

# Análisis Orientado a Objetos (AOO)

## Definición de objeto

Según la Real Academia Española (rae.es) un objeto es, en su sexta acepción, una cosa, la cual es definida como:

**“Todo lo que tiene entidad, ya sea corporal o espiritual, natural o artificial, real o abstracta. “,**

En tanto nuestra disciplina un objeto puede definirse como:

“Una representación de casi cualquier información compuesta que un sistema (SOO) debe entender, entendiendo como *información compuesta* a algo que tiene muchas propiedades o atributos diferentes.”<sup>1</sup>

“Es una entidad tangible que exhibe algún comportamiento bien definido. Desde la perspectiva de la cognición humana un objeto es cualquiera de las siguientes cosas: una cosa tangible y / o visible, algo que puede comprenderse intelectualmente, algo hacia lo que se dirige un pensamiento o acción”.<sup>2</sup>

Basándonos en estas definiciones podemos asegurar que un objeto puede ser:

- ✦ una entidad externa, al sistema ( cualquier cosa que produzca o consuma información)
- ✦ una cosa (informe, listado)
- ✦ un suceso ( cualquier transacción)
- ✦ un evento ( activación de una alarma, encendido de luces, inicio de proceso)
- ✦ un rol (vendedor, cajero, supervisor, empleado)
- ✦ una unidad organizacional ( sucursal, departamento)
- ✦ un lugar (estante, góndola, cajón, anaquel)
- ✦ una estructura (archivo)

Cualquier elemento debe ser considerado de manera particular de acuerdo al dominio de la aplicación que se considere. Por ejemplo: un auto tendrá un conjunto de atributos si es considerado dentro de un sistemas de gestión de seguros y otros , tal vez distinto, si es visto como un articulo a ser vendido en una concesionaria.

Estas definiciones de objetos son incompletas ya que en términos del análisis es fundamental determinar cuáles son las responsabilidades que debe atender.

Una responsabilidad es: “Un contrato u obligación de un clasificador. Las responsabilidades están relacionadas con las obligaciones de un objeto en cuanto a su comportamiento”<sup>3</sup>

Una responsabilidad no es lo mismo que un método, pero los métodos se implementan para llevar a cabo responsabilidades. Las responsabilidades se implementan utilizando métodos.

Un objeto se caracteriza por tener un estado (conjunto de valores de sus atributos en un momento dado), comportamiento (conjunto de responsabilidades que atiende) y una identidad (un atributo que lo distingue unívocamente de cualquier otro objeto de su clase)

---

1 Roger Pressman, Ingeniera de Software 6º Edición.

2 Grady Booch, Análisis y diseño orientado a objetos con aplicaciones

3 Craig Larman, UML y patrones .Segunda edición

## **Identificación de objetos**

Para identificar objetos existen dos métodos:

- 1- Distinguir de un texto determinado los sustantivos y los verbos. Los sustantivos son posibles clases candidatas o atributos de las mismas. Los verbos son posibles responsabilidades. Las clases candidatas pasaran a ser definitivas cuando de acuerdo al contexto le asignemos los atributos.
- 2- El método número 1 depende de la redacción de cada individuo en particular, por eso el método número 2 es complementario y consiste en la Tabla de categorías de objetos de Larman. Todo objeto va a formar parte de una o más categorías.

### **Categorías de objetos según Larman**

Objetos tangibles o físicos	Animal, anteojos, libro, camisa
Especificaciones, diseños o descripciones de las cosas	Ingredientes de una receta, composición de un medicamento, detalle nutricional de un producto
Lugares	Aeropuerto, puerto, terminal ferroviaria, supermercado
Transacciones	Compra, reserva, inscripción, alquiler, préstamo
Líneas de transacción	Línea de compra, línea de reserva, línea de inscripción, línea de alquiler, línea de préstamo
Roles	Piloto, supervisor, gerente, colectivo, propietario
Contenedores de otra cosa	Frasco, taza, botella, depósito, gaveta
Cosas en un contenedor	Galletitas, canicas, agua, papeles, prendas de vestir
Otros sistemas externos	Sistemas de autorización, sistemas de registro, sistemas de control
Abstracciones	Políticas, descuento, disponibilidad
Organizaciones	Cadena hotelera, restaurante, sucursal, departamento
Hechos	Compra, pago, reserva, aterrizaje, vuelo
Reglas y políticas	Política de descuento, de reintegro, promociones
Catálogos	Catálogo de piezas, de películas, de libros, de productos varios
Registros legales	Documentos: recibo, cheque, contrato, remito

Como podrá apreciarse esta categorización no es taxativa, dado que un objeto cualquiera puede bien ser incluido en más de una de estas categorías.

## **Definición de paradigma**

Por paradigma entendemos a cualquier esquema formal con sus elementos y reglas de funcionamiento que nos permite organizar la realidad desde un punto de vista determinado de manera tal que simplifica la tarea de análisis de la situación en estudio.

En Análisis de Sistemas (AS) se trabaja usualmente con dos paradigmas: el Paradigma Estructurado (PE)<sup>4</sup> y el Paradigma Orientado a Objetos<sup>5</sup>.

### **Comparación entre el Paradigma Estructurado y el Paradigma Orientado a Objetos.**

Paradigma Estructurado	Paradigma Orientado a Objetos
.Nacido en la década del 70	.Nacido en la década del 80.
.La dimensión de la descomposición es por funciones.	.La dimensión de la descomposición es por objetos.
. Hay independencia entre datos y funciones. Lo cual es una dificultad en el momento de realizar actualizaciones en el sistema.	.Los datos y las funciones para manejar esos datos se unifican en una misma unidad que es el objeto. Lo cual es una ventaja en el momento de realizar actualizaciones en el sistema.
.La independencia entre datos y funciones provoca una dificultad en el momento de agregar funcionalidades nuevas al sistema (o sea extender el mismo).	.Al tener los datos y las funciones concentradas en el objeto va a facilitar en el momento de agregar nuevas funcionalidades al sistema.
.Curva de aprendizaje corta.	.Curva de aprendizaje larga.

Analizando lo anteriormente descripto se pueden observar las ventajas del Paradigma Orientado a objetos. Que son las siguientes:

- Menor dificultad en el momento de identificar objetos dado un sistema.
- Menor dificultad en el momento de realizar actualizaciones.
- Menor dificultad en el momento de extender el sistema.
- Posibilidad de reutilización.

## **Definición de clases**

Una clase es una abstracción de las cosas que forman parte del problema y de la solución.

Una clase no es un objeto individual, sino que representa un conjunto de objetos. Así, se puede pensar en una pared como una clase con ciertas propiedades (atributos) comunes, como altura, longitud, grosor, etc.

Una clase es una descripción de un conjunto de objetos que comparten los mismos atributos y responsabilidades.<sup>6</sup>

Un atributo es una propiedad de una clase identificada con un nombre. Un atributo representa alguna propiedad del elemento que se está modelando que es compartida por todos los objetos de esa clase. Cuando se crea un objeto/instancia a cada atributo se le asigna un valor. Por ejemplo:

Un objeto 1 de la clase puerta seria Altura: 2 metros, longitud: 1,20 metros, ancho: 10 centímetros.

Un objeto 2 de la clase puerta seria Altura: 2.20 metros, longitud: 1,30 metros, ancho: 12 centímetros.

Crear un objeto/instancia consiste en colocarle los valores a los distintos atributos de la clase.

---

<sup>4</sup> Ver Yourdon.

<sup>5</sup> Ver Larman / Pressman

<sup>6</sup> Grady Booch, James Rumbaugh, Ivar Jacobson. El lenguaje unificado de modelado.. Primera edición.

Un objeto siempre es instancia de una clase.

Existen instancias de cosas que no son objetos (Por ejemplo: los Templates.)

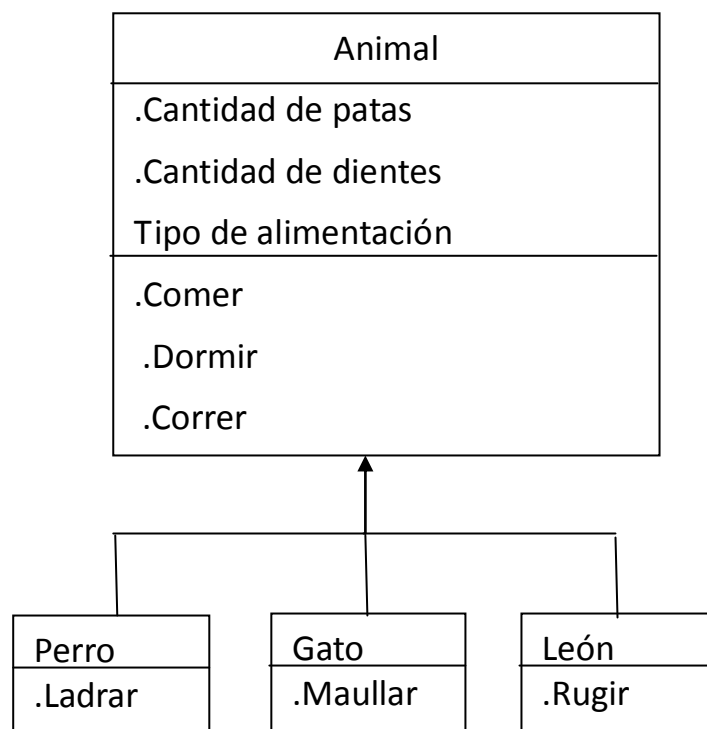
Hay que tener en cuenta que los atributos de una clase y el nombre de la misma dependen del contexto en el cual se esté trabajando. Por ejemplo: en un negocio de aberturas en el contexto de una venta; las puertas serian la clase producto cuyos atributos pueden ser descripción, precio de venta.

Si creamos un objeto 1 de la clase abertura se le podrían asignar los siguientes valores a los atributos. Descripción: ventana de aluminio para cocina de 1,50 por 1 metro, Precio: 320\$.

## **Relaciones entre clases**

Las relaciones entre clases son:

- 1- Herencia: Es una relación entre clases por la cual una o más clases (denominadas subclases) heredan los atributos y el comportamiento de una clase (denominada superclase) y le agregan atributos y comportamiento propio. Por ejemplo: tenemos un clase llamada Animal (superclase) y tres clases llamadas perro, gato, león (subclases). La clase Animal tendrá los siguientes atributos cantidad de patas, cantidad de dientes, tipo de alimentación, etc. y las siguientes responsabilidades comer, dormir, correr, la clase Perro, gato, león hereda todos los atributos y todas las responsabilidades de la clase Animal y le agrega en el caso de la Clase Perro la responsabilidad ladrar, en el caso de la clase Gato la responsabilidad maullar y en el caso de león la responsabilidad rugir.



La herencia es también llamada relación “Es un”. Por ejemplo: el León es un animal.

La herencia se puede obtener de dos formas: por especialización o por generalización.

Por especialización comenzaríamos con la superclase Animal y veríamos que animales podemos distinguir que tengan comportamiento en común y también comportamiento propio.

Por generalización comenzaríamos por las subclases y tendríamos que detectar atributos y comportamiento igual para crear la superclase Animal.

Ventajas de la herencia:

- . Reutilización de código ya testado.

- .Se pueden agregar subclases (sin que se altere el sistema) siempre y cuando tengan el mismo comportamiento que la superclase y agregue su comportamiento propio. Por ejemplo: se puede agregar la subclase serpiente cuyo comportamiento propio que agrega al de la superclase seria sisea.

- .Se pueden eliminar subclases sin que se altere el sistema. Por ejemplo: se puede eliminar la

Subclase Perro sin que el sistema se altere.

Desventaja de la herencia:

- .Genera alto acoplamiento entre clases (es el grado de conectividad entre dos clases).

Para que se entienda este concepto se detallara un ejemplo a continuación. Si quisiéramos reutilizar la subclase Perro en otro sistema en el cual no existe la herencia deberíamos agregarle todos los atributos y comportamiento de la superclase Animal. Esto demuestra que la clase Perro en la herencia no es independiente.

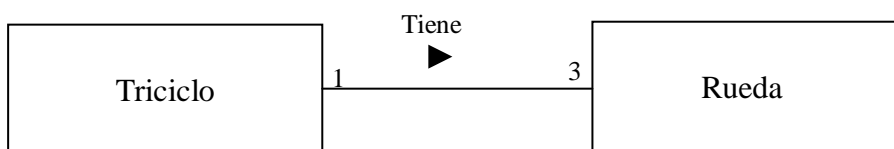
Asociación. La asociación se divide en:

### 2.1) Asociación simple:

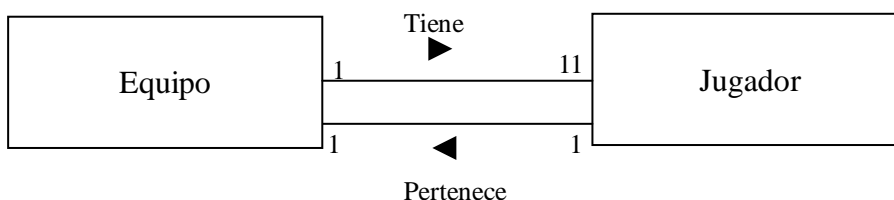
Es una relación semántica entre clases. A la misma se le pueden agregar adornos tales como sentido de lectura de la asociación, nombre de la asociación y cardinalidad.



Si el sentido de lectura de la asociación es de la Clase A hacia la Clase B, la cardinalidad indicara una instancia de la Clase A con cuantas instancias de la Clase B están relacionadas.

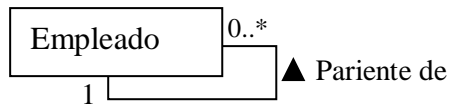


Si el sentido de lectura de la asociación es de la Clase Triciclo hacia la Clase Rueda, el nombre de la asociación seria Tiene y la cardinalidad indicara una instancia de la Clase Triciclo se relaciona con tres instancia de la Clase Rueda.



Si el sentido de lectura de la asociación es de la Clase Equipo hacia la Clase Jugador, el nombre de la asociación sería Tiene y la cardinalidad indicara que una instancia de la Clase Equipo se relaciona con una o muchas instancias de la clase Jugador. Si el sentido de lectura de la asociación es de la Clase Jugador hacia la Clase Equipo y estuviéramos refiriéndonos a un equipo de futbol profesional, una instancia de la Clase Jugador se relacionaría con una sola instancia de la Clase Equipo y el nombre de la asociación sería Pertenece.

Este ejemplo muestra que se puede dar el caso de tener que utilizar dos asociaciones con distinto sentido de lectura según que necesitemos modelizar.



Se puede tener relaciones entre instancias de la misma Clase empleado, si el nombre de la asociación es Pariente de, una instancia de la Clase Empleado puede estar relacionada con cero o muchas instancias.

## 2.2) Agregación (Relación Parte de o Tiene un).

Una clase puede tener atributos que sean de tipo numérico, string un objeto de otra clase.

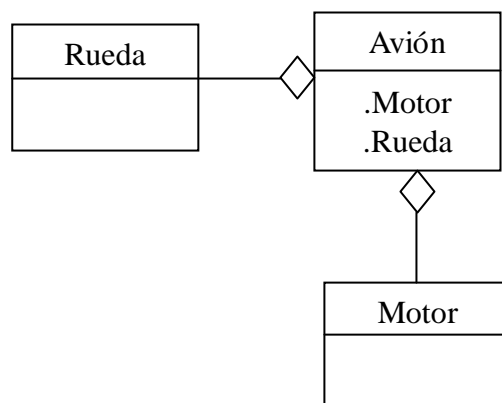
En la Agregación se tiene una clase que está formada por atributos que son objetos de otra clase.

Por ejemplo: Si tenemos la Clase Avión (el todo) puede tener como atributo un objeto de la Clase Rueda y otro de la Clase Motor (llamadas las partes).

Estas clases son independientes entre sí, eso significa que si no existiese una de las partes el Todo seguiría existiendo las Partes. Por ejemplo: si no existe la rueda el avión va a continuar siendo un avión aunque no pueda funcionar. Lo mismo si no existe el avión las ruedas van a existir.

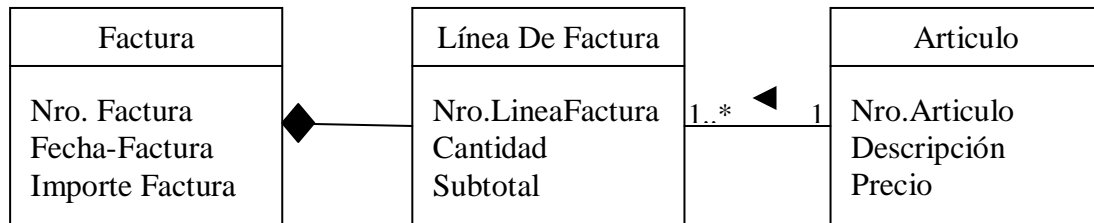
Se llama relación “Parte de” si se la observa de las partes hacia el todo, uno puede decir la rueda es parte del avión.

Se la pueda llamar también relación “Tiene un” si se la observa del todo hacia las partes. Por ejemplo: el avión tiene un motor.



### 2.3) Composición.

Es una relación parecida a la relación de agregación, su diferencia estriba en si deja de existir el “todo” las partes también dejan de existir. Por ejemplo: si tenemos la clase Factura (seria el todo) y la clase línea de factura (serian las partes). En este caso si la clase Factura dejara de existir, la clase línea de factura deja de existir.



## **Bibliografía:**

Roger Pressman  
Ingeniera de Software  
6° Edición.  
Mc Graw Hill  
2005

Grady Booch  
Análisis y diseño orientado a objetos con aplicaciones  
Primera edición  
Pearson Educación  
1996

Craig Larman  
UML y patrones  
Segunda edición  
Prentice Hall  
2003

Edward Yourdon.  
Análisis Estructurado Moderno  
Primera edición  
Prentice Hall  
1993

Grady Booch, James Rumbaugh, Ivar Jacobson.  
El lenguaje unificado de modelado.  
Primera edición.  
Addison Wesley  
2000