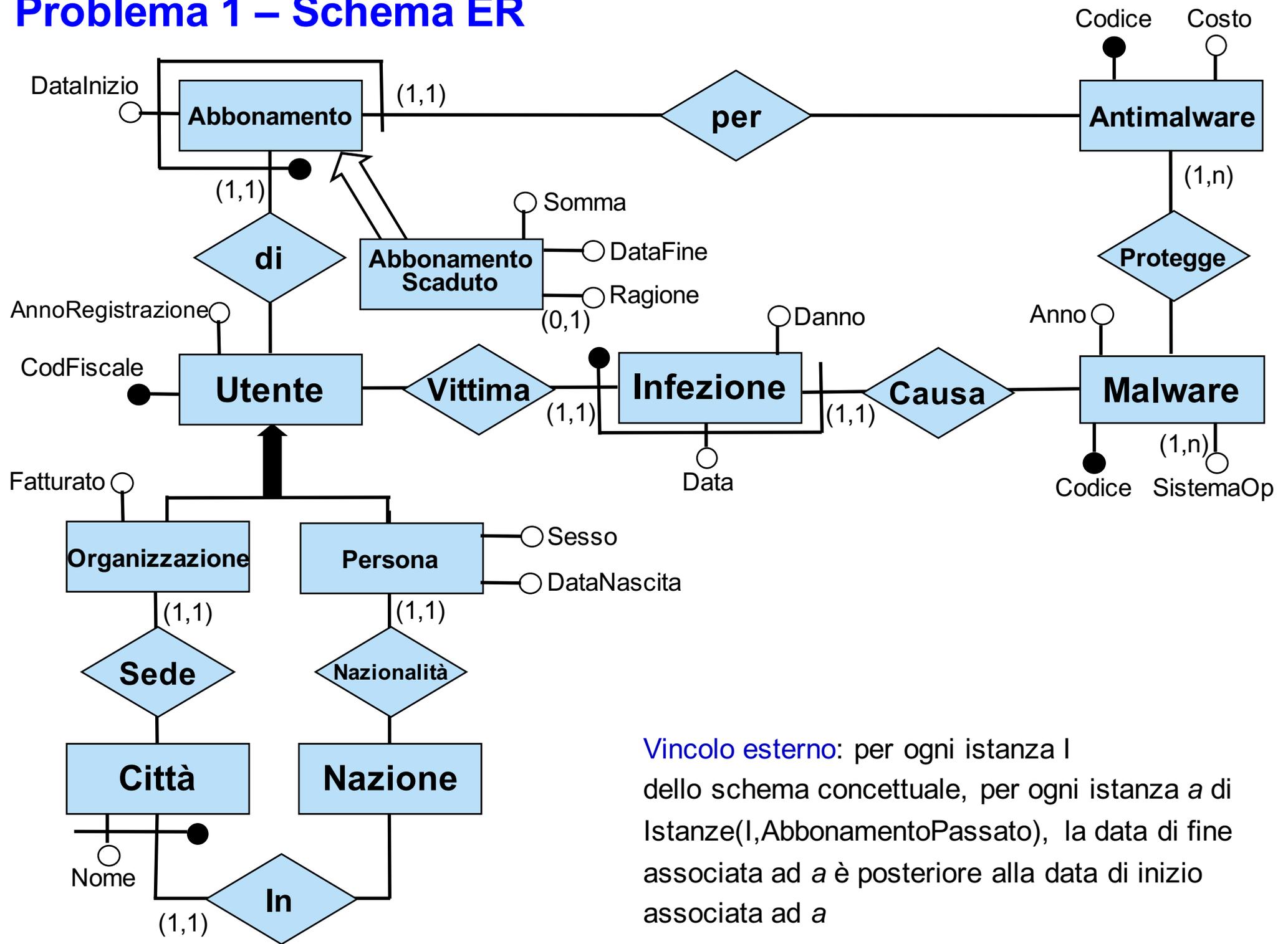


Basi di dati

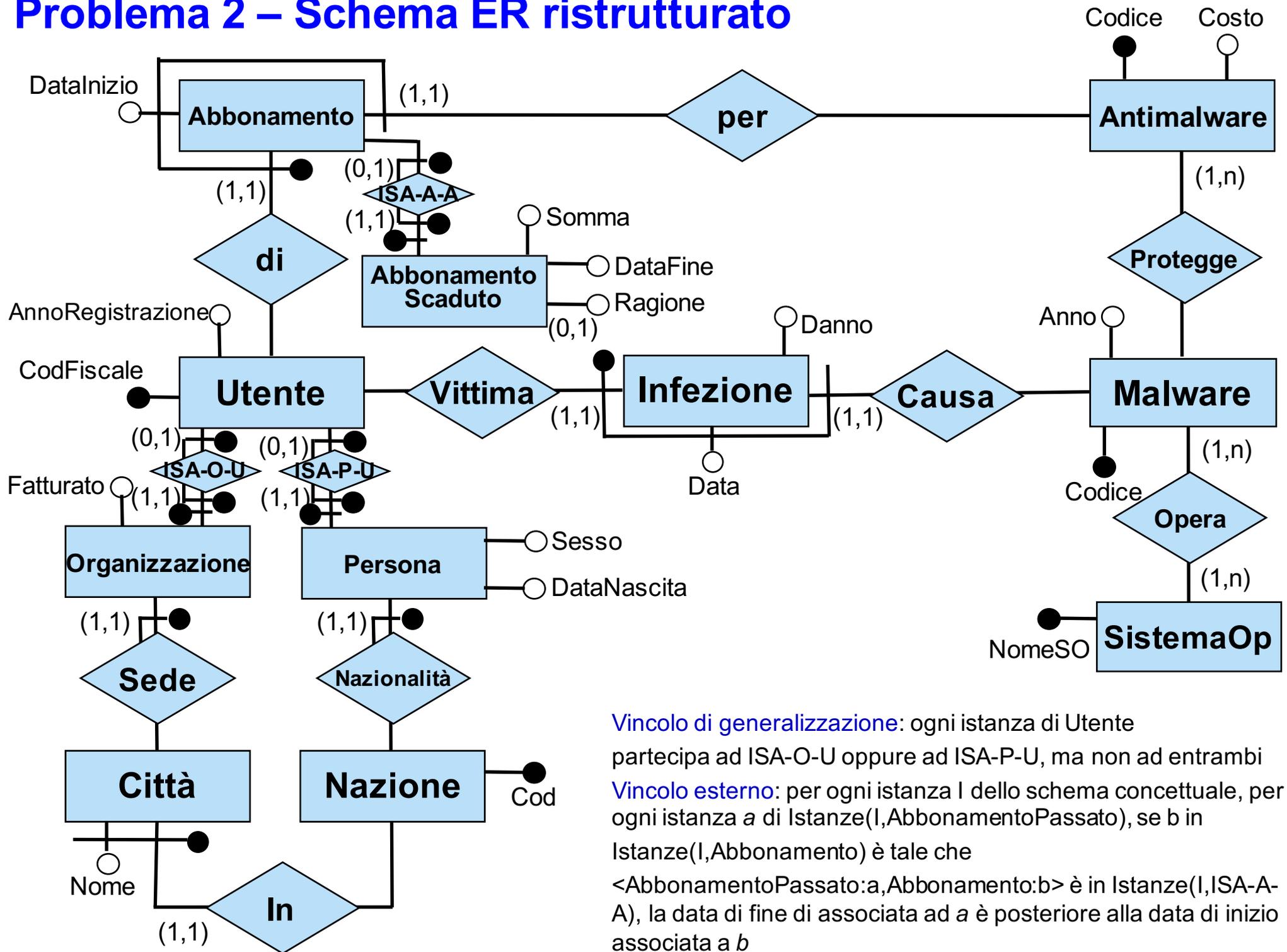
Appello del 26-01-2018

Anno Accademico 2017/18

Problema 1 – Schema ER



Problema 2 – Schema ER ristrutturato



Vincolo di generalizzazione: ogni istanza di Utente partecipa ad ISA-O-U oppure ad ISA-P-U, ma non ad entrambi

Vincolo esterno: per ogni istanza l dello schema concettuale, per ogni istanza a di Istanze(l , AbbonamentoPassato), se b in Istanze(l , Abbonamento) è tale che \langle AbbonamentoPassato: a , Abbonamento: b \rangle è in Istanze(l , ISA-A-A), la data di fine di associata ad a è posteriore alla data di inizio associata a b

Problema 2 - Schema logico dopo la traduzione diretta

Utente(CodFiscale, AnnoRegistrazione)

vincolo di generalizzazione: $Utente[CodFiscale] \subseteq Persona[CodFiscale] \cup Organizzazione[CodFiscale]$

Organizzazione(CodFiscale, Fatturato)

foreign key: $Organizzazione[CodFiscale] \subseteq Utente[CodFiscale]$

vincolo di generalizzazione: $Persona[CodFiscale] \cap Organizzazione[CodFiscale] = \emptyset$

foreign key: $Organizzazione[CodFiscale] \subseteq Sede[CodOrganizzazione]$

Persona(CodFiscale, Sesso, DataNascita)

foreign key: $Persona[CodFiscale] \subseteq Utente[CodFiscale]$

foreign key: $Persona[CodFiscale] \subseteq Nazionalità[Persona]$

Sede(CodOrganizzazione, NomeCittà, NazioneCittà)

foreign key: $Sede[CodOrg] \subseteq Organizzazione[CodFiscale]$

foreign key: $Sede[NomeCittà, NazioneCittà] \subseteq Città[Nome, CodNazione]$

Città(Nome, CodNazione)

foreign key: $Città[CodNazione] \subseteq Nazione[Codice]$

Nazione(Codice)

Nazionalità(CodPersona, CodNazione)

foreign key: $Nazionalità[CodPersona] \subseteq Persona[CodFiscale]$

foreign key: $Nazionalità[CodNazione] \subseteq Nazione[Codice]$

Malware(Codice, Anno)

inclusione: $Malware[Codice] \subseteq Opera[CodMalware]$

Opera(CodMalware, SistOp)

foreign key: $Opera[CodMalware] \subseteq Malware[Codice]$

foreign key: $Opera[SistOp] \subseteq SistemaOp[NomeSO]$

SistemaOP(NomeSO)

inclusione: $SistemaOp[NomeSO] \subseteq Opera[SistOp]$

Problema 2 - Schema logico dopo la traduzione diretta

Antimalware(Codice, Costo)

inclusione: Antimalware[Codice] \subseteq Protegge[CodAntimalware]

Protegge(CodAntimalware,CodMalware)

foreign key: Protegge[CodAntimalware] \subseteq Antimalware[Codice]

foreign key: Protegge[CodMalware] \subseteq Malware[Codice]

Infezione(CodMalware, CodUtente, Data, Danno)

foreign key: Infezione[CodMalware] \subseteq Malware[Codice]

foreign key: Infezione[CodUtente] \subseteq Utente[CodFiscale]

Abbonamento(CodAntimalware, CodUtente, DataInizio)

foreign key: Abbonamento [CodAntimalware] \subseteq Antimalware[Codice]

foreign key: Abbonamento[CodUtente] \subseteq Utente[CodFiscale]

AbbonamentoScaduto(CodAntimalware, CodUtente, DataInizio, DataFine, Somma, Ragione*)

vincolo di tupla: DataFine \geq DataInizio

foreign key: AbbonamentoScaduto[CodAntimalware,CodUtente,DataInizio] \subseteq
Abbonamento[CodAntimalware,CodUtente,DataInizio]

Problema 2 - Schema logico dopo la ristrutturazione

- *Indicazione 1*: Accorpamento di Sede a Organizzazione (sfruttando l'accoppiamento forte).
- *Indicazione 2*: Decomposizione verticale di Organizzazione per scorporare l'attributo Fatturato, simultaneo accorpamento di Utente con la proiezione di Organizzazione su Fatturato (sfruttando l'accoppiamento debole) con introduzione di un attributo Tipo, ed aggiunta degli opportuni vincoli.
- *Ulteriore ristrutturazione*: Eliminazione della relazione inutile SistemaOp (sfruttando l'accoppiamento forte).

Utente(CodFiscale, AnnoRegistrazione, Tipo, Fatturato*)

vincolo di dominio: Tipo = "persona" OR Tipo = "organizzazione"

vincolo di tupla: Fatturato is null se e solo se Tipo = "persona"

Organizzazione(CodFiscale, NomeCittàSede, NazioneCittàSede)

foreign key: Organizzazione[NomeCittàSede,NazioneCittàSede] \subseteq Città[Nome,CodNazione]

Persona(CodFiscale, Sesso, DataNascita)

foreign key: Persona[CodFiscale] \subseteq Nazionalità[Persona]

Città(Nome, CodNazione)

foreign key: Città[CodNazione] \subseteq Nazione[Codice]

Nazione(Codice)

Nazionalità(CodPersona, CodNazione)

foreign key: Nazionalità[CodPersona] \subseteq Persona[CodFiscale]

foreign key: Nazionalità[CodNazione] \subseteq Nazione[Codice]

Malware(Codice, Anno)

inclusione: Malware[Codice] \subseteq Opera[CodMalware]

Opera(CodMalware, SistOp)

foreign key: Opera[CodMalware] \subseteq Malware[Codice]

Antimalware(Codice, Costo)

inclusione: Antimalware[Codice] \subseteq Protegge[CodAntimalware]

Problema 2 - Schema logico dopo la ristrutturazione

Protegge(CodAntimalware, CodMalware)

foreign key: Protegge[CodAntimalware] \subseteq Antimalware[Codice]

foreign key: Protegge[CodMalware] \subseteq Malware[Codice]

Infezione(CodMalware, CodUtente, Data, Danno)

foreign key: Infezione[CodMalware] \subseteq Malware[Codice]

foreign key: Infezione[CodUtente] \subseteq Utente[CodFiscale]

Abbonamento(CodAntimalware, CodUtente, DataInizio)

foreign key: Abbonamento [CodAntimalware] \subseteq Antimalware[Codice]

foreign key: Abbonamento[CodUtente] \subseteq Utente[CodFiscale]

AbbonamentoScaduto(CodAntimalware, CodUtente, DataInizio, DataFine, Somma, Ragione*)

vincolo di tupla: DataFine \geq DataInizio

foreign key: AbbonamentoScaduto[CodAntimalware,CodUtente,DataInizio] \subseteq
Abbonamento[CodAntimalware,CodUtente,DataInizio]

Vincoli esterni

- Persona[CodFiscale] = select CodFiscale from Utente where Tipo = "persona"
- Organizzazione[CodFiscale] = select CodFiscale from Utente where Tipo = "organizzazione"

Viste

- View SistemaOp = select SistOp from Opera
- View Sede = select codFiscale, NomeCittà, RegioneCittà from Organizzazione

Problema 3: soluzione

```
select partito, count(*), min(età)
from (select codice
      from partito p
      where not exists (select codice
                       from partito
                       where schieramento = p.schieramento and
                             codice <> p.codice)
      ) v join candidato c on v.codice = c.partito
group by c.partito
```

Problema 4: soluzione

$M(\underline{P}, N)$

inclusione $M[P] \subseteq F[M]$

$F(\underline{H}, \underline{M}, G, D)$

foreign key $F[M] \subseteq M[P]$

$E(\underline{C}, A, B)$

foreign key $E[C] \subseteq R[E]$

$R(\underline{E}, F1, F2)$

foreign key $R[E] \subseteq E[C]$

foreign key $R[F1, F2] \subseteq F[H, M]$

Vincolo esterno

- Nell'equi-join tra R ed E con la condizione $R.E = E.C$, gli attributi $\langle A, F1, F2 \rangle$ formano una chiave, nel senso che non ci possono essere due tuple diverse con la stessa combinazione di valori per questi attributi.