Prima iterazione

Analisi e progettazione del software

Flavio Colacicchi

13/03/2025

L'analisi enfatizza un'investigazione di un problema e dei suoi requisiti, ovvero si focalizza sul che cosa e non sul come, e non è direttamente interessata alle soluzioni del problema.

Esistono diversi modi per effettuare l'analisiciascuno dei quali è in qualche modo legato a una "strategia risolutiva" che si vuole adottare o a uno specifico "tipo di sistema" che si vuole realizzare

L'analisi orientata agli oggetti (OOA) si basa principalmente sull'identificazione dei concetti nel dominio del problema e su una loro descrizione a oggetti

L'analisi del software ha lo scopo di comprendere il problema e si deve focalizzare su tre aspetti

1. Le inormazioni che il sistema deve gestire (dominio informativo)
   Rappresentato attraverso il modello di dominio
2. Le funzioni (o operazioni) che il sistema dovrà gestire (alterazioni al dominio informativo)
   Rappresentate con operazioni di sistema e diagrammi di sequenza di sistema
3. Il comportamento del sistema (come cambia la rappresentazione del sistema a seguito di operazioni)
   Rappresentato tramite contratti

Lo scopo della modellazione è comprendere e favorire la comunicazione.

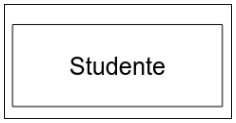
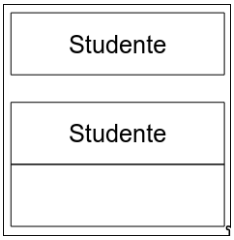
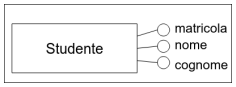
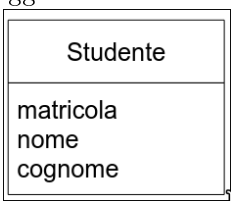
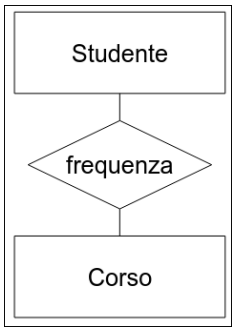
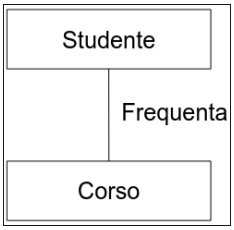
Il dominio del problema è simile alla progettazione concettuale delle basi di dati:

- Rappresentare le specifiche formali della realtà di interesse
- In termini di una descrizione formale e compatta
- Indipendente dai scriteri di rappresentazione utilizzati nei sistemi di gestione

Mentre nella modellazione del dominio

-

Basi di dati	Ingegneria del software
I diagrammi si chiamano schemi	I diagrammi si chiamano modelli
Formalismi per esprimere schemi si chiamano modelli (e.g. Schemi E.R.)	I formalismi per esprimere i modelli si chiamano linguaggi (e.g. UML)

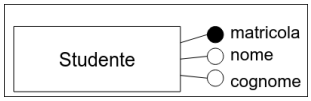
ER	UML
Un'entità rappresenta una classe di oggetti che hanno proprietà comune	Una classe concettuale rappresenta un insieme di cose o concetti con caratteristiche simili
	
Le relative istanze sono chiamate istanze o occorrenze	Le relative istanze sono chiamate istanze o oggetti
Un attributo descrive una proprietà elementare di un'entità o di una relazione	Un attributo rappresenta una proprietà elementare degli oggetti di una classe
	
Una relazione rappresenta un legame logico tra due o più entità	Un'associazione rappresenta una relazione (una connessione significativa) tra classi e le istanze si chiamano collegamenti
	

Le relazioni possono essere binarie o anche N-arie

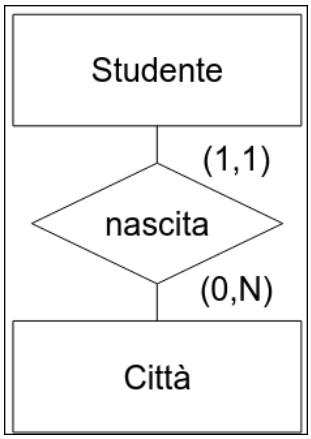
Le relazioni possono avere attributi

Il nome è generalmente un sostantivo che indica una relazione

Per ciascuna entità va indicato (almeno) un identificatore



Una cardinalità caratterizza la partecipazione (minima e massima) di un'entità a una relazione

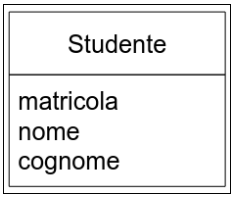


Le associazioni sono in genere binarie e le associazioni N-arie sono possibili, ma poco comuni

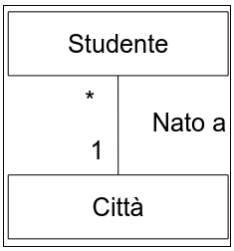
È possibile rappresentare associazioni con attributi, ma è poco comune

Il nome è generalmente un verbo che indica una relazione

È possibile associare identificatori alle classi, ma è poco comune



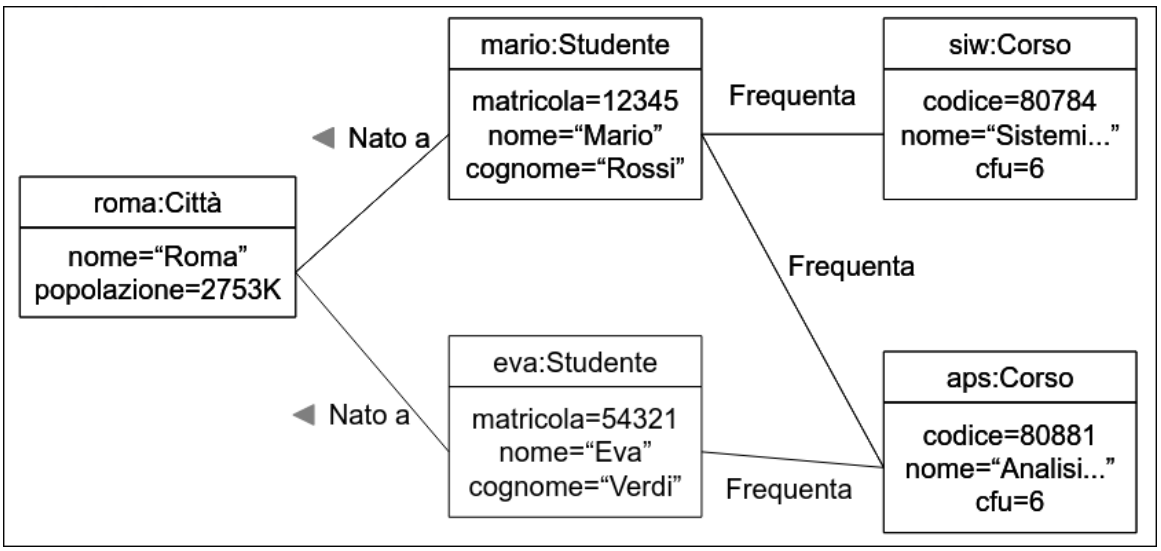
Una molteplicità indica quante istanze di una classe possono essere associate a un'istanza dell'altra classe



(A,B)	A..B
(A,A)	A
(0,N)	*
(1,N)	1..*

Per rappresentare solo informazioni che devono essere gestite in modo persistente	Per rappresentare tutte le informazioni che devono essere gestite da un'applicazione
Un'entità rappresenta sia una classe di istanze che il relativo "insieme"	Una classe rappresenta una classe di oggetti ma non il relativo "insieme"
In genere nessuna entità ha una sola istanza	Può essere ok avere classi che hanno un solo oggetto
Le entità hanno in genere almeno un attributo	Può essere ok avere classi senza attributi con cautela
Le entità rappresentano informazioni	Può essere ok avere classi che rappresentano solo comportamento con cautela

Esempio di rappresnetazione tramite UML di un dominio



Notare che i **:** sono **obbligatori per la definizione di un oggetto** mentre sono **vietati denna definizione di una classe**