



Universidad Tecnológica Nacional
FACULTAD REGIONAL CORDOBA

PARADIGMAS DE PROGRAMACION

Unidad III

Paradigma Orientado a Objetos

Polimorfismo

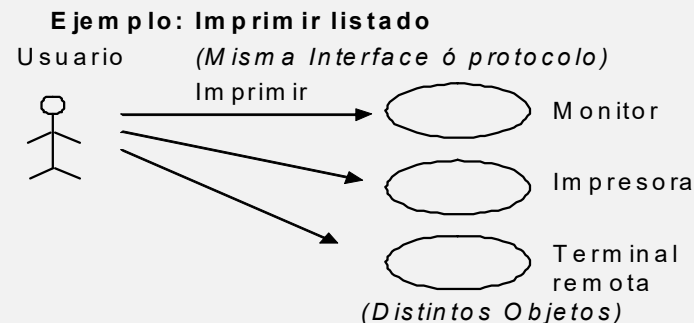


CONTENIDOS ABORDADOS

- Polimorfismo.
- Polimorfismo en Smalltalk.
- Caso de estudio.

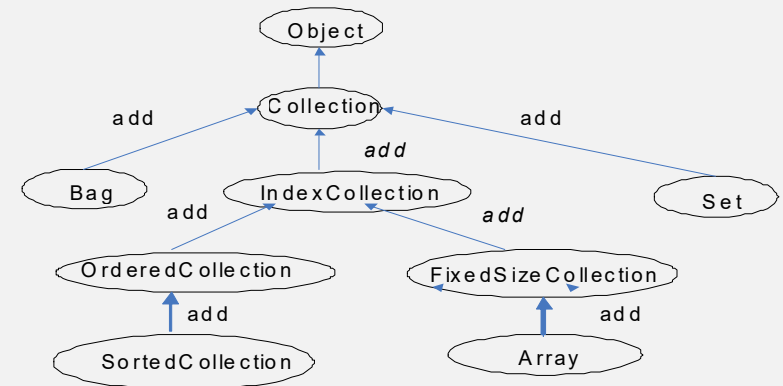
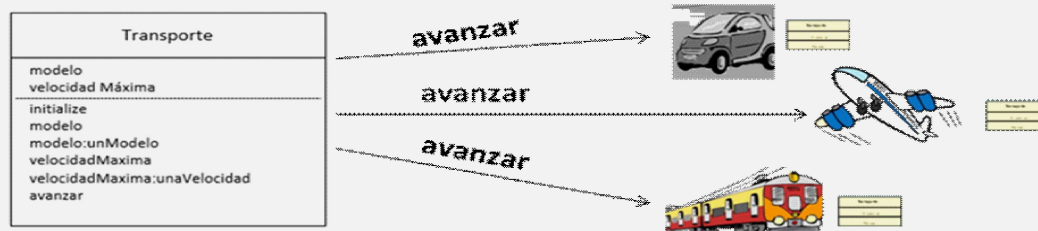
Polimorfismo

- Proviene del griego y significa que posee varias formas diferentes.
- Capacidad de que diferentes objetos reaccionen de distintas formas a un mismo mensaje.
- Posibilidad de acceder a un variado rango de comportamientos a través de la misma interfaz.
- Un mismo identificador puede tener distintas formas (distintos cuerpos de función, distintos comportamientos) dependiendo, en general, del contexto en el que esté inserto.



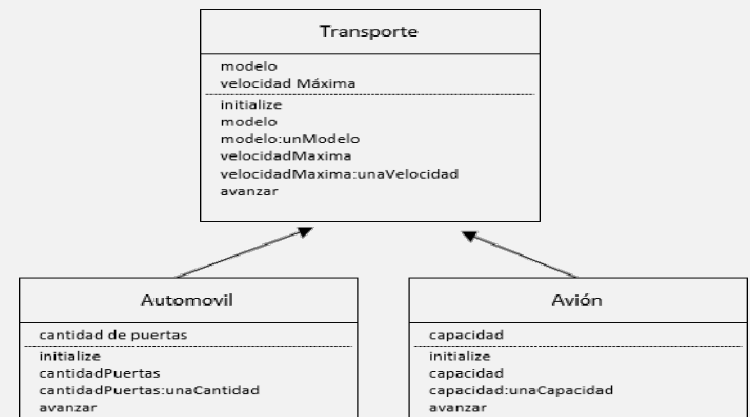
Polimorfismo de Subtipado

- Es la capacidad de referirse a objetos de clases distintas en una jerarquía utilizando el mismo mensaje (**protocolo**) para realizar la misma operación, pero de manera diferente.
- Facilita el almacenamiento y transporte de objetos de distintos tipos. Si varios objetos de distintos tipos están relacionados a través de un ancestro en común, con los mismos mensajes se los pueden manipular a través de éste.



Polimorfismo de Subtipado

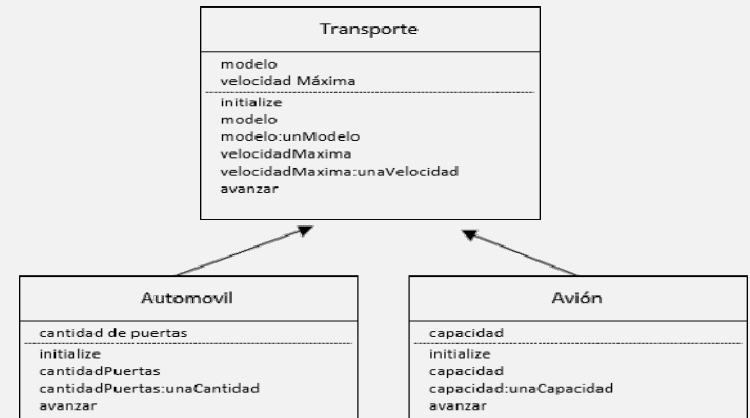
- Diseño e implementación de una jerarquía de clases.
- La definición del o los mensajes comunes (métodos polimórficos) residen en la clase base.
- Los mensajes comunes se irán redefiniendo en las clases derivadas, para aplicar el comportamiento correspondiente.
- En la jerarquía de clases puede presentarse la definición de clases abstractas o no. En el caso las clases abstractas suelen presentarse en el nivel superior de la jerarquía.



Polimorfismo de Subtipado

No se pueden evaluar los métodos abstractos,
por ejemplo avanzar de Transporte:

```
avanzar  
"Método abstracto"  
^ self subclassResponsibility.
```



Al intentar evaluar:

```
transporte := Transporte new.
```



No se produce error de
excepción, pero no se
debería instanciar.

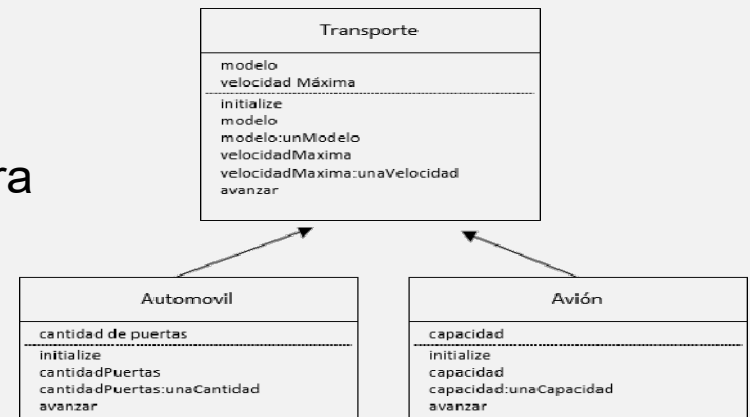
```
transporte avanzar.
```



Se produce error de excepción,
porque avanzar no está
implementado.

Polimorfismo de Subtipado

- Cuando se crean objetos polimórficos, será posible enviar mensajes polimórficos. Cualquiera de ellos podrá responderlo de una manera determinada.



- En el ejemplo, el mensaje **avanzar** puede ser respondido por cualquier objeto de la jerarquía:

```
automovil := Automovil new. avion := Avion new.
```

```
...
```

```
automovil avanzar.
```

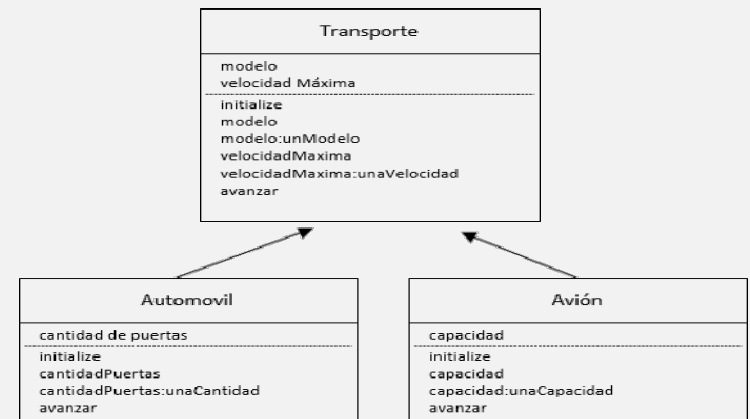


```
avion avanzar.
```



Polimorfismo de Subtipado

- En el caso de los métodos que no sean polimórficos se invocan haciendo referencia al objeto en el que esta definido.
- En el caso del mensaje **capacidad** solo puede ser respondido por un objeto **Avión**.



```
automovil := Automovil new. avion := Avion new.
```

...

```
automovil capacidad. ❌  
avion capacidad. ✅
```

Para evaluar si un objeto es de una clase determinada se utiliza:

```
(avion class = Avion) ó (avion className = 'Avion')  
ó (avion isKindOfClass: Avion)
```