

CÁTEDRA



Ingeniería de Software II

2019

UNIDAD 2: Workflow de diseño



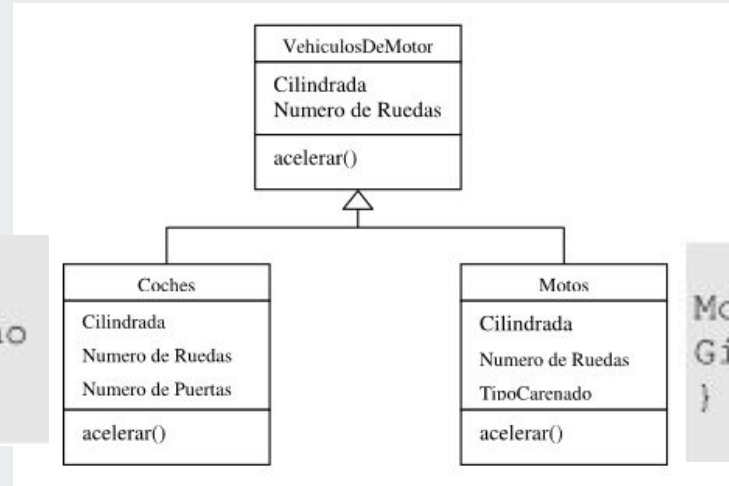
Es posible construir una nueva clase a partir de otra mediante el mecanismo de la herencia. Mediante el uso de la herencia las clases pueden extender el comportamiento de otra clase permitiendo con ello un gran aprovechamiento del código. Su principal virtud es el Polimorfismo.

Cuando una clase deriva de otra, hereda todas sus variables y métodos. Estas funciones y variables miembro pueden ser redefinidas (overridden) en la clase derivada, que puede también definir o añadir nuevas variables y métodos.

UNIDAD 2: Workflow de diseño

Recordamos:

POLIMORFISMO



```
Coches::acelerar(){
Pisar más el pedal derecho
}
```

```
Motos::acelerar(){
Girar más el puño derecho
}
```

Dos métodos implementen distintas acciones aún teniendo el mismo nombre dependiendo del objeto que ejecuta o de los parámetros que recibe.

UNIDAD 2: Workflow de diseño



• En las **clases de análisis** si se identifica una relación del tipo 'es un' se representa usando herencia. Representa la relación de negocio usando herencia.

En las **clases de diseño** se utiliza para reutilizar código. Facilita la implementación de una clase hijo usando herencia.

Relaciones entre clases

- Asociación
- Herencia
- Agregación

La asociación se utiliza en la etapa de análisis y al comienzo del diseño. Luego las asociaciones se refinan en las demás relaciones.

UNIDAD 2: Workflow de diseño

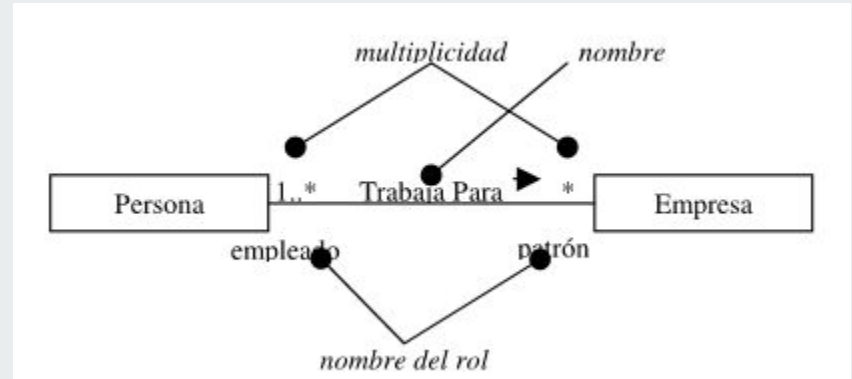
Asociación

Es una relación estructural que especifica que los objetos de un elemento se conectan a los objetos de otro.

Esto es suficiente durante el análisis de un problema, donde sólo es necesario identificar esas dependencias.

Multiplicidad en una asociación:

- Uno a uno
- Uno a muchos
- Muchos a muchos



UNIDAD 2: Workflow de diseño



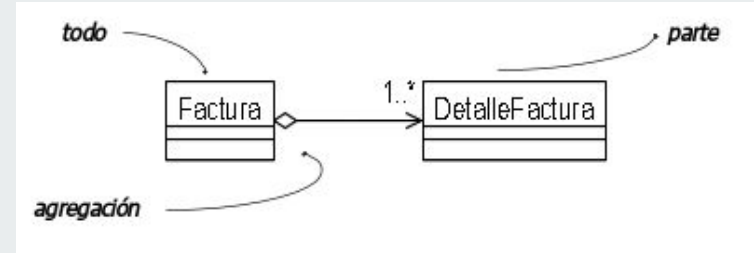
Los adornos que se aplican a las asociaciones para facilitar su comprensión:

- **Nombre:** **Se utiliza para describir la naturaleza de la relación.** Para que no exista ambigüedad en su significado se le puede dar una dirección al nombre por medio de una flecha que apunte en la dirección que se pretende que el nombre sea leído.
- **Rol:** Cuando **una clase participa en una asociación esta tiene un rol específico que juega en dicha asociación.** El rol es la cara que dicha clase presenta a la clase que se encuentra en el otro extremo. Las clases pueden jugar el mismo o diferentes roles en otras Asociaciones.
- **Multiplicidad:** **En muchas situaciones del modelado es conveniente señalar cuantos objetos se pueden conectar a través de una instancia de la asociación.** Este “cuantos” se denomina multiplicidad del rol en la asociación y se expresa como un rango de valores o un valor explícito. Cuando se indica multiplicidad en un extremo de una asociación se está indicando que, para cada objeto de la clase en el extremo opuesto debe haber tantos objetos en este extremo. Se puede indicar una multiplicidad de exactamente uno (1), cero o uno (0..1), muchos (0..*), uno o más (1..*) e incluso un número exacto (por ejemplo, 5).

UNIDAD 2: Workflow de diseño

Agregación

Es un tipo especial de asociación, que representa una relación completamente conceptual entre un “todo” y sus “partes”.



El conjunto puede existir algunas veces independientemente de las partes

Las partes pueden existir independientemente el conjunto

El conjunto está en cierto modo incompleto si faltan algunas de las partes

Es posible tener propiedad compartida de las partes por varios conjuntos

Las jerarquías de agregación son posibles

El todo siempre sabe de las partes, pero si la relación es de un solo sentido, del todo a la parte, las partes no saben del todo.

UNIDAD 2: Workflow de diseño

Composición

Es una variación de la agregación simple, con una fuerte relación de pertenencia y vidas coincidentes de la parte con el todo.

Las partes pertenecen exactamente a un componente de cada vez.

El compuesto tiene la única responsabilidad de disponer de todas sus partes, esto significa responsabilidad para su creación y destrucción.

El compuesto puede también liberar partes, en tanto que la responsabilidad para éstas se asuma por otro objeto

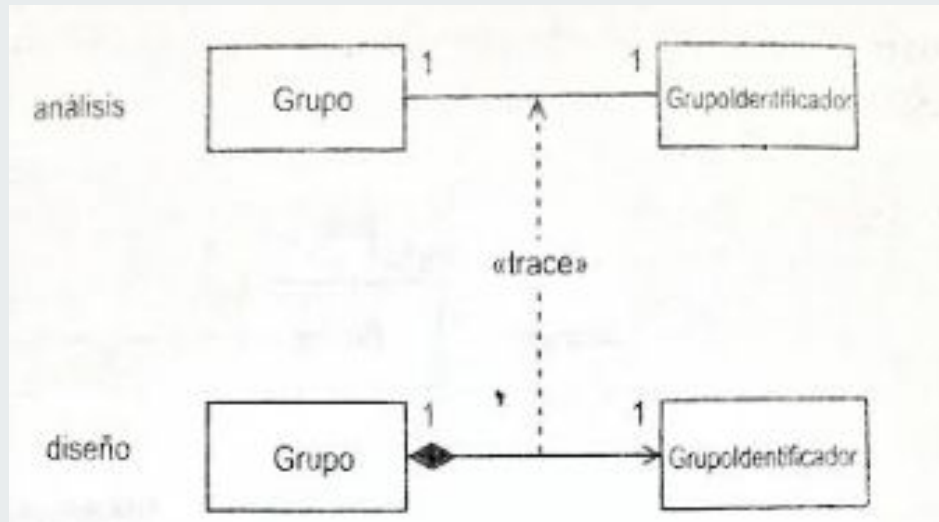
Si el compuesto se destruye, debe destruir todas las partes o pasar la responsabilidad de éstos a otro objeto.

Cada parte pertenece exactamente un compuesto por lo que solamente puede tener jerarquías de composición.



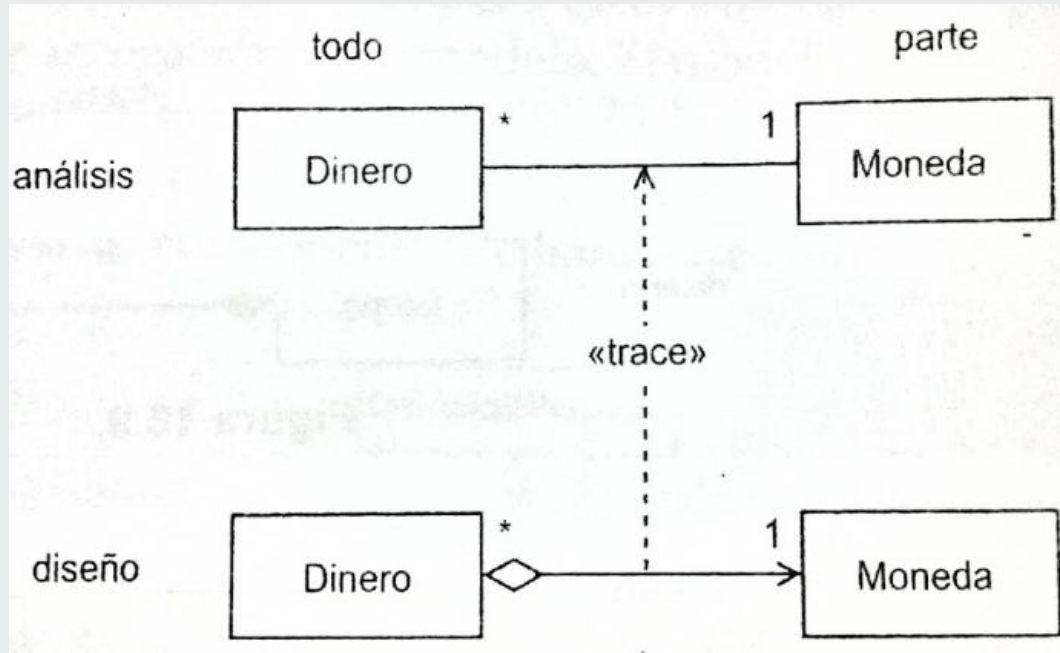
UNIDAD 2: Workflow de diseño

Asociaciones 0/1 a 1..... Usar composición



UNIDAD 3: Mejorar las relaciones de análisis

Asociaciones * a 1.... Usar agregación



UNIDAD 3: Mejorar las relaciones de análisis

Trabajando con el proyecto del equipo:

Pasamos de las clases de análisis a las clases de diseño:

- Verificar que se cumplen las características deseables (totalidad y suficiencia, sencillez, alta cohesión y bajo acoplamiento)
- Utilizar herencia para la reutilización de código.
- Añadir multiplicidad, dirección y nombre a las asociaciones.
- Convertir las asociaciones en agregación o composición cuando sea posible:
 - Decidir qué lado de la asociación es todo y cuál es la parte.
 - Examinar la multiplicidad del todo, si es 0..1, o exactamente 1, puede ser capaz de utilizar composición, o puede optar por añadir un atributo.
 - Asociaciones muchos a 1s: utilice agregación. Como existen muchos en el todo, no se utiliza composición
 - Asociación de uno a muchos. Se utilizan colecciones (será dado en clases siguientes)

UNIDAD 3: Mejorar las relaciones de análisis



Se abrirá un espacio para que puedan hacer entrega individual del trabajo en equipo agregando:

- 1 Diagrama de clases de diseño incorporando los cambios que considere necesario para satisfacer los requisitos planteados.
- 2 Si cambiara el proceso también actualizar el diagrama de actividad

UNIDAD 3: Mejorar las relaciones de análisis



Fuente:

UML 2 - “17 Diseñar clases”

- “18 Mejorar relaciones de análisis”