

Programación Orientada a Objetos

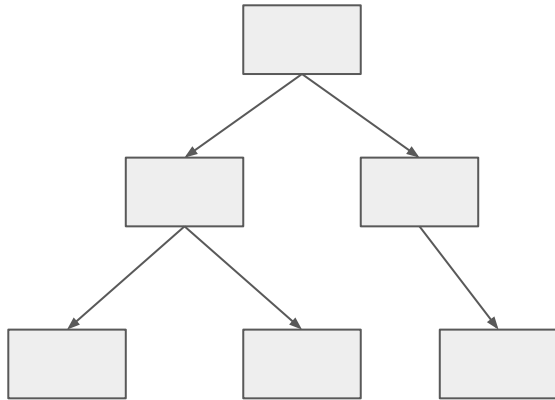
Introducción

Contenidos

- Universo del discurso
- Ontología
- El paradigma OO
- Objetos, instancias y clases
- Reusabilidad y modularidad
- Bibliografía

Universo del discurso

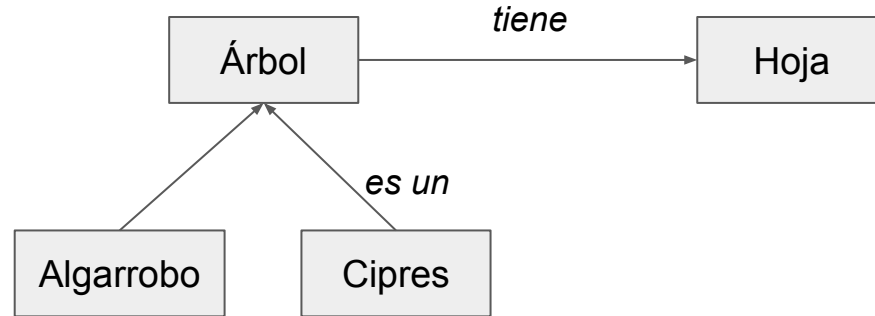
Modelo. *Es una representación abstracta de una realidad particular. Se trata de darle significado a una representación abstracta. Grafos, tablas, ecuaciones, árboles, conjuntos, etc. adquieren un significado particular según la idea que se intenta expresar.*



Puede ser un árbol de decisiones o un árbol de taxonomías

Universo del discurso

Entidades. Son los sujetos del modelo. Pueden ser concretas o abstractas. Interactúan con otras entidades. Forman parte de una jerarquía. Tienen propiedades. Pueden componer o ser compuestas.



Universo del discurso

Es el conjunto de cosas acerca de las cuales se habla en un determinado contexto. Dependiendo del dominio de discurso, una misma proposición podrá ser verdadera o falsa.

Por convención, el dominio de discurso es siempre un conjunto no vacío.

*En la teoría de **modelos**, el universo de discurso es el conjunto de **entidades** en que un modelo se basa.*

Ontologías

Es la rama de la filosofía que estudia lo que hay, así como las relaciones entre los entes (por ejemplo, la relación entre un universal —como el rojo— y un particular que lo "tiene" —como una manzana) o la relación entre un acto (como el que Sócrates bebiera la cicuta) y sus participantes (Sócrates y la cicuta).

Aplicaciones como Web Semantica

Nuestro interes se basa en su habilidad como concepto para describir el Universo del Discurso.

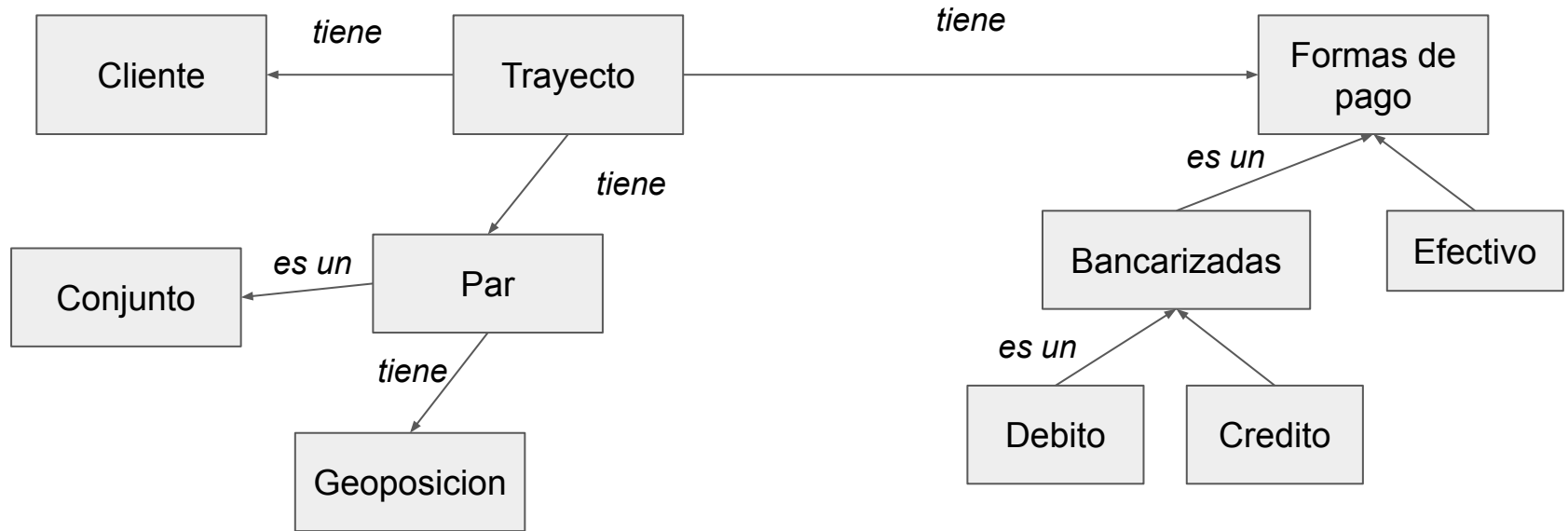
Aplicación

Una remisería tiene varios vehículos a su disposición, incluso tiene una camioneta para pequeños fletes. Los clientes solicitan viajes desde un origen a un destino dentro de la ciudad. Abonan el servicio con tarjeta de débito, crédito o efectivo. El dueño de la remisería, Juan Carlos, desea implementar un sistema que le organice todo, porque la planilla Excel que comparte con las dos secretarias se ha vuelto muy compleja.

Describir las entidades mas importantes para un modelo para este U.D., sus interacciones y relaciones

Aplicación - Remiseria

Posible modelo



Paradigma Orientado a Objetos

Parte del concepto de "objetos" como base, los cuales contienen información en forma de atributos o propiedades y código en forma de métodos.

Tiene una base natural próxima a las ontologías, de ahí su ventaja en el diseño e implementación

Por lo tanto no requiere un mapeo a estructuras preexistentes o generales.

Todo es visto como objetos y relaciones e interacciones entre ellos

Programación Orientado a Objetos

Comparación con otro diseño

Representación de un juego de Ta-Te-Ti

Tradicional, el tablero es una matriz cuyos valores representan al jugador.

POO. Se define a un tablero como un objeto compuesto de celdas, con adyacencias, que pueden contener o no una ficha que a su vez representa a un jugador.

Desventajas: Evidentemente, mayor diseño conceptual.

Ventajas: El diseño representa mejor la realidad que se analiza

Objeto - Primer acercamiento

En concreto, y en un primer acercamiento a la definición podemos decir que:

Es una estructura de datos que a su vez tiene funcionalidades.

Ejemplo: Auto, tiene color, velocidad, cantidad de ruedas, y tiene un comportamiento: encender, acelerar, detener.

Auto
color: int velocidad: double
iniciar() acelerar() detener()

Clase

Es una abstracción de una entidad observable, a nuestro criterio.

Componen o pueden componer una jerarquía de clases, donde aparecen las *super-clases* y las *sub-clases* o *sub-tipos*.

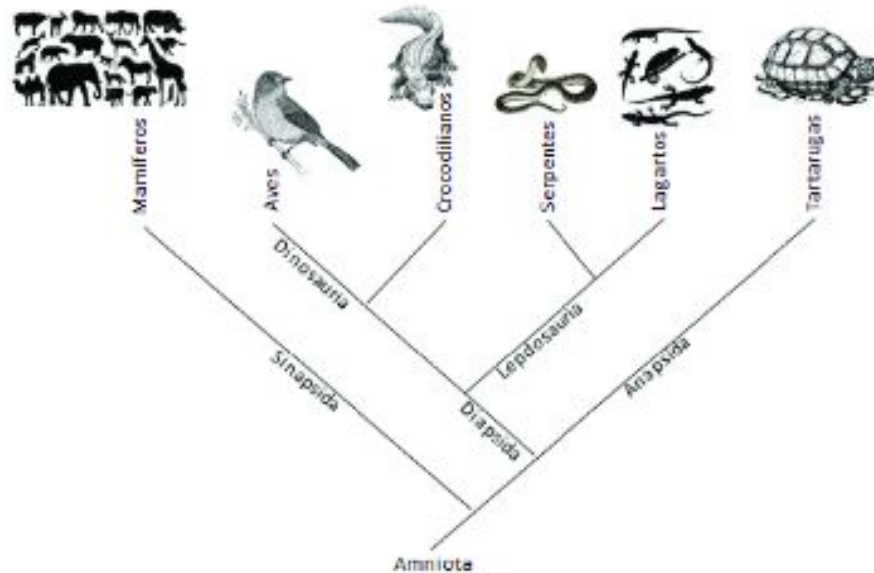
Clase Libro, tiene título, autores, páginas, género, etc. Es un tipo de material impreso ?

Un ejemplar particular, existente en una biblioteca es una *instancia* de ese libro conceptual. Por lo tanto:

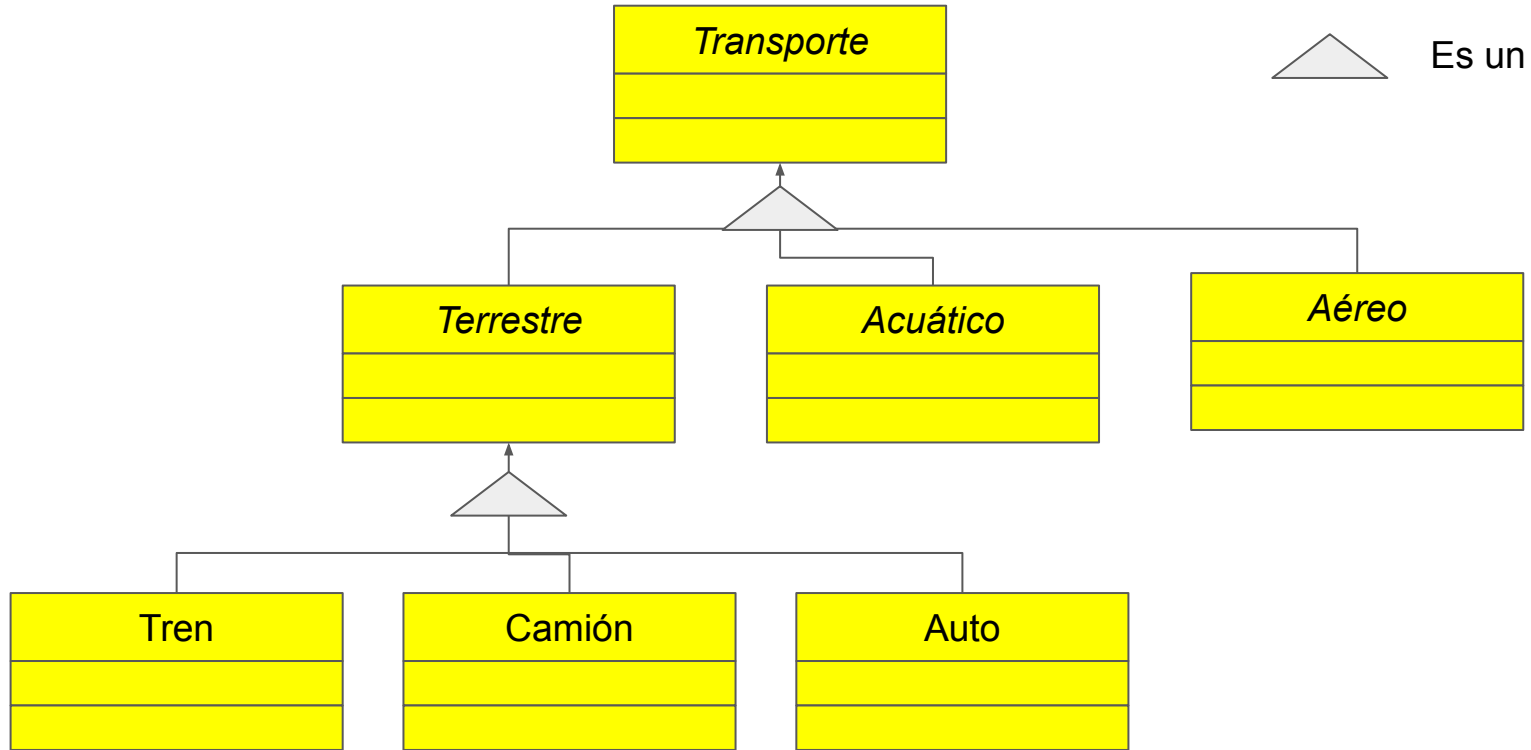
Un objeto es una instancia de una clase determinada

Clase - Jerarquía

La tendencia normal es pensar las clases en función de sus generalizaciones, para poder agruparlas. Hay muchos ejemplos en la biología.

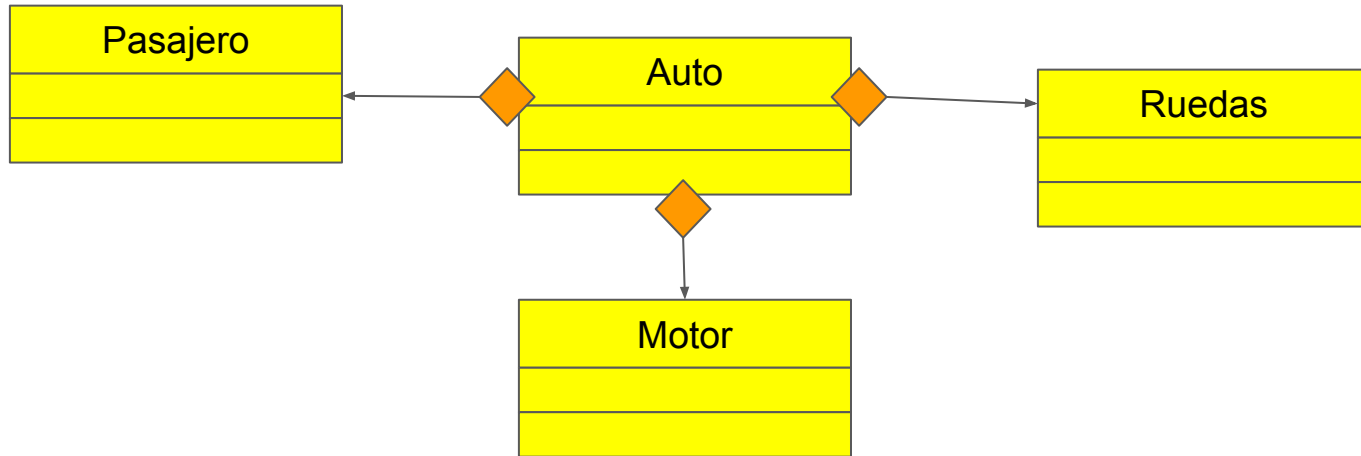


Clase - Jerarquía



Clase - Composición

Básicamente. Luego veremos en mas detalle.



Reusabilidad

Fundamentalmente la herencia que se desprende de una jerarquía, provee una enorme reusabilidad del código.

Todos los métodos que se describen en una *super-clase* se reutilizan en las subclases.

Bibliografía

- [1] Francisco Morero. “Introducción a la OOP”. Grupo EIDOS, 1999-200
(<https://kataix.umag.cl/~ruribe/Utilidades/Introduccion%20a%20la%20Programacion%20Orientada%20a%20Objetos.pdf>)
- [2] Meyer, B. “Object Oriented Software Construction”. Prentice Hall, 1988.
- [3] Design Patterns, Elements of Reusable Object-Oriented Software. E. Gamma, R. Helm, R. Johnson, and J. Vlissides. Addison-Wesley. Massachusetts, 1994.

Próxima Clase

- Mensajes. Métodos.
- Variables de instancia.