

IL MODELLO CONCETTUALE ENTITÀ-RELAZIONE (ER) (CAPITOLO 5 DELLA VERSIONE ITALIANA)

Obbiettivo:

Introdurre la progettazione concettuale

Definire il linguaggio E-R

Discuterne i costrutti principali

Esempi

Fasi di sviluppo di una BD

❖ Analisi dei Requisiti

❖ **Progettazione Concettuale**

❖ Progettazione Logica

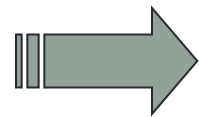
❖ (scelta del DBMS)

❖ Raffinamento dello Schema

❖ Progettazione Fisica e ulteriore

❖ raffinamento dello schema

❖ Progettazione delle Applicazioni e della Sicurezza



Analisi dei requisiti

- Consente di determinare quali dati devono essere memorizzati e quali applicazioni (distinte) insisteranno su di essi
- Requisiti utente
- Processo informale

Progettazione Concettuale

- ❖ Il modello concettuale serve descrivere tutte le informazioni che popoleranno la futura base di dati e le relazioni che nel dominio applicativo sussistono tra loro.
- ❖ Tali proprietà sono formalizzate attraverso un linguaggio molto generale (alto potere espressivo) che è detto linguaggio ER e il risultato della descrizione è detto schema ER (Entita'-Relazioni)

Progettazione logica

- Dipende dal DBMS sottostante
- Nei DB relazionali , coincide con la definizione dello *schema logico*

Raffinamento dello schema

- Analisi dello schema logico iniziale
- Determinazione dei problemi:
 - incompletezza
 - ridondanze
- La normalizzazione è un processo formalizzato che garantisce, tramite una teoria algebrica rigorosa, alcune proprietà importanti

Progettazione Fisica

- Analisi dei carichi
- Definizione degli indici o ristrutturazione
- Obiettivo: massimizzare le prestazioni rispetto alle applicazioni previste

Progettazione Applicazioni e Sicurezza

- Definizione dei processi applicativi
- Determinazione di (classi di) processi simili
- Creazione di gruppi di utenti, privilegi e regole di accessibilità

Progettazione Concettuale

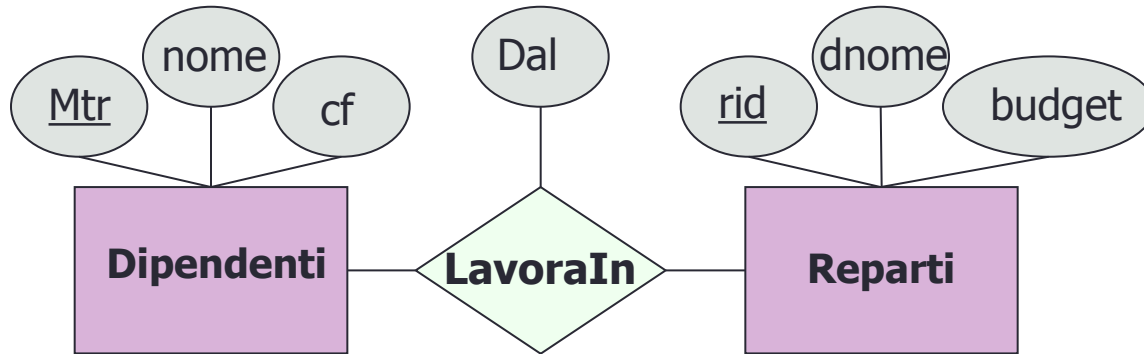
- Obiettivi
 - Quali sono le **entità** e le **relazioni** dell'organizzazione?
 - Quali **informazioni** su queste entità e relazioni dovrebbero essere memorizzate nella base di dati?
 - Quali sono i **vincoli** di integrità o le business rules in vigore?
- Vantaggi
 - Uno “schema” di base di dati nel modello ER può essere rappresentato graficamente (diagrammi ER)
 - Si può tradurre un diagramma ER in uno schema relazionale

Progettazione Concettuale

- **Conseguenze**

- Raffinamento dello schema (normalizzazione): controllo dello schema relazionale per trovare ridondanze e relative anomalie.
- Progettazione fisica e ulteriore raffinamento dello schema: si considerano il carico di lavoro e le prestazioni del sistema per effettuare ulteriori modifica sullo schema.

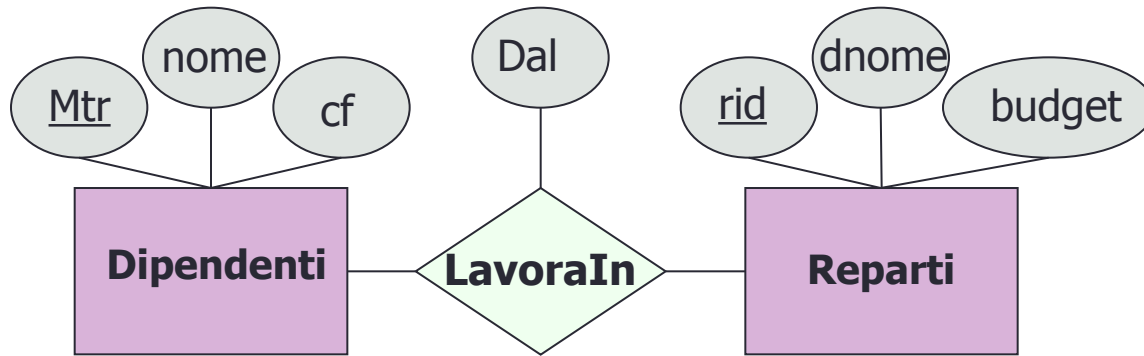
Elementi del modello ER



- ◆ **Entità** è un oggetto del mondo reale distinguibile da altri oggetti. E' descritta in un modello ER da un insieme di attributi e dal loro dominio.
- ◆ **Insieme di Entità:** Una collezione di entita' simili, per esse definisco un attributo chiave che le distingue.

Dipendenti(Mtr:char(20),Nome:char(20),cf:char(20))

Elementi del modello ER



- ◆ **Entità** è un oggetto del mondo reale distinguibile da altri oggetti. E' descritta in un modello ER da un insieme di attributi e dal loro dominio.
- ◆ **Insieme di Entità:** Una collezione di entita' simili, per esse definisco un attributo chiave che le distingue.

Dipendenti(Mtr:char(20),Nome:char(20),cf:char(20))

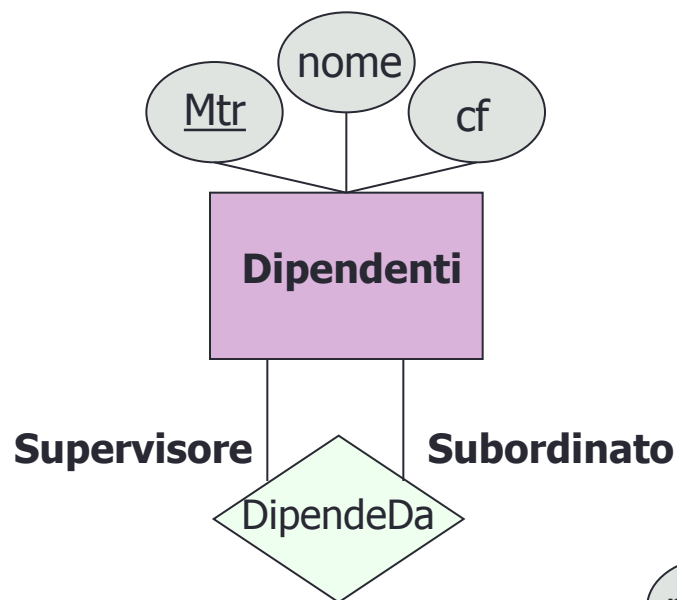
- ◆ **Relazione:** Un'associazione tra due o piu' entità
- ◆ **Insieme di Relazioni:** Una collezione di associazioni
Ogni relazione e' individuata solo dalle entita' che vi partecipano

LavoraIn={(m,r) | m ∈ Dipendenti.Mtr, r ∈ Reparti.rid}

Entità e Chiavi

- Una chiave è un sottoinsieme degli attributi di una classe di entità che, se completamente istanziato (cioè assegnato ad un valore costante), consente di individuare univocamente una istanza della classe
- ES:
 - *Numero Matricola* (oppure *Codice Fiscale*)
 - 4-pla: (*Nome, Cognome, DataNasc, LuogoNasc*)

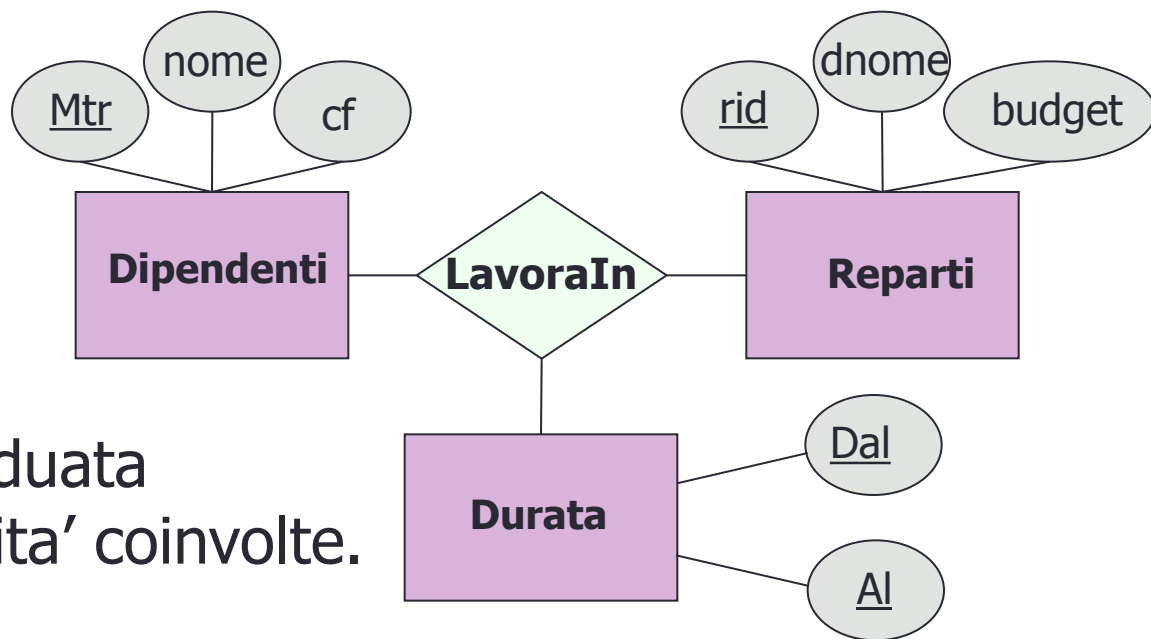
Ruoli e Relazioni Ternarie



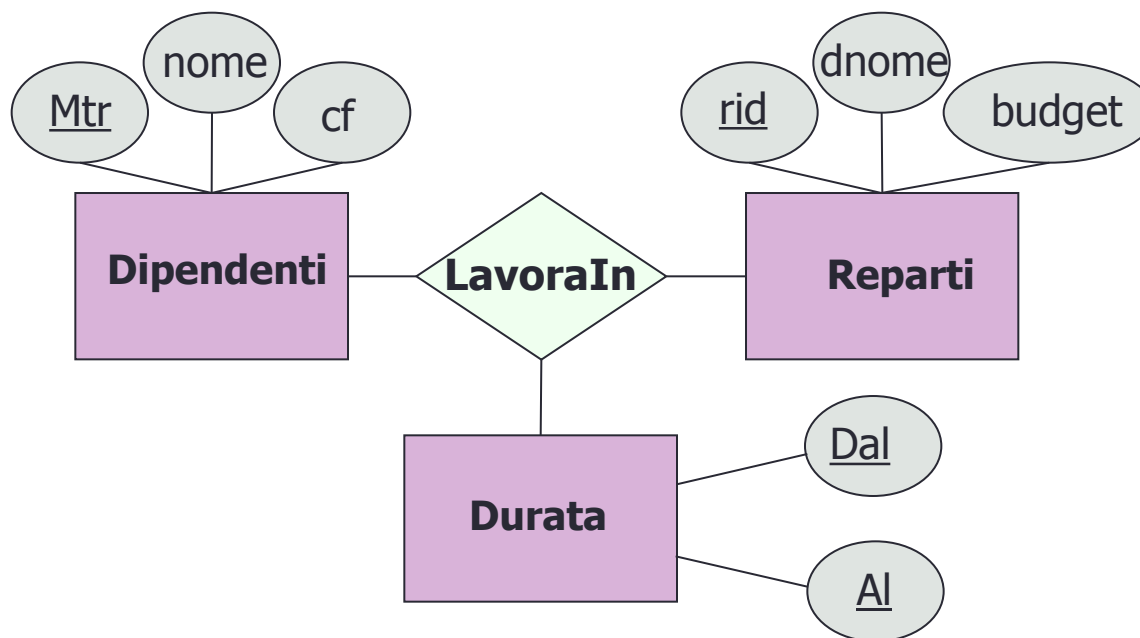
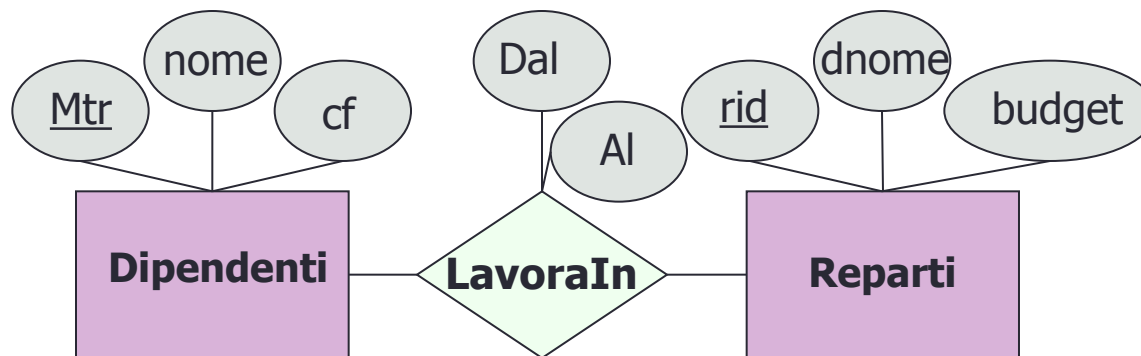
Le Relazioni possono coinvolgere anche entita' appartenenti allo stesso Insieme di Entita'.
In tal caso vengono per definiti dei RUOLI

Sono lecite relazioni tra più entità.

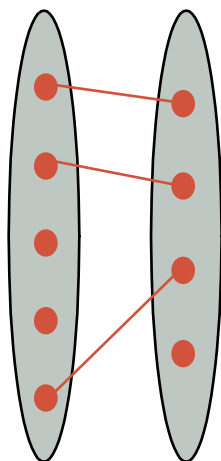
La relazione è individuata dalle chiavi delle entità coinvolte.



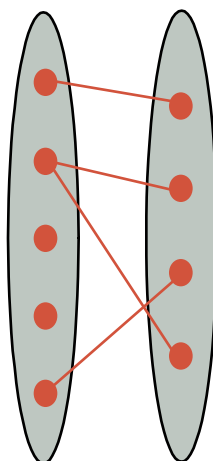
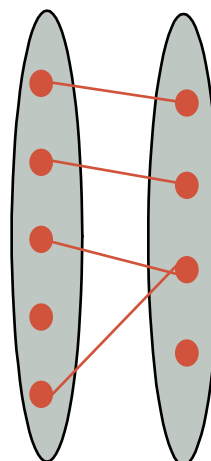
Relazioni binarie vs. ternarie



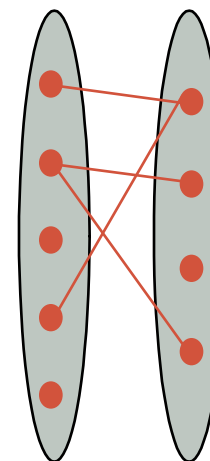
Vincolo di CHIAVE



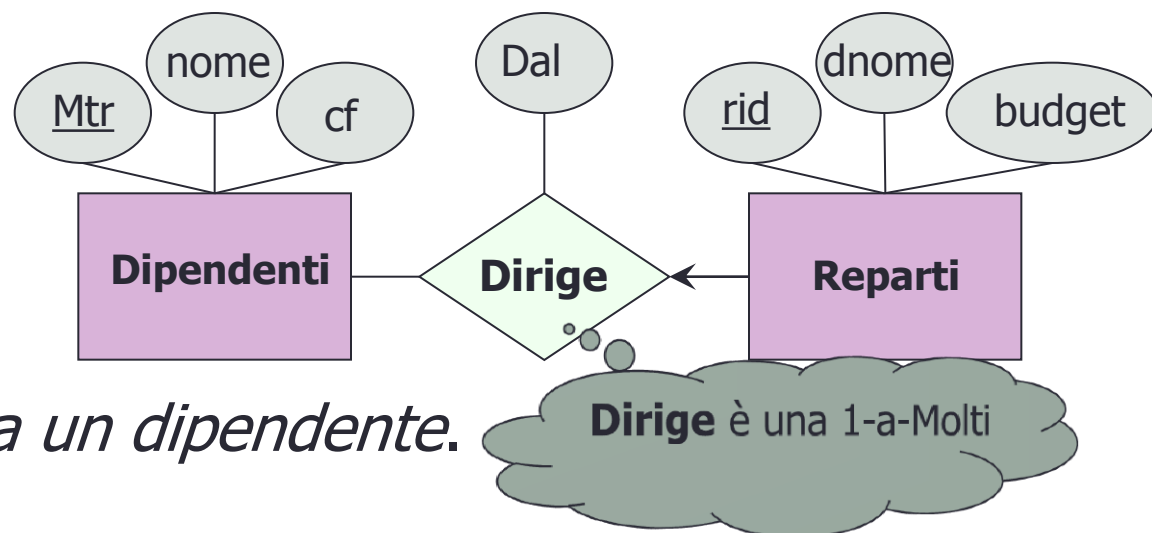
1-a-1

1-a-Molti

Molti-a-1



Molti-a-molti



VINCOLO DI CHIAVE:

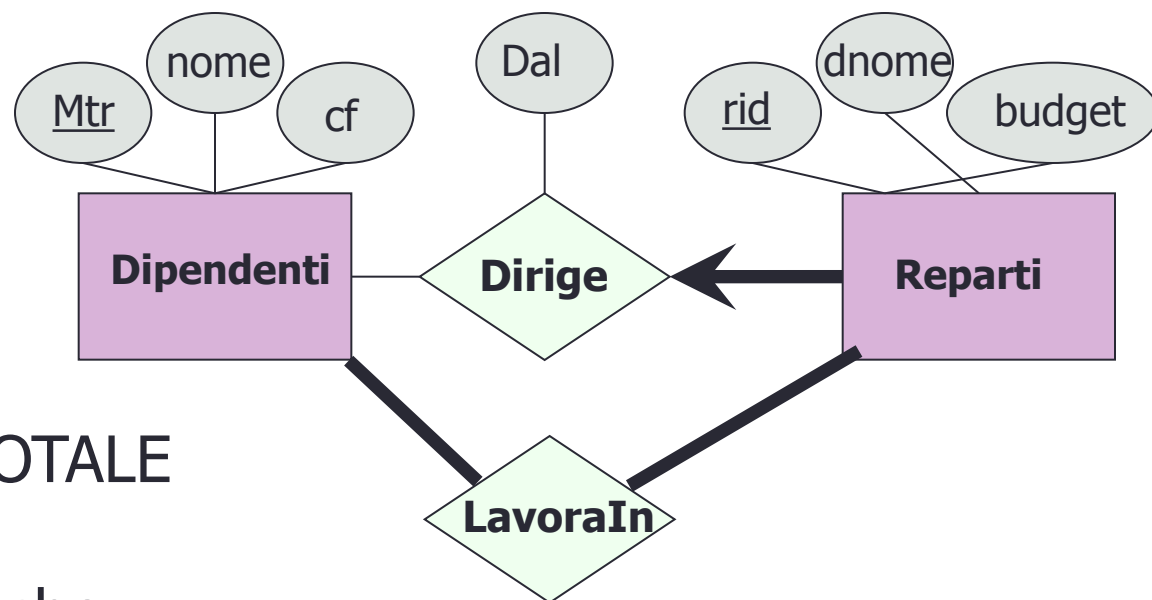
Ogni Reparto può essere gestito al più da un dipendente.

Esempi di Vincoli di Chiave

- Molti a uno:
 - GiocatoreSerie_A – gioca_per - Squadra
 - Album – contiene – Canzone_originale
- Uno-a-Molti
 - Scrittore – scrive – libro
- Molti-a-Molti
 - Studente – segue – Corso
 - Compilation – contiene - Canzone

Vincolo di PARTECIPAZIONE

Definisco un
VINCOLO DI
PARTECIPAZIONE TOTALE



Se voglio esprimere che:

- Per ogni Reparto vi deve essere esattamente un responsabile
- Ogni Dipendente lavora in almeno un Reparto
- Ogni Reparto ha almeno un Dipendente

Altrimenti la partecipazione e' detta PARZIALE.

Esempi di Vincoli di Partecipazione Totale

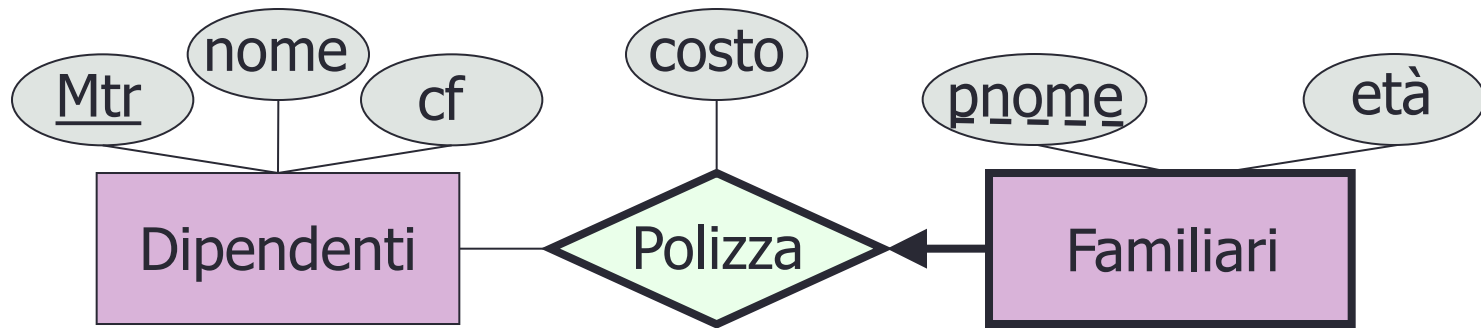
- Totale

- Scrittore vs. scrive-Libro.
- Libro vs. scritto_da - Scrittore
- Presidente vs. dirige – Azienda
- Azienda vs. diretta_da – Presidente
- Album vs. scritto_da - Musicista

- Parziale

- Studente vs. segue – Corso (nessun corso)
- Album vs. prodotto_da – Produttore (se Autore = Produttore)

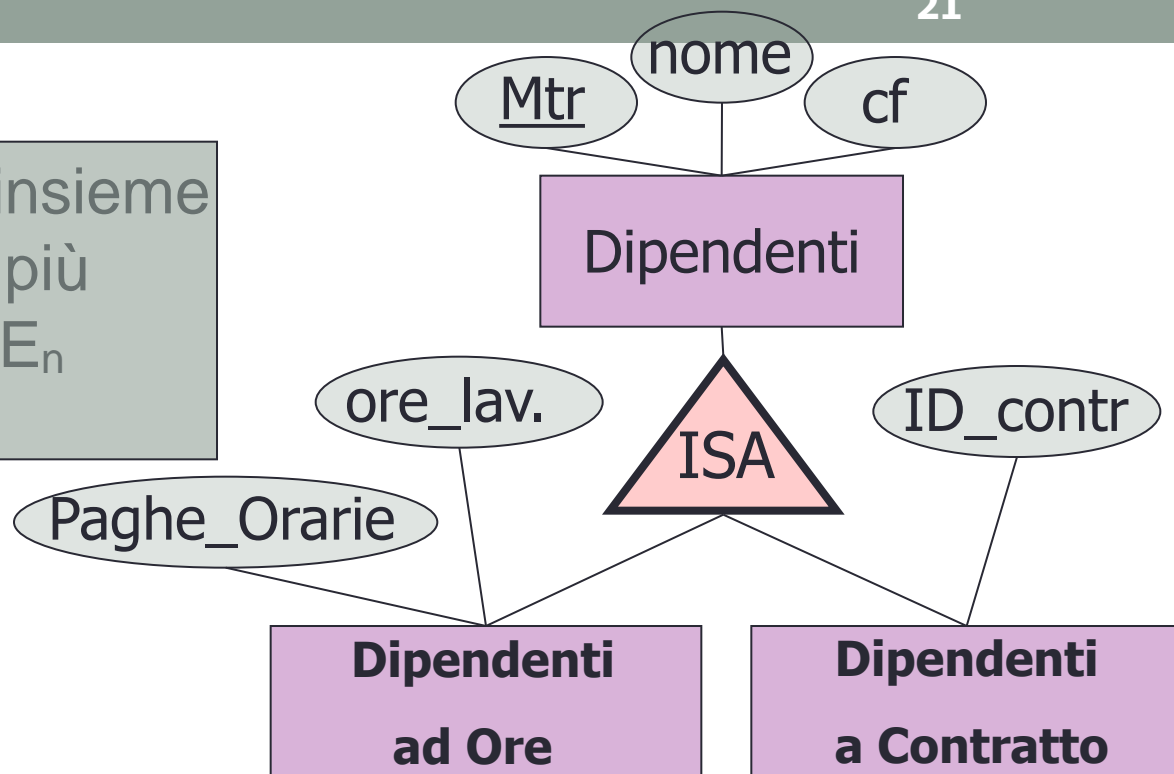
Entità Deboli (Weak Entity set)



- Un'Entità Debole puo' essere identificata solamente considerando la chiave primaria di un'altra entità (proprietario)
- L'insieme entità debole deve partecipare ad una relazione con l'insieme entità proprietario con vincolo di chiave e di partecipazione totale

Gerarchie ISA

Legame logico tra un insieme entità E detto padre e più insiemi entità E_1, E_2, \dots, E_n dette figlie.



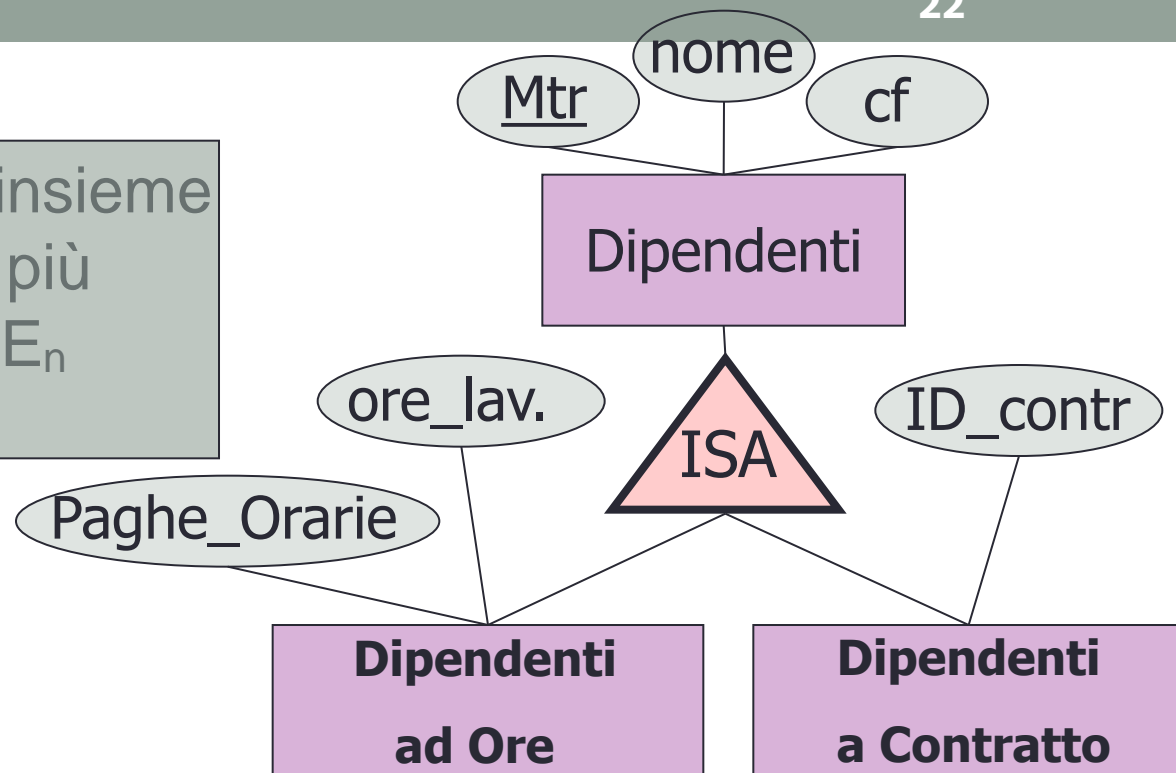
Si dice che:

- E è **GENERALIZZAZIONE** di E_1, E_2, \dots, E_n
- E_1, \dots, E_n sono **SPECIALIZZAZIONI** di E

Ei ISA E: ogni entità di E_i deve essere considerata anche un'entità di E

Gerarchie ISA

Legame logico tra un insieme entità E detto padre e più insiemi entità E_1, E_2, \dots, E_n dette figlie.

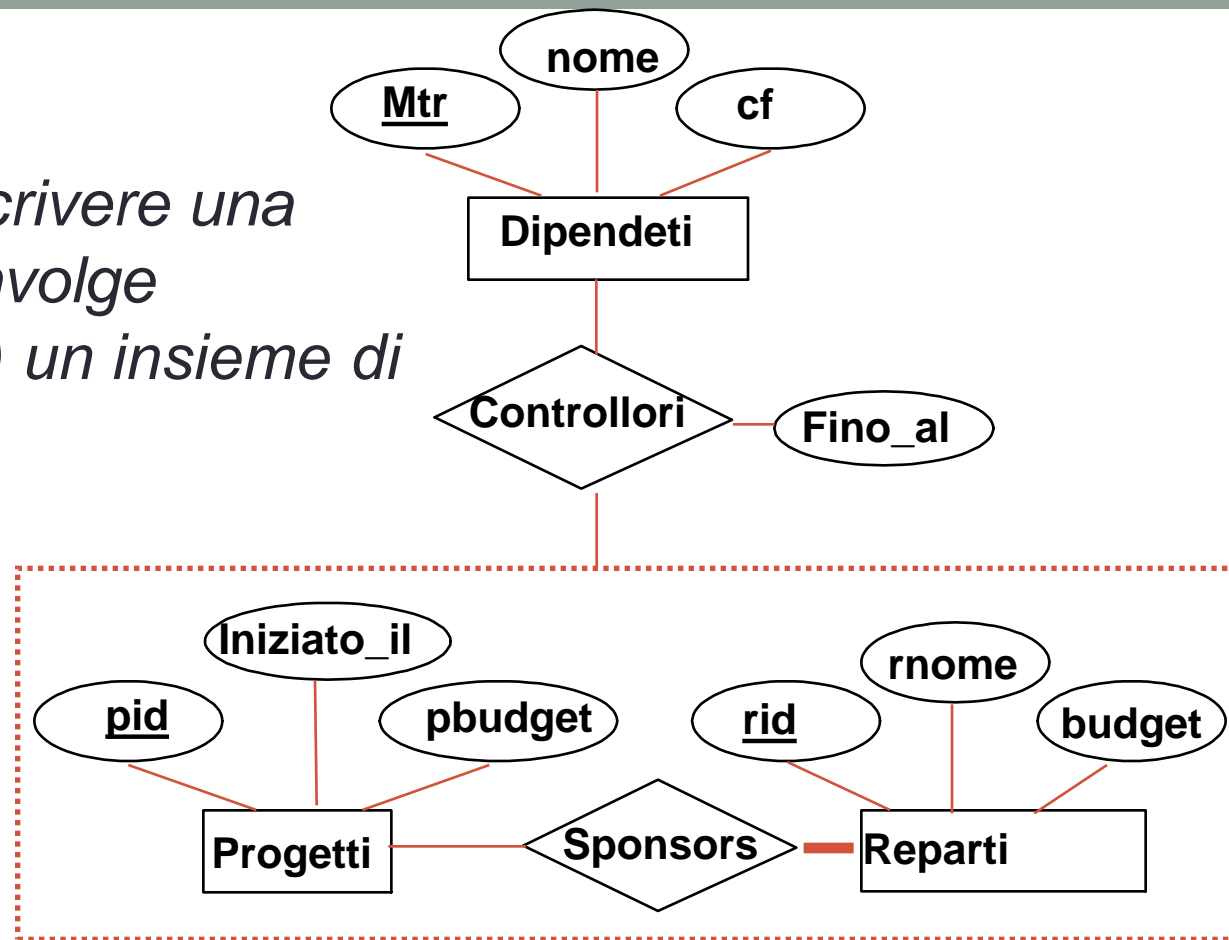


Vincolo di Sovrapposizione: sono permessi overlap tra le entità presenti negli insiemi entità figlie E_1, E_2, \dots, E_n

Vincolo di Copertura: ogni entità in E deve essere presente in almeno un insieme delle entità figlie

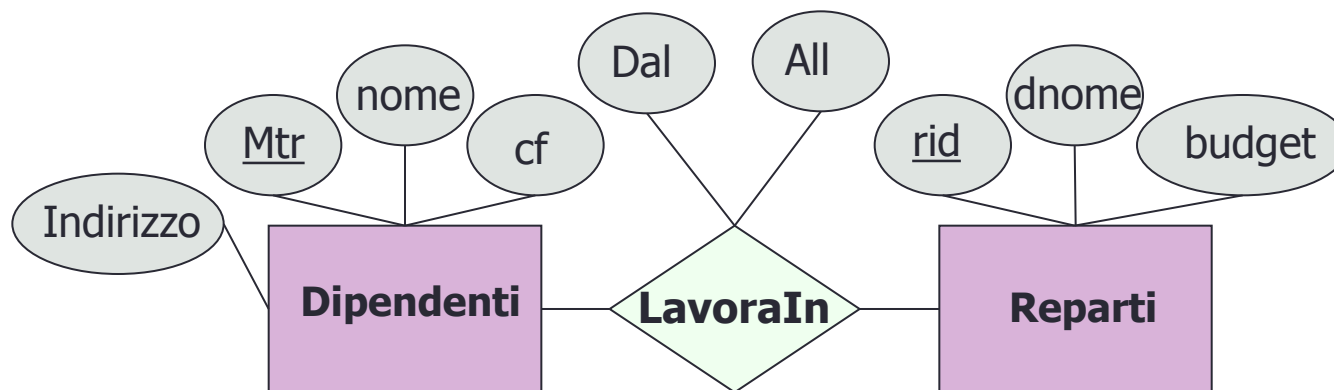
Aggregazione

- *Utilizzata per descrivere una relazione che coinvolge (insieme di entita') un insieme di relazioni*



- ◆ *Ci permette di trattare un insieme di relazioni come un insieme di entità allo scopo di permetterne la partecipazione in (altre) relazioni*

Attributi oppure Entità?



L'informazione dell'*indirizzo* del Dipendente deve essere modellato come un attributo oppure come un insieme entità?

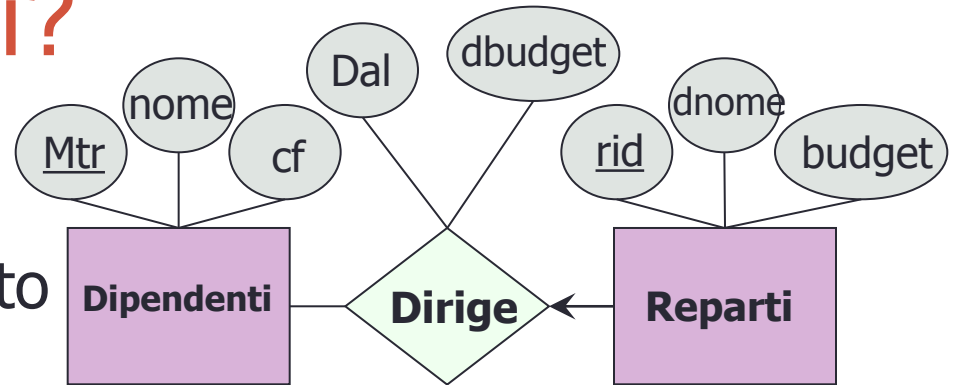
Modellerò indirizzo come un insieme entità (collegata da relazione a Dipendenti) ad esempio se:

- Ho diversi indirizzi per ogni Dipendente.
- Se l'attributo definito sull'insieme relazione deve distinguere le associazioni tra le entità coinvolte nella relazione (ad es: CV)
- Se molte interrogazioni richiedono informazioni in base a tale Attributo (ad es: tutti i dipendenti che vivono in una data città)

Entità o Relazioni?

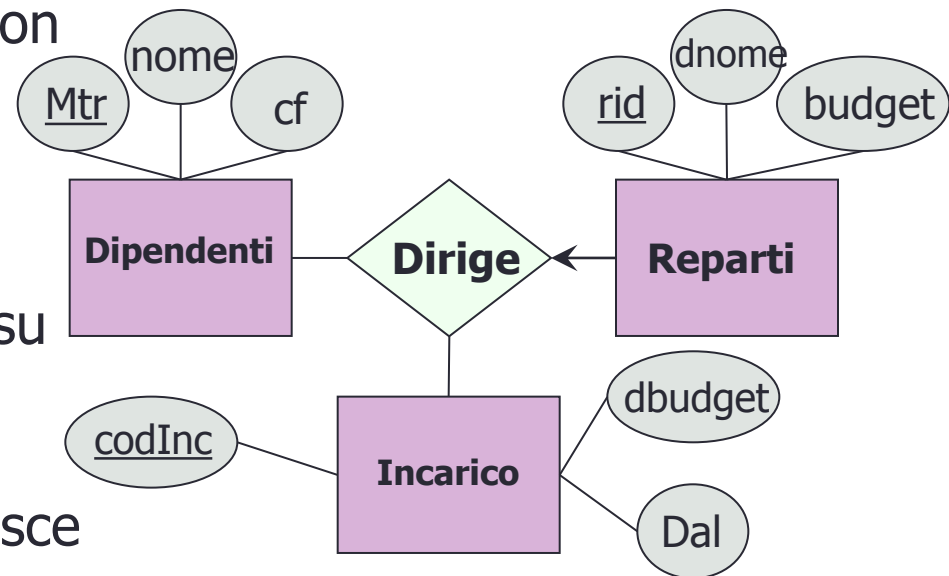


Se il direttore non ha un budget discrezionale separato per ogni Reparto che gestisce, l'attributo *dbudget* sulla relazione Dirige e' adatto?



Se il direttore ha un budget unico, con la rappresentazione descritta sopra avrei i seguenti problemi:

- Ridondanza del dato *dbudget* su tutti i Reparti gestiti da ogni dirigente.
- E' ingannevole perche' suggerisce che il budget e' legato alla relazione del dirigente con un determinato Reparto.



Relazioni Binarie o Ternarie?

◆ Se ogni Polizza è posseduta da un solo Dipendente:

- Un vincolo di chiave su Polizze significherebbe che una polizza può coprire solo un familiare!
- La mancanza di un vincolo di chiave consentirebbe a due Dipendenti di intestarsi la stessa Polizza

◆ Quali sono i vincoli aggiuntivi nel secondo diagramma?

