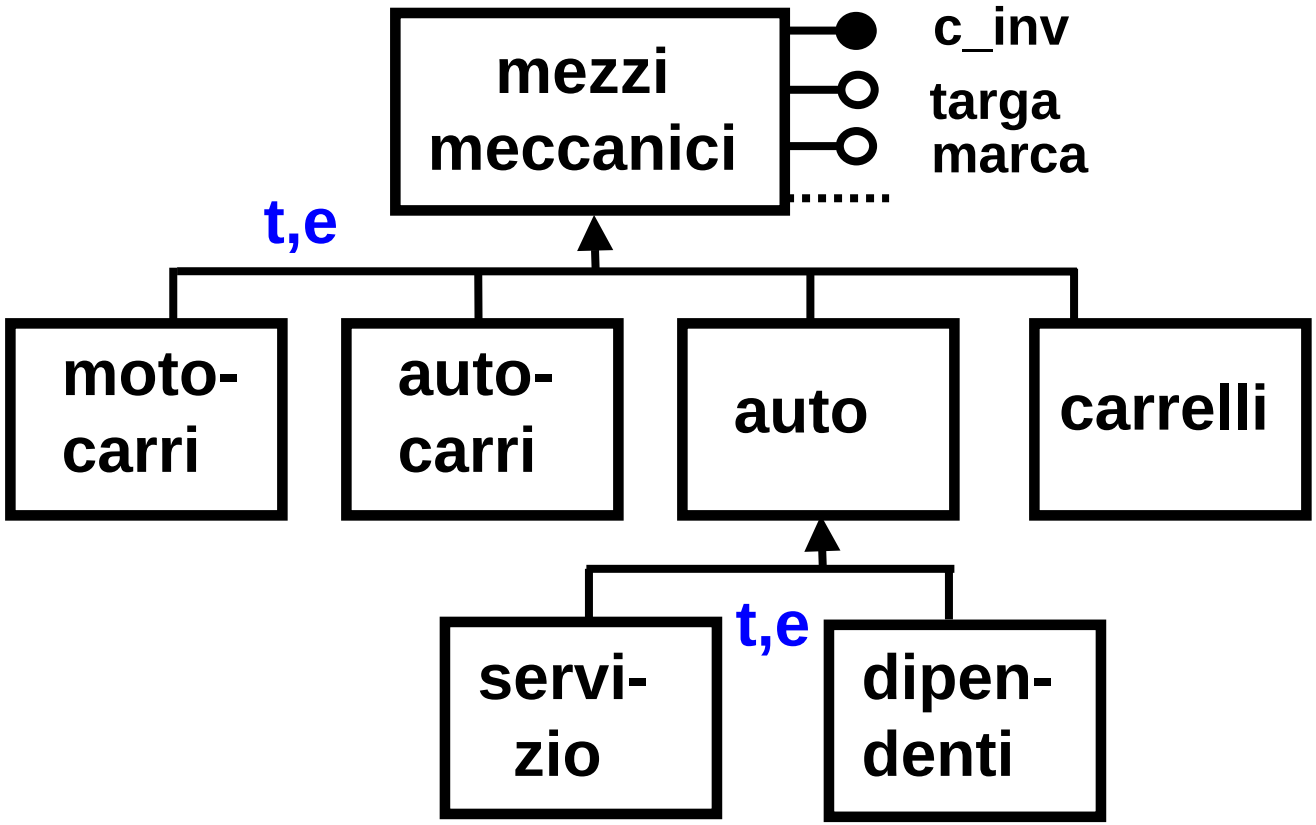
una delle gerarchie più note



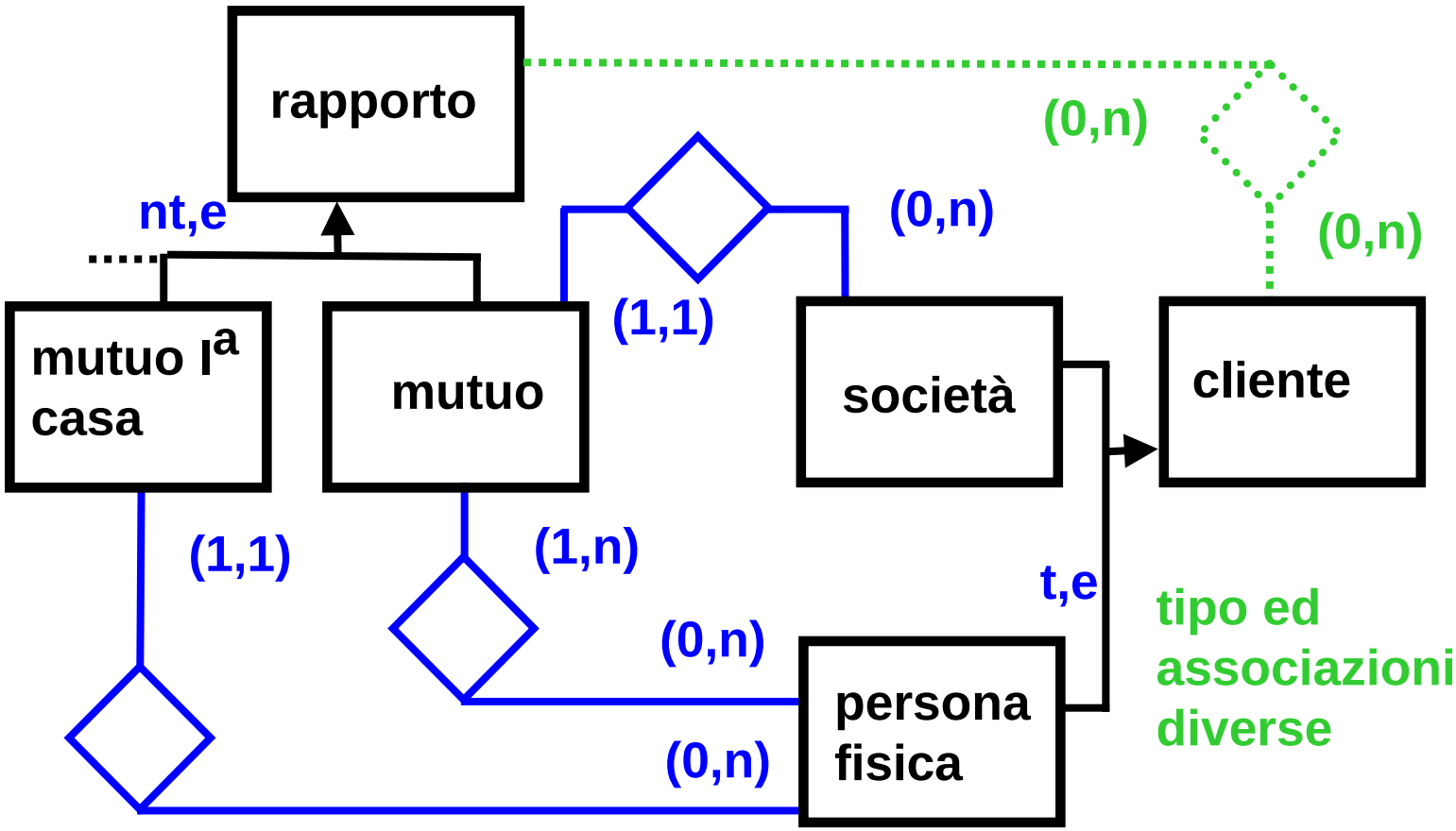
esempio: un comune



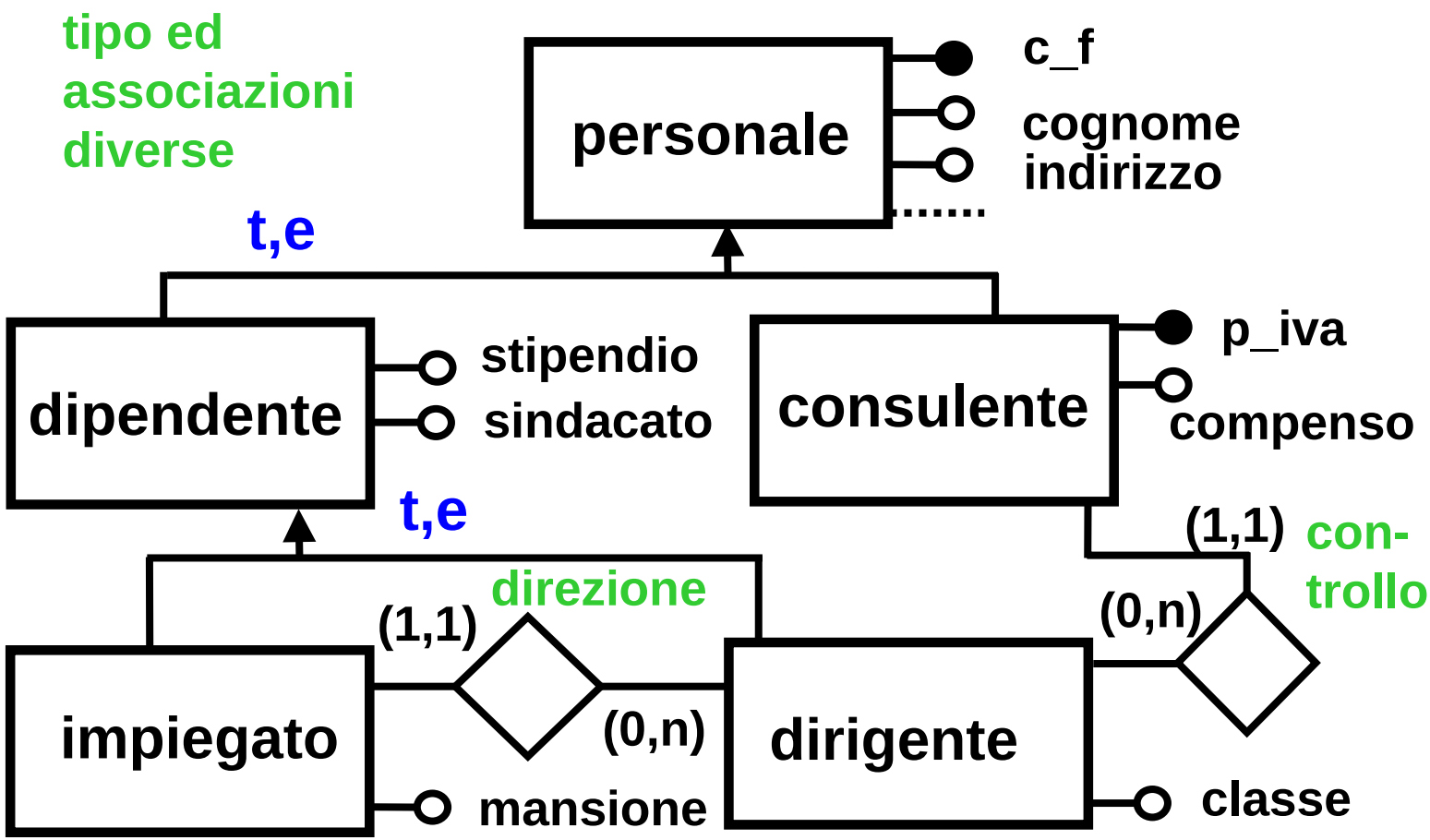
es.: parco mezzi mecc.



es.: clienti-banca

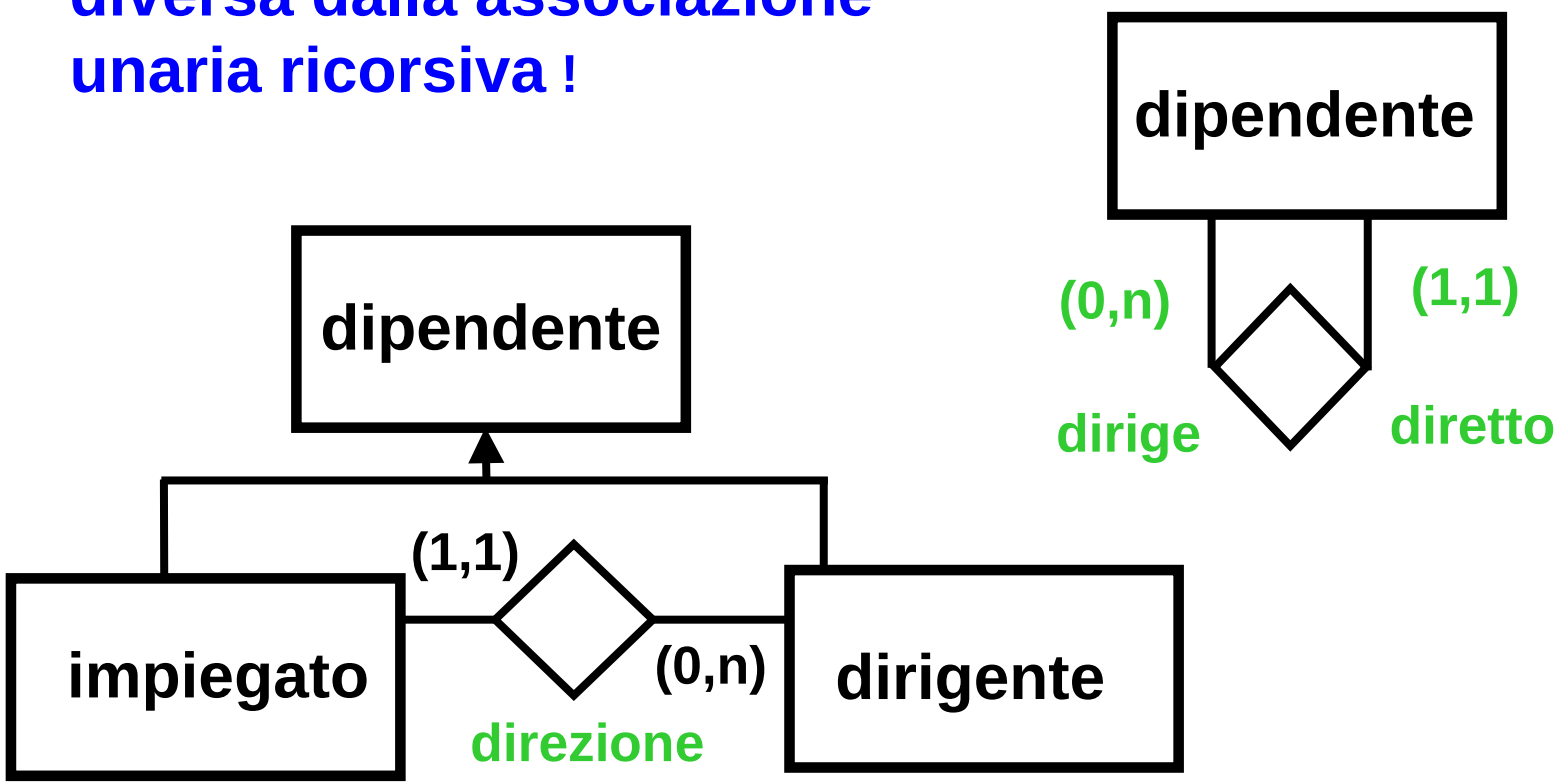


es.: personale d'azienda



gerarchie isa

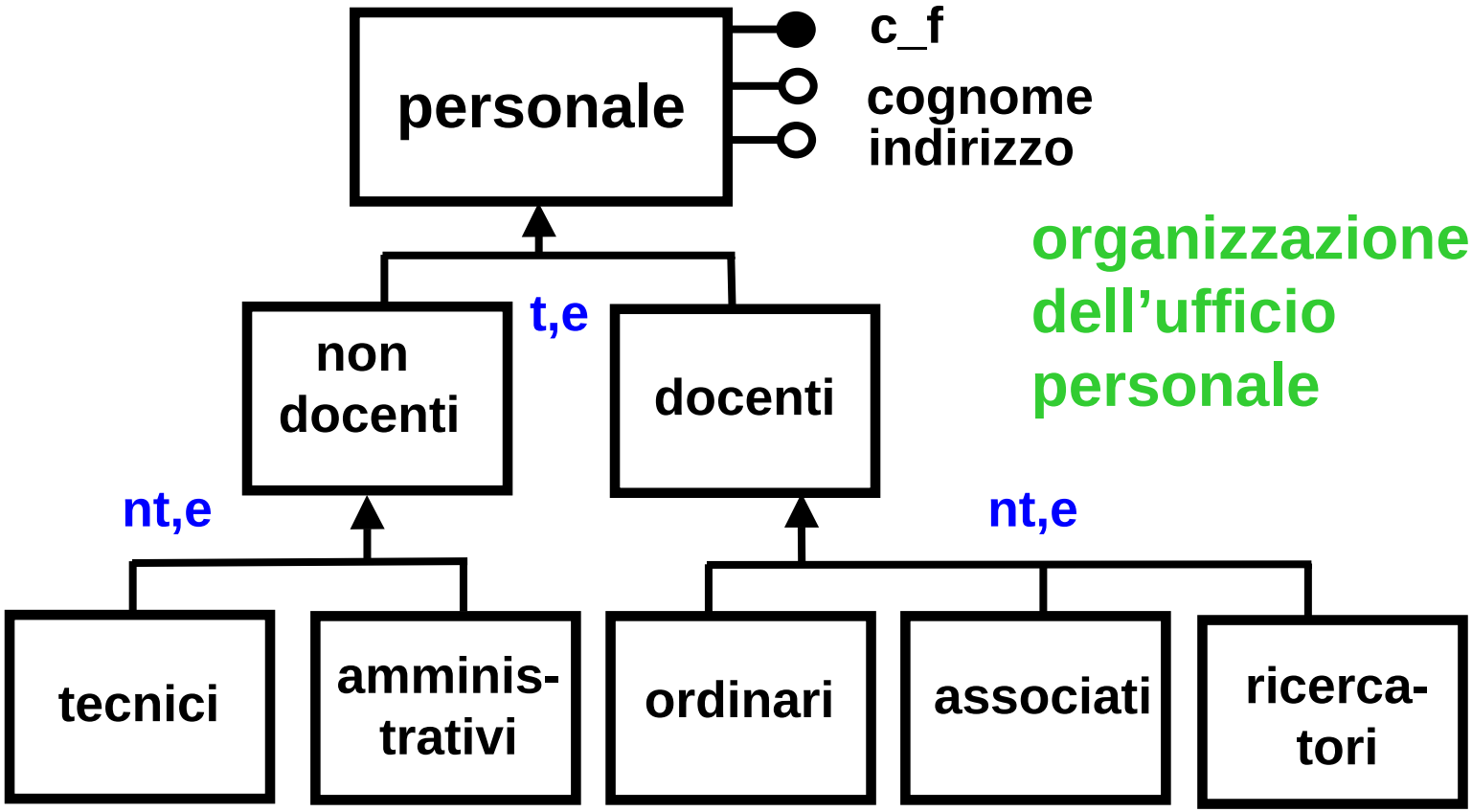
attenzione : la gerarchia isa è diversa dalla associazione unaria ricorsiva !



uso delle gerarchie

- **scomposizione di entità in sottoclassi**
- **ricomposizione di schemi parziali in uno schema generale**
- **espansione di schemi consolidati per trattare nuovi sottoproblemi**
- **spesso la gerarchia riflette più o meno esattamente la diversificazione del lavoro di uffici diversi (viste d'utente)**
  - **vediamo un esempio:**

diversificazione: università



strategie di progetto

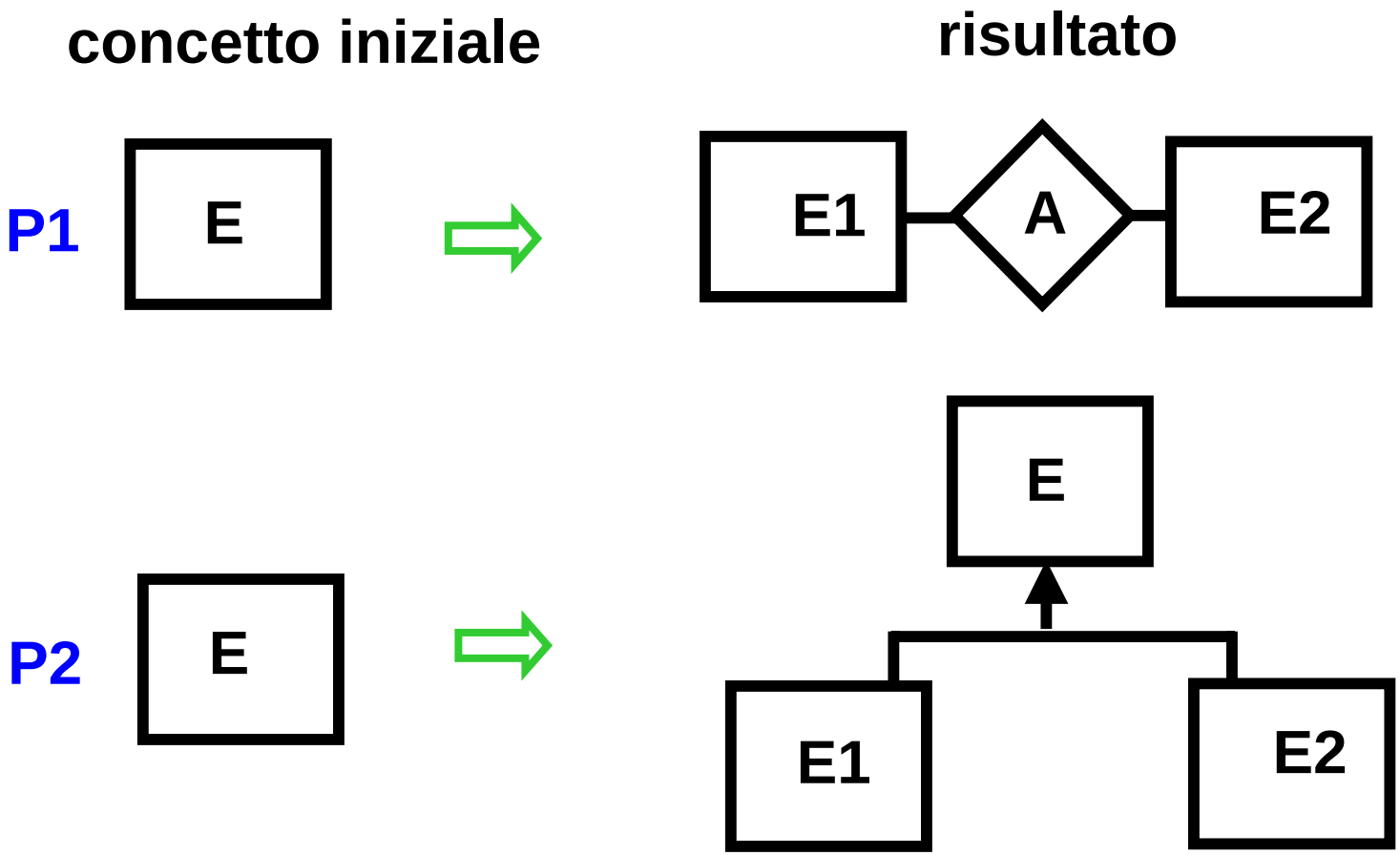
- Lo sviluppo dello schema, come tutti i progetti di ingegneria, si può eseguire seguendo quattro strategie fondamentali:
- Top-Down
- Bottom-Up
- Inside-Out
- Mista

strategie di progetto

Top-Down:

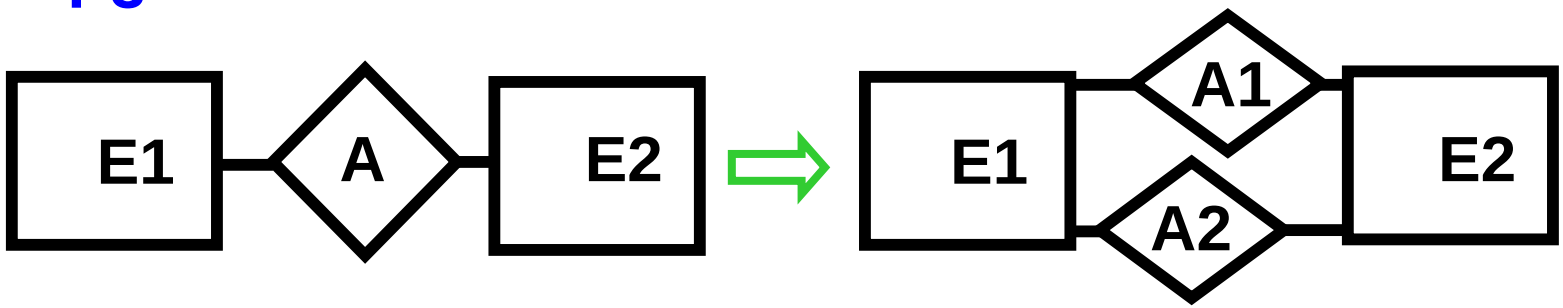
- a partire dalle specifiche si costruisce uno schema iniziale
- dallo schema iniziale si arriva per raffinamenti successivi a schemi intermedi e poi allo schema finale
- i raffinamenti prevedono l'uso di trasformazioni elementari (primitive) che operano sul singolo concetto per descriverlo con maggior dettaglio

primitive di trasformazione

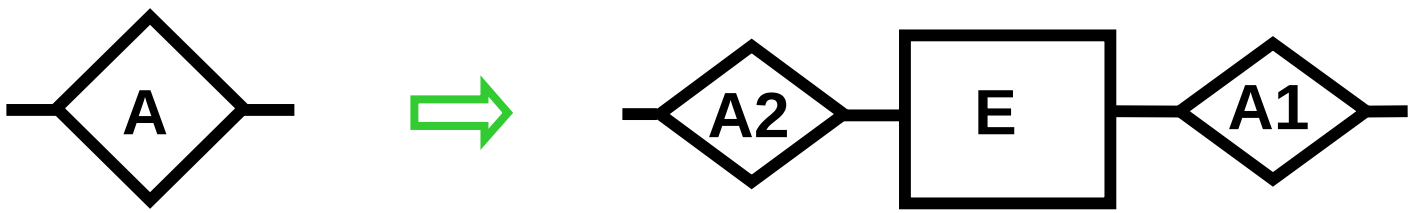


primitive di trasformazione

P3

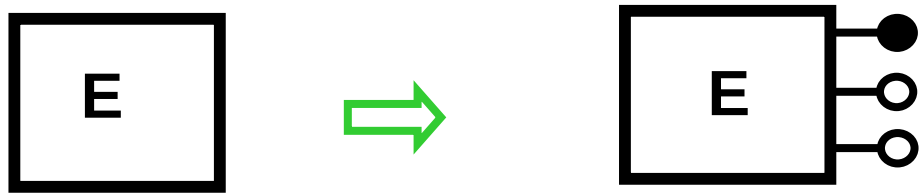


P4

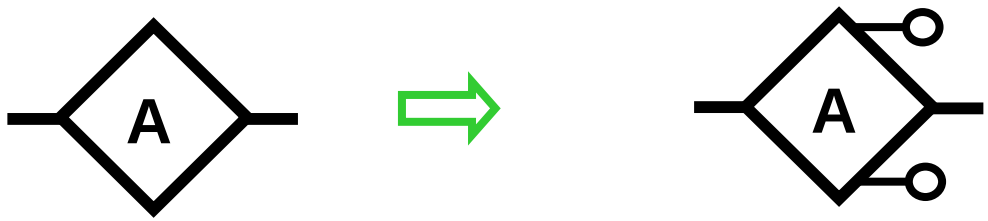


primitive di trasformazione

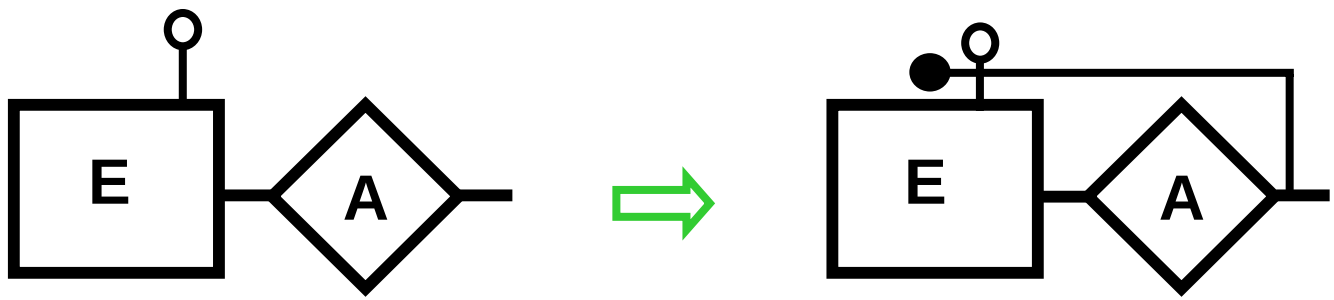
P5



P6

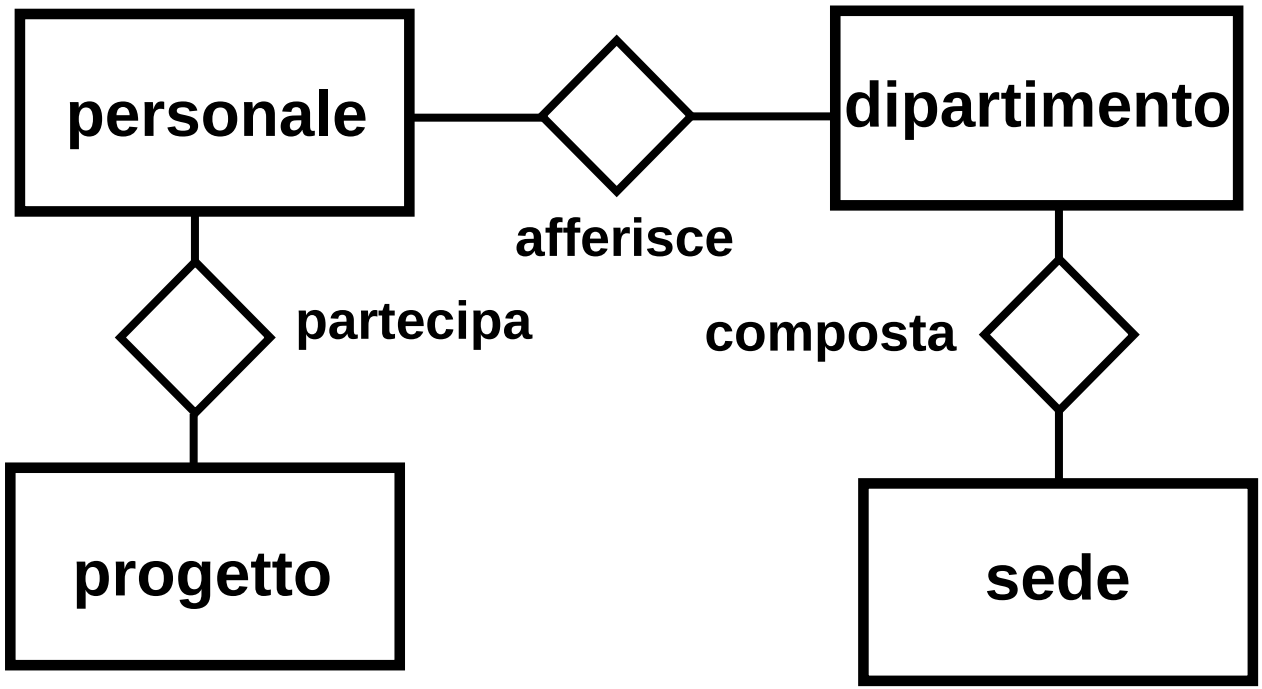


P7



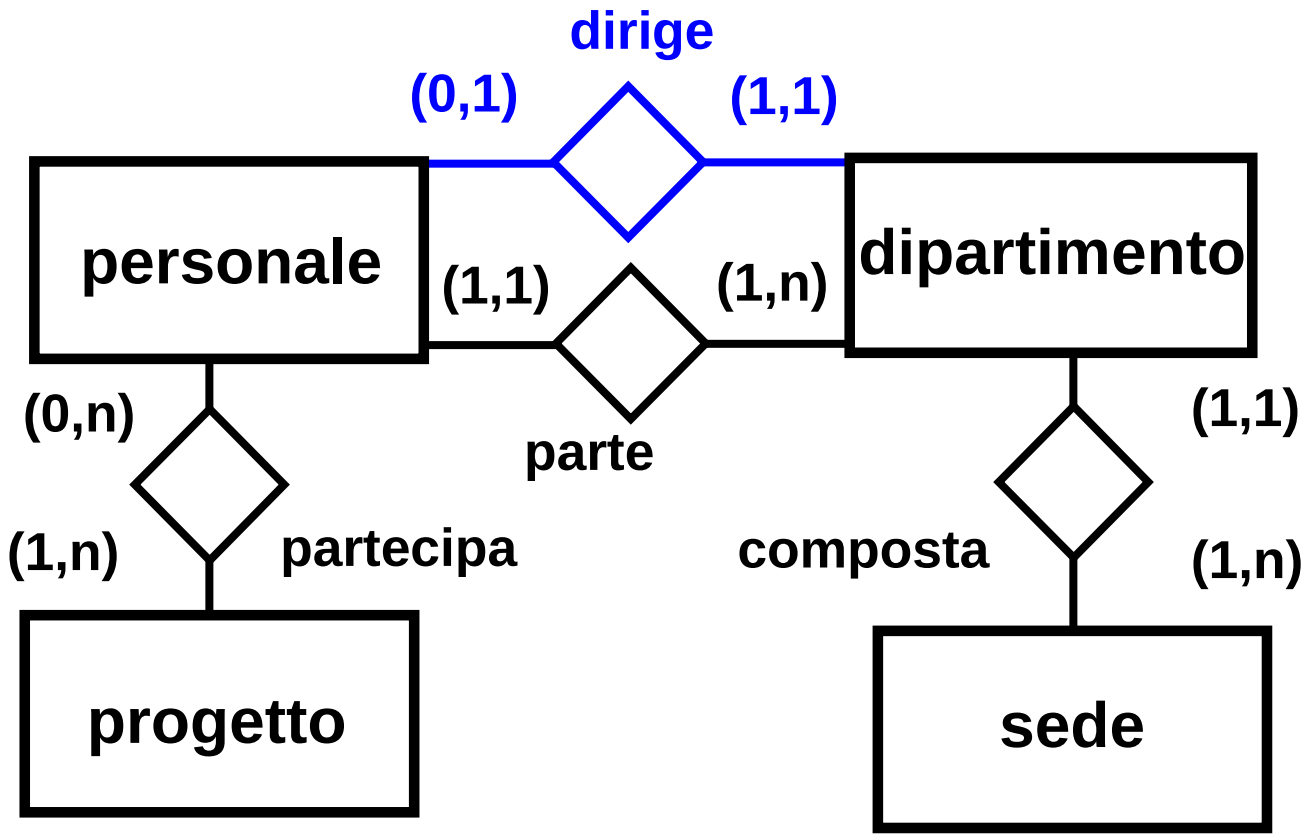
es.: sviluppo top-down

schema iniziale

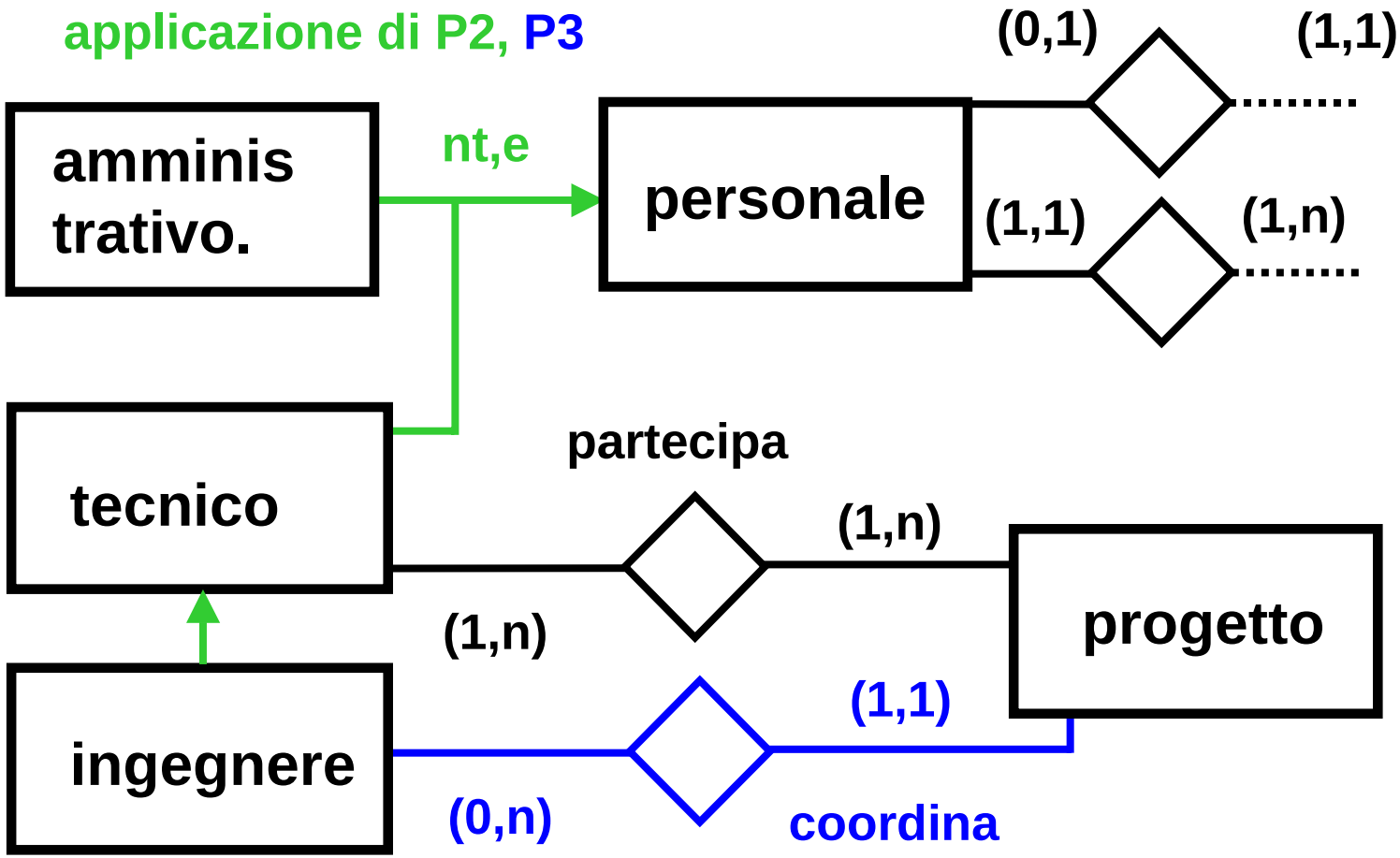


es.: sviluppo top-down

applicazione di P3

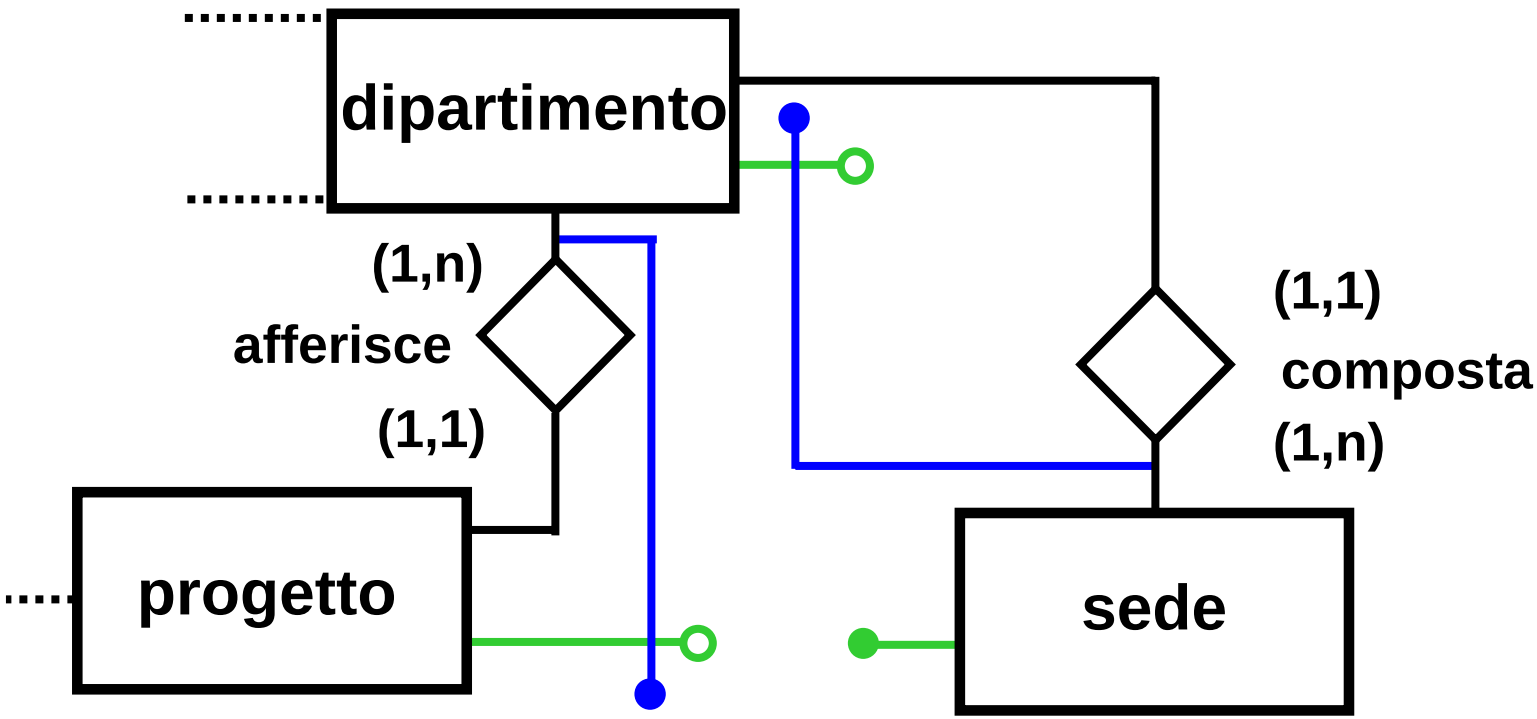


es.: sviluppo top-down



es.: sviluppo top-down

applicazione di P5, P7



strategia top down

- vantaggi:
  - il progettista descrive inizialmente lo schema trascurando i dettagli
  - precisa lo schema gradualmente
- problema:
  - non va bene per applicazioni complesse perché è difficile avere una visione globale precisa iniziale di tutte le componenti del sistema

strategia bottom-up

- le specifiche nascono suddivise per sottoprogetti descriventi frammenti limitati della realtà da schematizzare
- si sviluppano i sottoschemi separati
- si fondono i sottoschemi per ottenere lo schema finale
- un esempio può essere costituito dai due ultimi lucidi dell'es. precedente lasciando per ultimo i collegamenti tra personale e dipartimenti e progetti

strategia bottom-up

- **vantaggi:**
  - **diversi progettisti elaborano gli schemi parziali, il singolo progettista ha una visione più precisa del proprio settore**
  - **va bene per applicazioni complesse**
- **problema:**
  - **conflitti e difficoltà di integrazione, non è facile avere una visione globale**

ulteriori strategie

- **inside-out: è una variante della bottom-up, si sviluppano schemi parziali in aggiunta a sottoschemi già definiti precedentemente e separatamente**
- **strategia mista: si parte da uno schema scheletro generale preciso, poi lo si suddivide in sottoschemi da dettagliare e fondere alla fine**