

La complejidad del Software

Es una propiedad esencial del software, depende 4 partes:

- El *dominio* del problema → Requisitos muy distintos, cambiantes, contradictorios.
- El *proceso* de desarrollo → Intervienen muchos desarrolladores, se necesita coordinar y utilizar una tecnología en común.
- La *flexibilidad* en el software → Sistema apto para cambiar con los requerimientos del cliente
- La *reutilización* del código → No reinventar la rueda.





Como lo explicó
el Cliente



Como lo entendió
el Líder de Proyecto



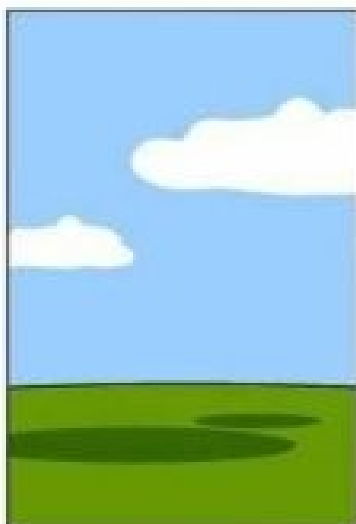
Como lo diseñó
Ingeniería



Como lo escribió
el Programador



Como lo describió
el Vendedor



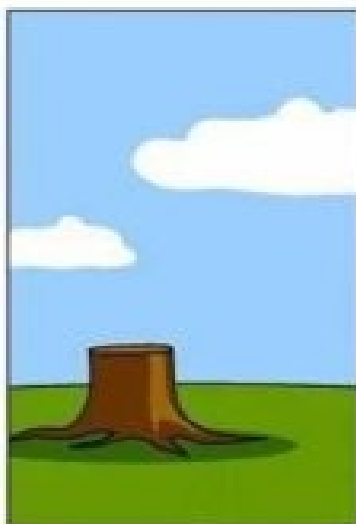
Como el proyecto ha
sido documentado



Que se instaló



Que se le facturó
al Cliente



Como fué el soporte
Técnico



Que necesitaba el
Cliente realmente.

Programación Orientada a Objetos

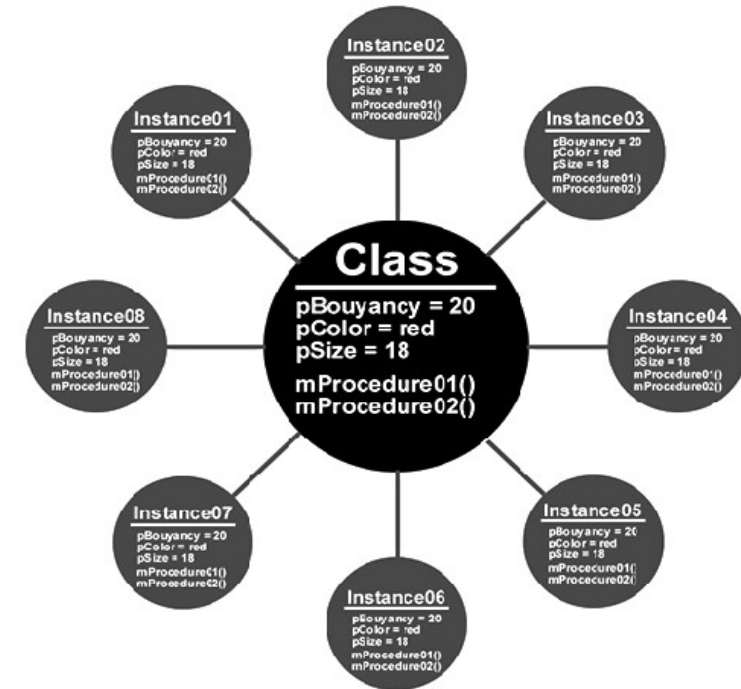
Es un paradigma de programación, es decir una forma de *analizar, diseñar y realizar soluciones*.

Objetivos:

- Escribir software fácilmente modificable y escalable.
- Escribir software reusable.
- Disponer de un modelo natural para representar un dominio.

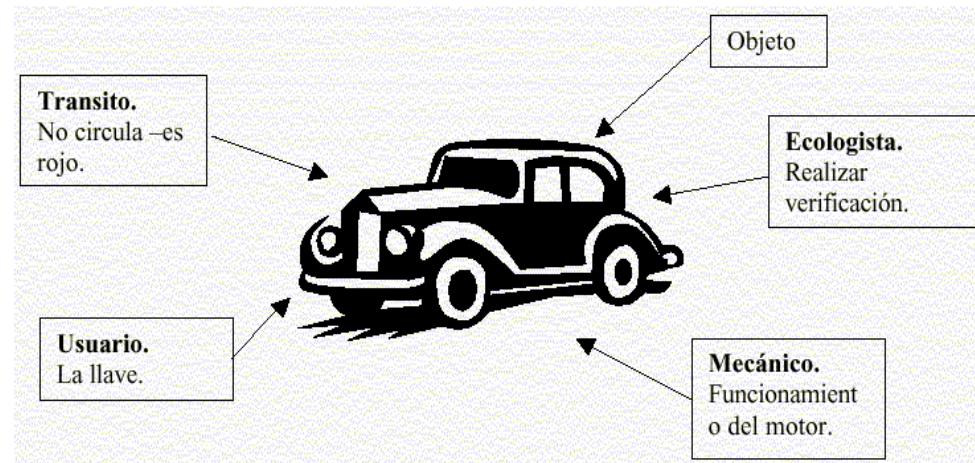
Ventajas:

- Fomenta la reutilización del software.
- El software desarrollado es más flexible al cambio.
- Es más cercano a pensar a la forma de las personas.



¿Qué es la abstracción?

- Consiste en aislar un elemento de su contexto o del resto de los elementos que lo acompañan.
- Denota según el **observador** las características esenciales de un objeto que lo distinguen de los demás.
- La vista exterior del objeto puede definirse como un contrato, el cual el mismo se compromete a cumplir y los demás objetos dependen.
- Debemos identificar propiedades del objeto (*atributos*) y comportamientos (*métodos*).



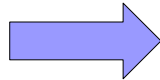
Objetos y Clases

Clases



- Modela una abstracción de los objetos.
- Define atributos y comportamientos.
- Es el plano o molde que define un objeto.

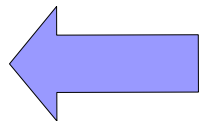
Objetos



- Está modelado *en función* de una clase.
- Es una *instancia* única de una clase.
- Retiene la estructura y el comportamiento de una clase.

¿Qué es un Objeto?

- Es una entidad que combina procedimientos e información.
- Ejecuta operaciones (***comportamiento***) y almacena información (***estructura***).
- Los procedimientos o acciones que puede realizar un objeto se denominan ***métodos***.
- La información que almacena un objeto sobre su estado se denomina ***atributos***.



Objeto

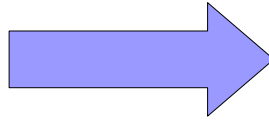
Los datos describen el *estado del objeto*.

Los procedimientos actúan sobre los datos del objeto, *modificándolos o brindando* información

¿Qué es un Objeto?

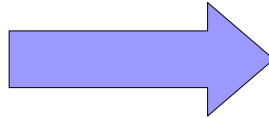
Un objeto posee...

Estado



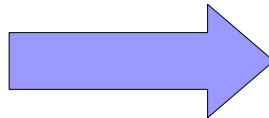
Esta dado por el conjunto de propiedades que posee y su valor en un momento dado.

Comportamiento



Está dado por un conjunto de acciones que el mismo puede realizar.

Identidad



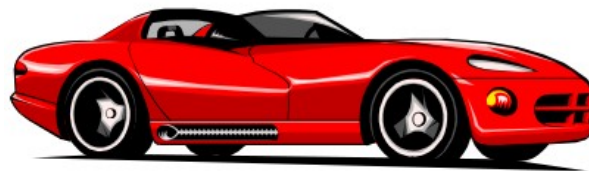
En todo momento es diferenciable del resto y es constante.

Estado de un Objeto

- Está dado por el conjunto de propiedades que el mismo posee.
- Es deseable encapsular el estado de un objeto, ocultando su representación al mundo exterior.
- Ningún objeto puede manipular directamente el estado de otro.

Comportamiento de un Objeto

- Determina como éste actúa y reacciona en términos de cambios de estado y solicitudes.
- Representan la interfaz por la cual se puede interactuar con el objeto.
- Los métodos asociados a un objeto comprenden el protocolo del mismo.



■ Atributos:

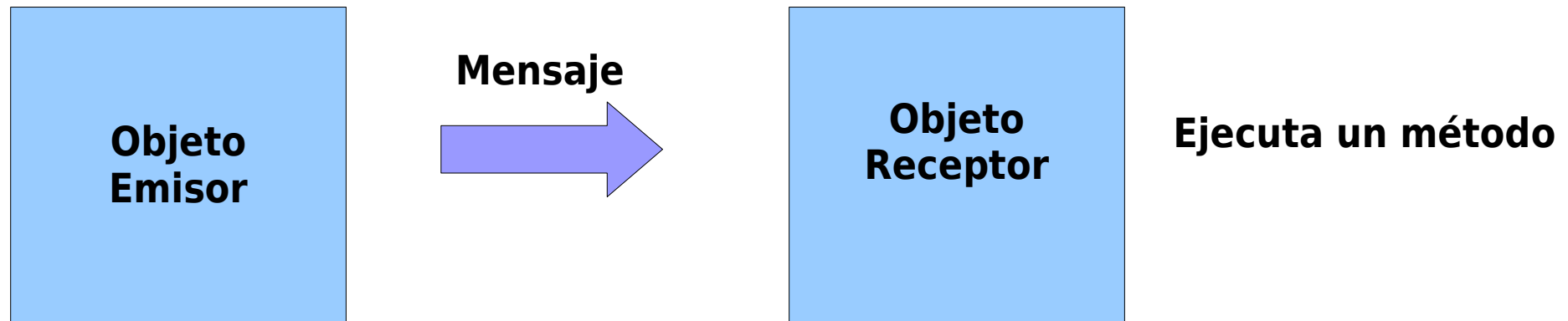
- color
- velocidad
- ruedas
- motor

■ Métodos:

- arranca()
- frena()
- dobla()

Los objetos interactúan a través de mensajes

- Los objetos se comunican mediante el envío de mensajes.
- El objeto emisor se debe ***enlazar*** o ***asociar*** al objeto receptor.
- El emisor del mensaje solicita al receptor que realice una operación.
- La llamada al método siempre se produce en el contexto de un objeto concreto.

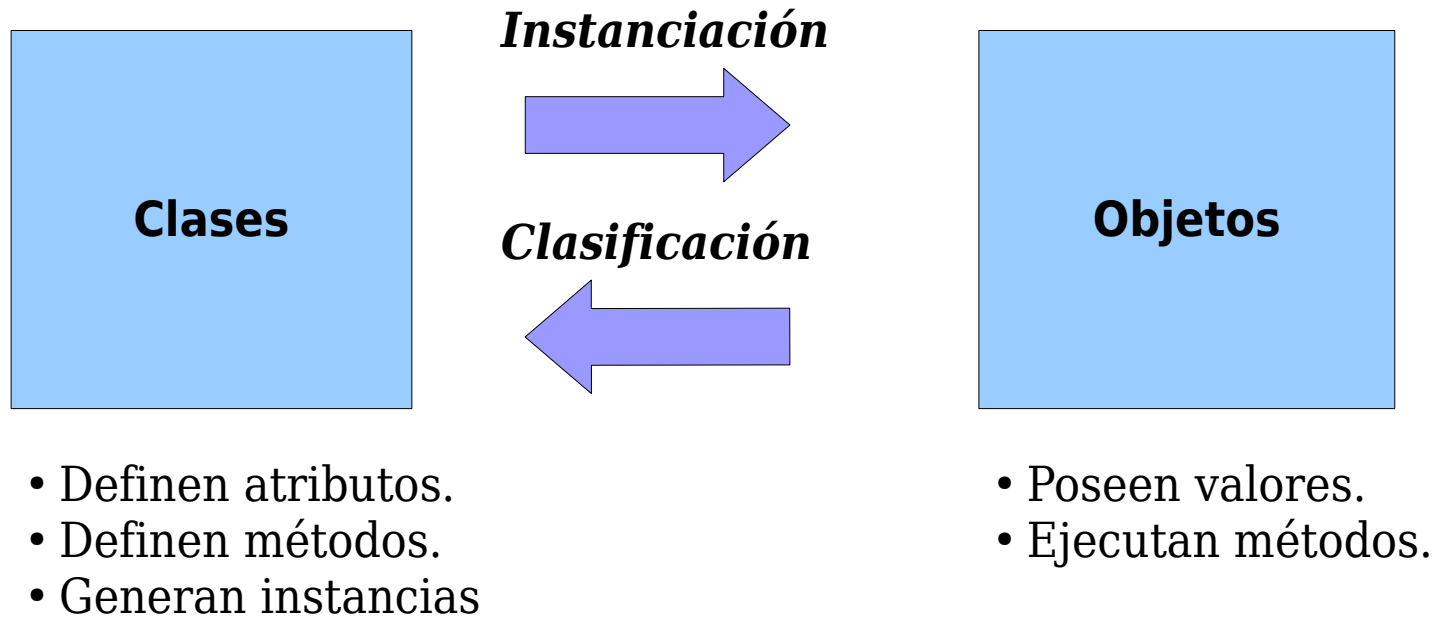


¿Qué es una Clase?

- Mecanismo utilizado en la POO para abstraer conceptos (clasificación).
- Describe las características comunes a todos los objetos que pertenecen a ella.
- Especifica:
 - ✓ El ***comportamiento*** de los objetos.
 - ✓ La ***estructura interna*** de los objetos.
 - ✓ La definición e implementación de todas sus ***acciones***.

Diferencias entre Clases y Objetos

- Las clases son ***definiciones estáticas*** que se pueden utilizar para entender a todos los objetos de una clase
- Los objetos son ***entidades dinámicas*** que existen en el mundo real.



Los atributos de un Objeto mantienen su estado

- Cada objeto conoce su estado actual.
- Un atributo es un fragmento de información que indica el estado de un objeto.



Birome

Atributos

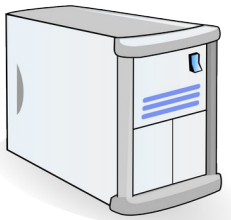
- Color
- **Marca**
- **Cantidad de Tinta**



Motocicleta

Atributos

- Modelo
- **Velocidad Máxima**
- **Cilindrada**



CPU

Atributos

- Microprocesador
- **Cantidad RAM**
- **Espacio en Disco**

Los Objetos tienen comportamientos

- Un objeto existe para proporcionar un comportamiento (funcionalidad) al sistema.
- Cada comportamiento se denomina ***operación*** o ***método***.



Birome

Métodos

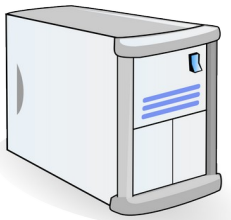
- Escribir
- ***Cambiar precio***



Motocicleta

Métodos

- Doblar
- Acelerar
- Frenar
- Estacionar



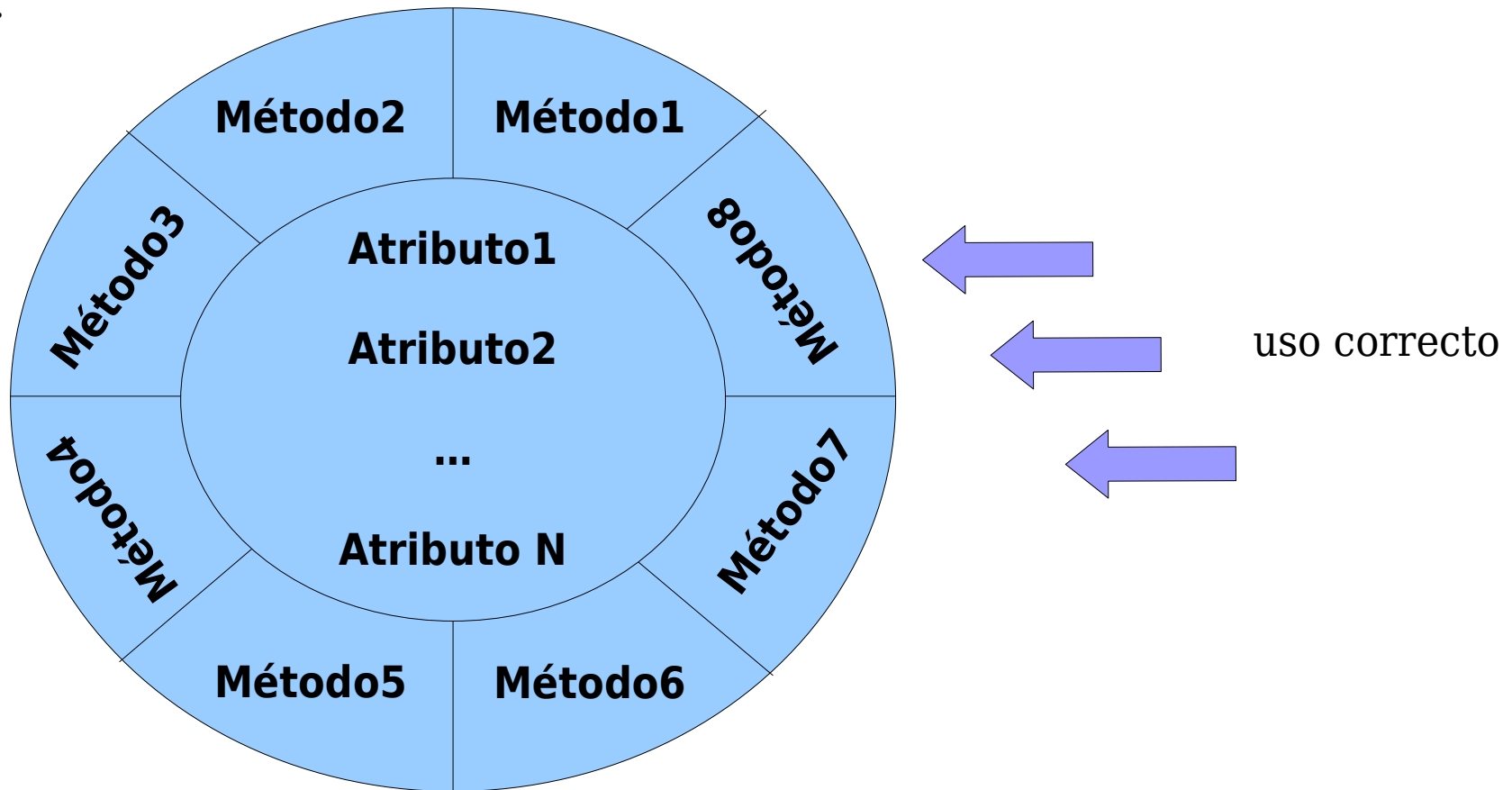
CPU

Métodos

- Encender
- Apagar
- Ejecutar programa

Encapsulamiento

- Es una propiedad que asegura que la información de un módulo este oculta al exterior.
- Ocultando el estado de un objeto, solamente podremos modificar sus atributos mediante sus métodos.



Constructores

- Métodos con el mismo nombre de la clase.
- Se utilizan para controlar el estado inicial en el que se crea un objeto.
- No poseen valor de retorno (ni siquiera void).
- No se pueden invocar directamente como otro método.