

Clase 7 - DA2 Teórico

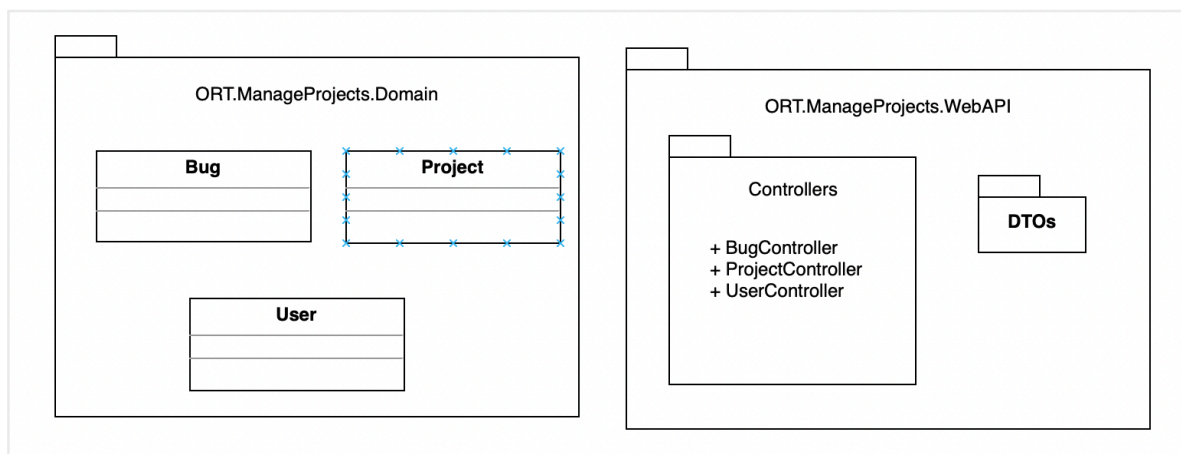
27/09/2021

HOY

- Dudas obligatorio
- Diagrama de objetos
- Diagrama de estructura compuesta
- Diagrama de componentes
- Diagrama de entrega

Diagramas de paquetes

Los representamos de esta manera:



Ahora utilizando la notación de Nesting:

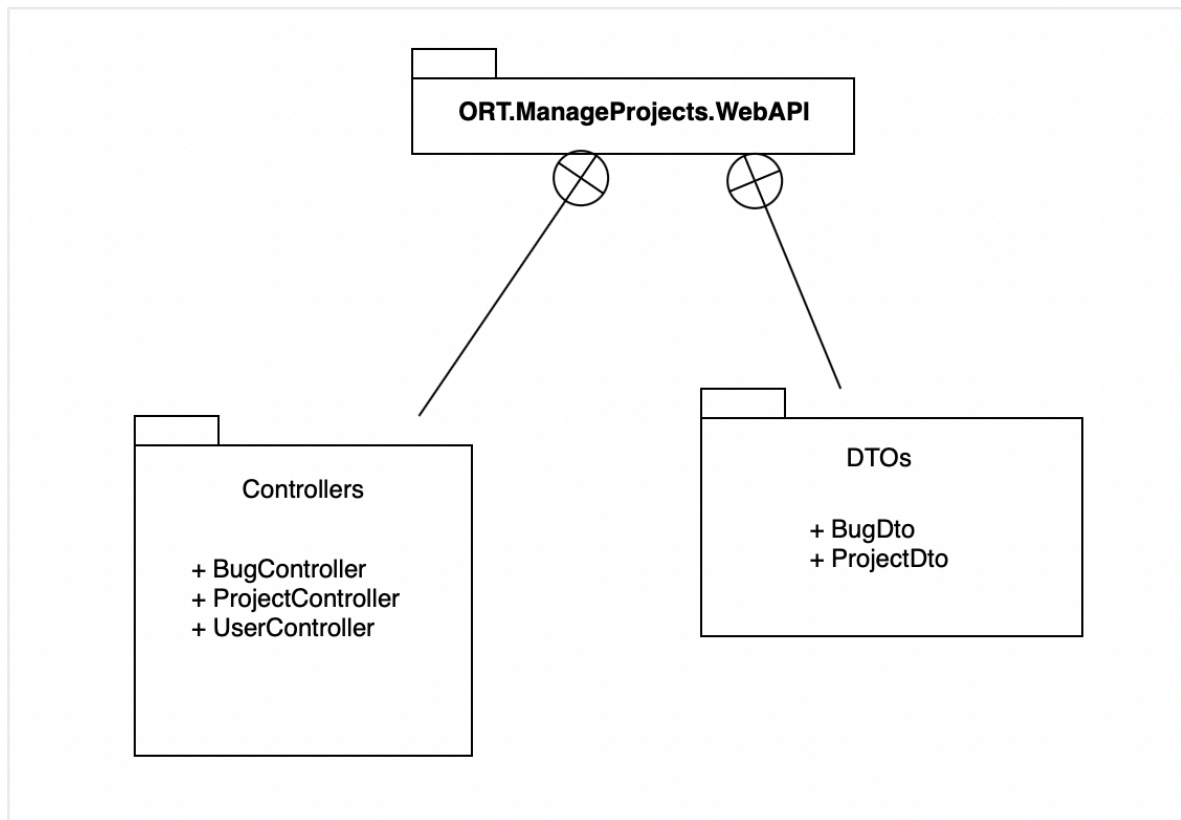


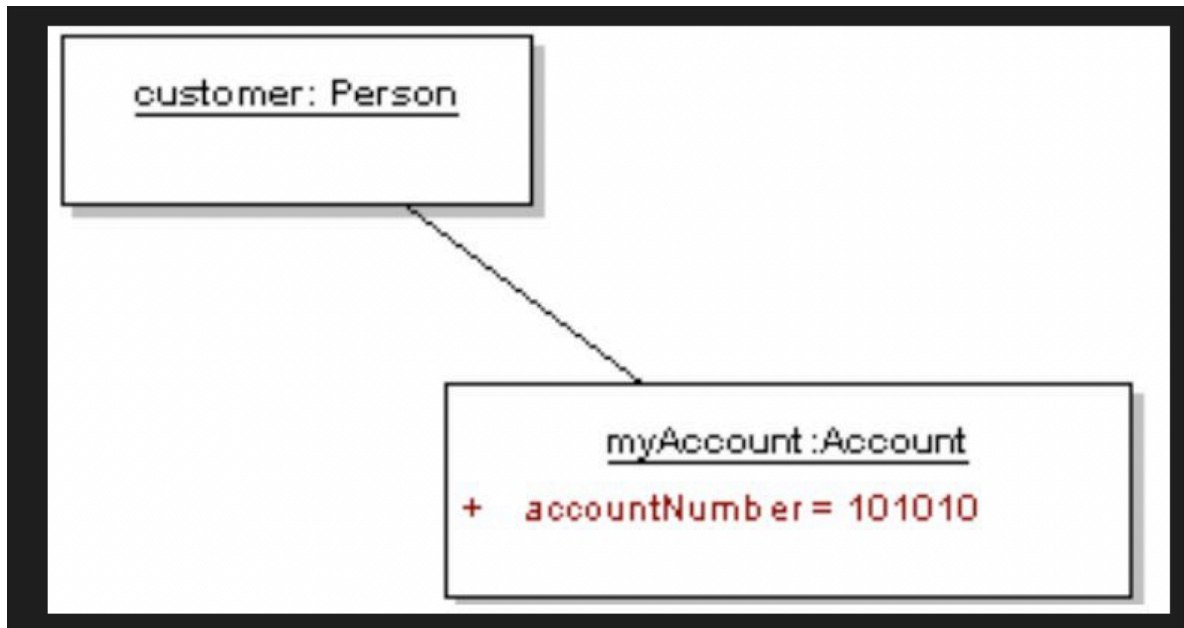
Diagrama de objetos:

Este diagrama es como si fuera una foto de un diagrama de clases en un punto del tiempo. Es decir, nos permite representar los objetos en un momento particular viendo los valores de sus atributos y relaciones entre sí.

¿Cuándo los usamos?

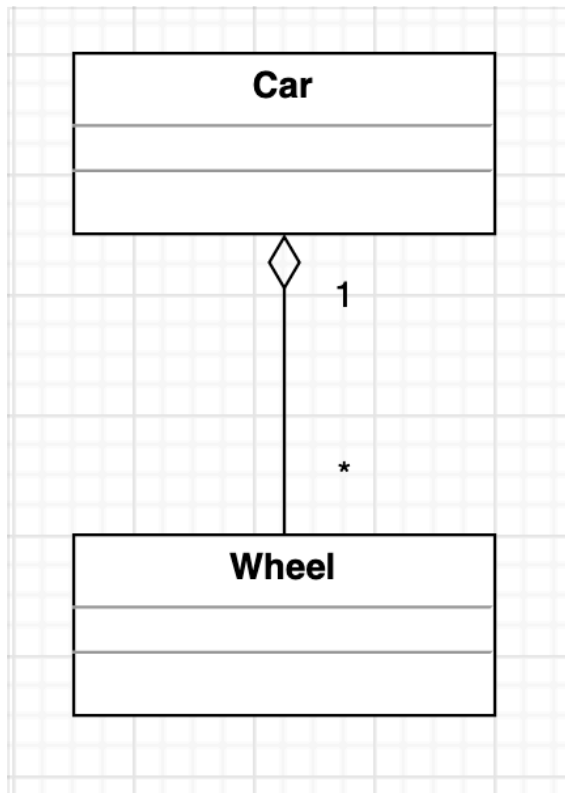
- Cuando queremos mostrar el contexto de ejecución de nuestro sistema (objetos en memoria, su estado y sus relaciones).
- Esta relacionado con el diagrama de comunicaciones e incluso podríamos contruir un diagrama de objetos desde un diagrama de comunicación.
- De un diagrama de clases puedo armar muchos de diagramas de objetos.

Notación:

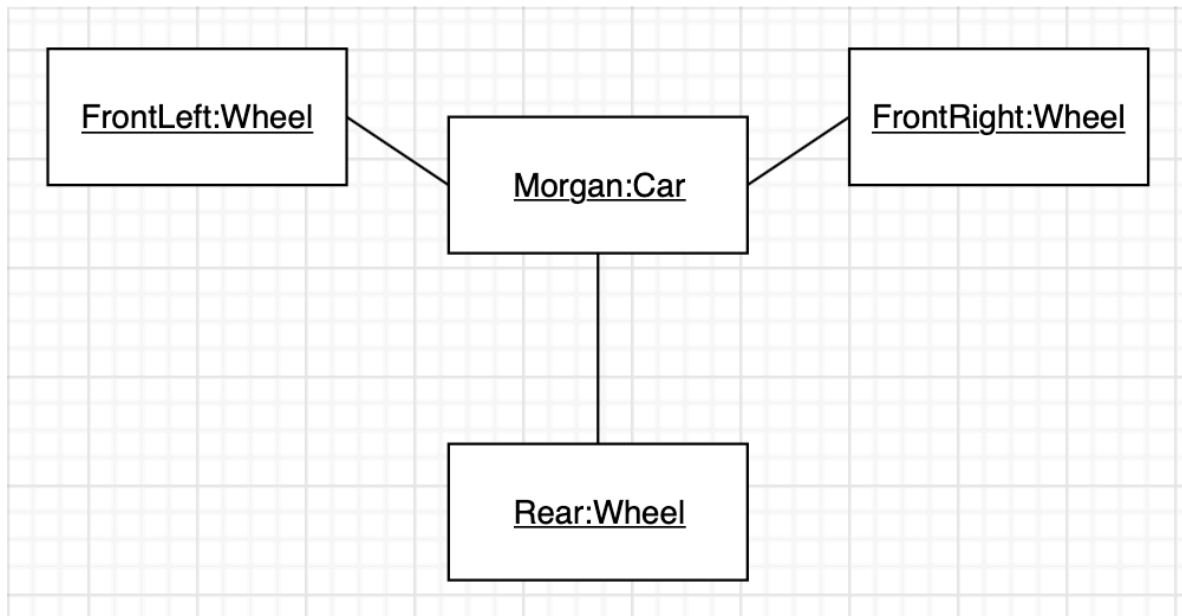


Ejemplo de auto Morgan: era un auto antiguo que tenía tres ruedas.

Primero haremos un diagrama de clases para resolver este problema:



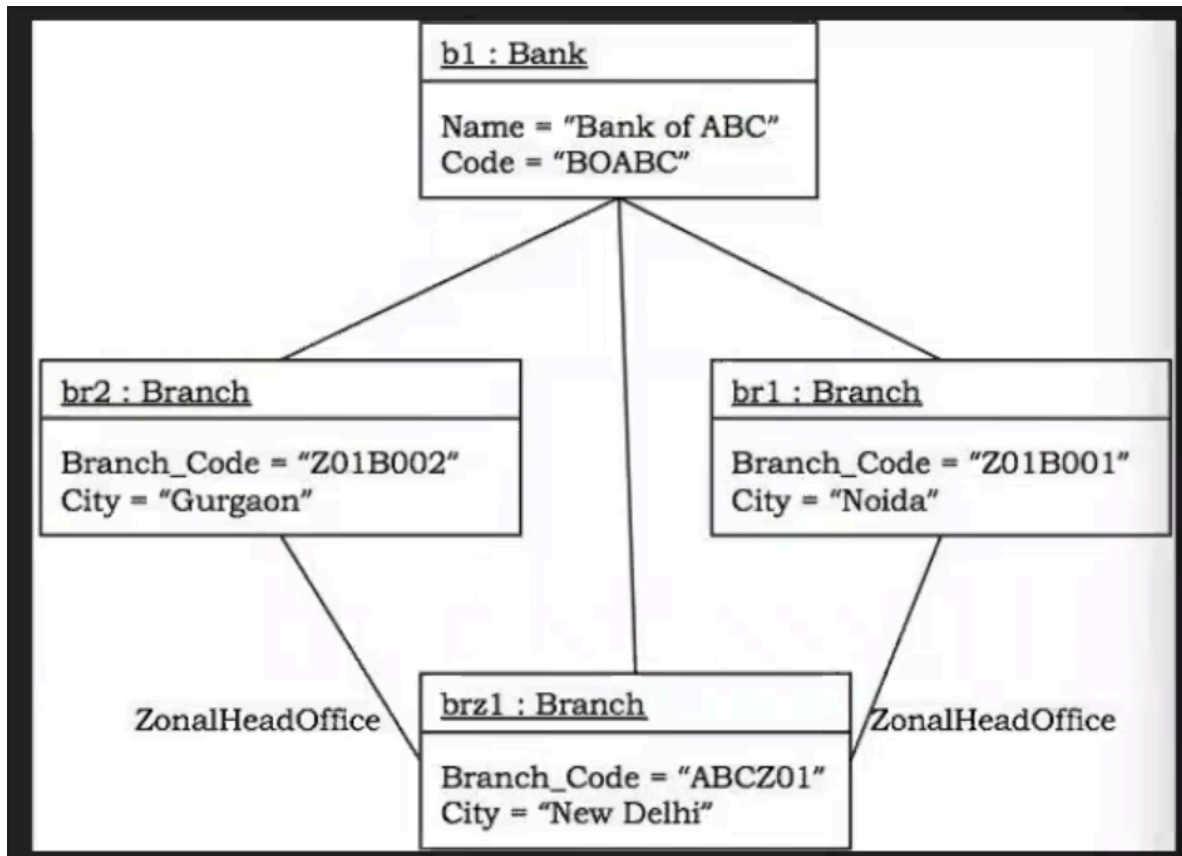
Ahora vamos a transformarlo en un diagrama de objetos:



Otro ejemplo:

Tenemos una instancia b1 de Banco. Asociado a ese banco tenemos diferentes sucursales. A su vez, para una determinada zona tienen una casa central y todas están asociadas a una "central".

Dado el diagrama de objetos:



Realizamos el diagrama de clases:

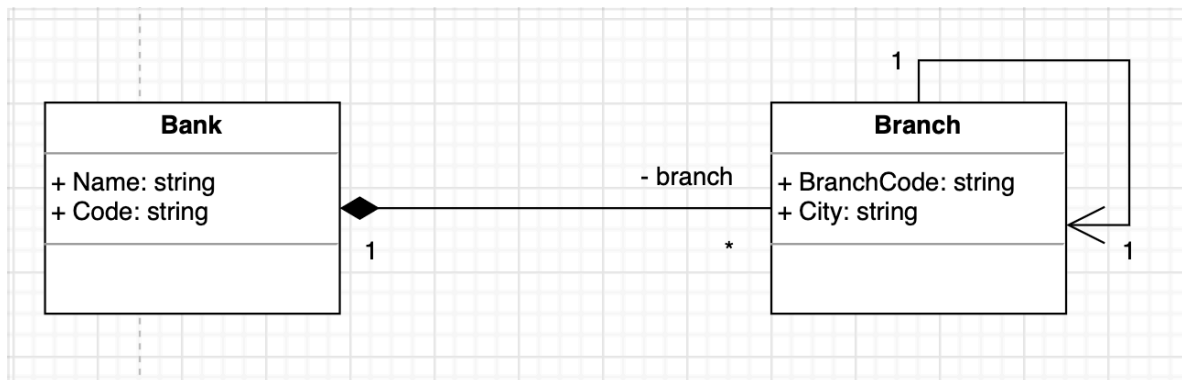
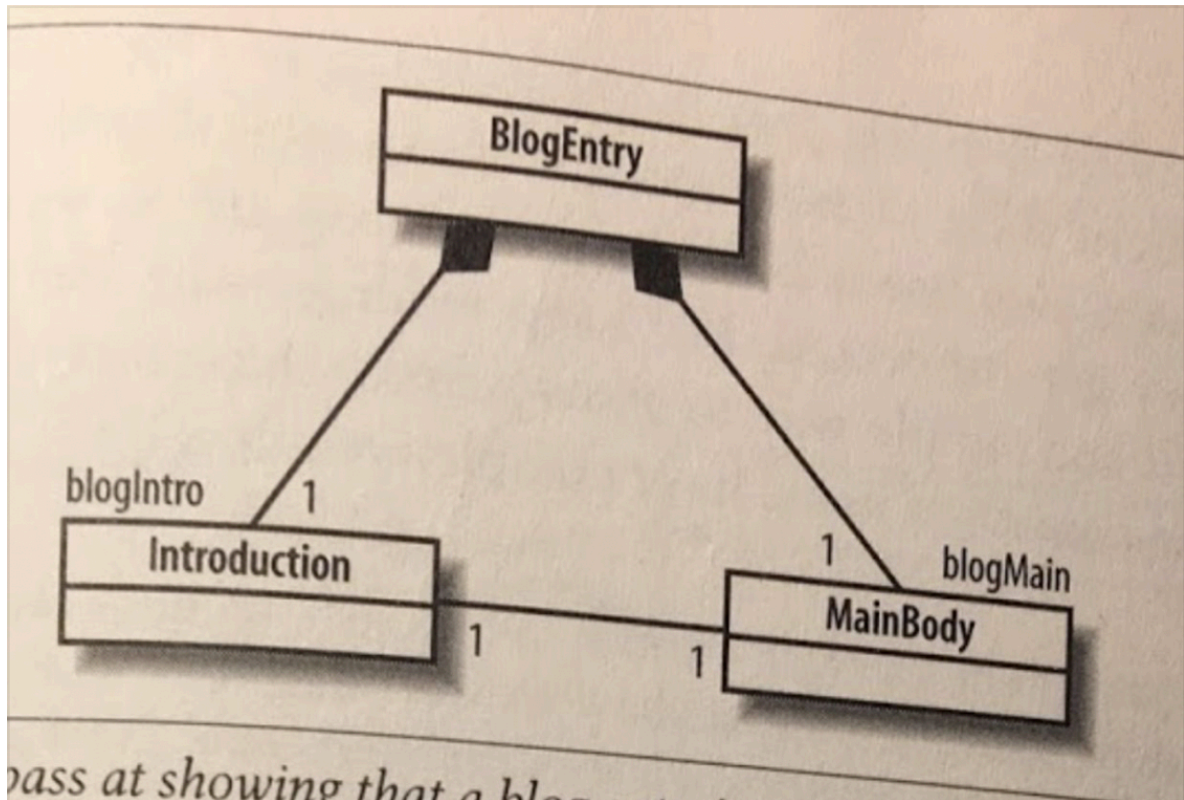


Diagrama de estructura compuesta

Estos diagramas nos muestran cómo ciertos objetos se componen internamente, bajo una relación (composición y/o agregación).

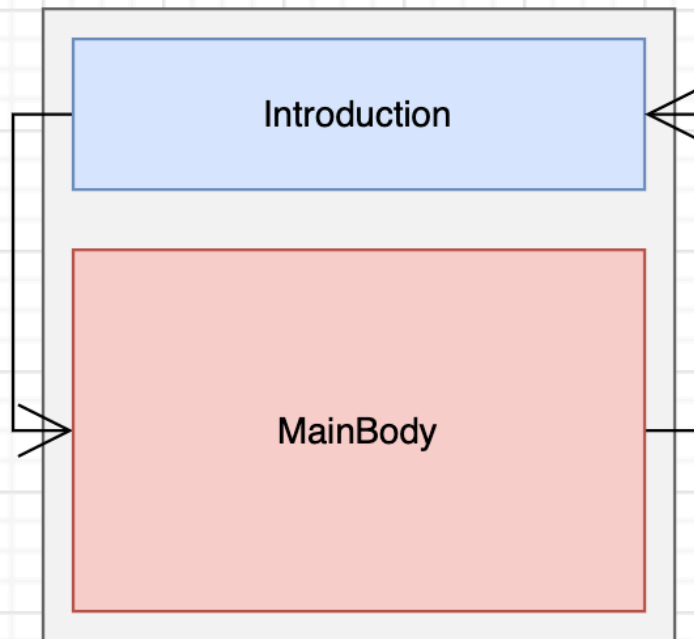
Particularmente los vemos porque hay ciertos escenarios en los que un diagrama de clases no es suficiente para poder modelarlos.

Por ejemplo en este diagrama de clases tenemos un escenario que no podemos ilustrar solamente con este diagrama.

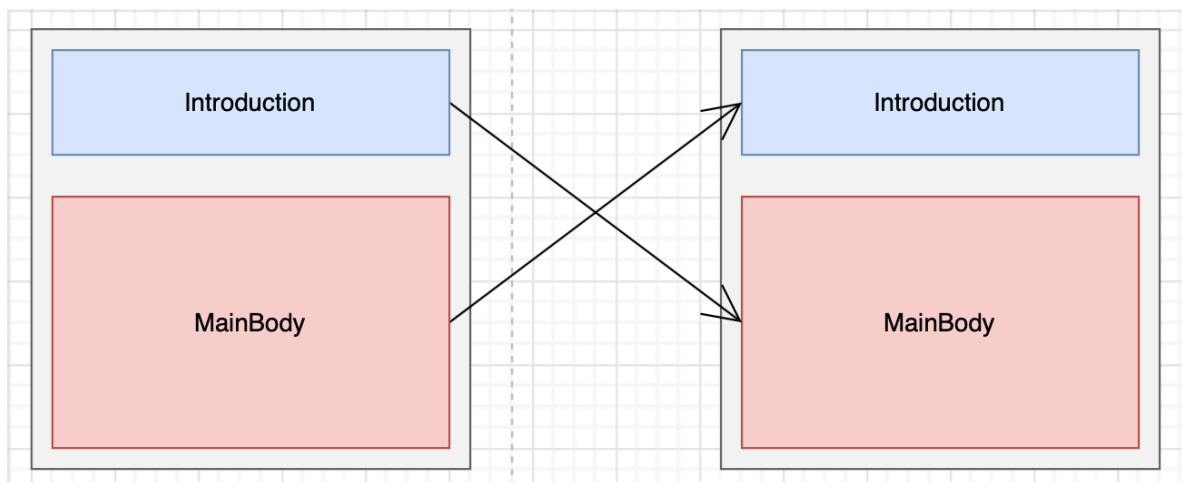


Ya que si lo expresamos de esta manera existen estos dos posibles casos de asociaciones:

BlogEntry (es una página donde posteamos artículos)



y



Esta última configuración es la que no queremos representar. Para lo cual vamos a tener que representarlo mediante un diagrama de estructura compuesta:

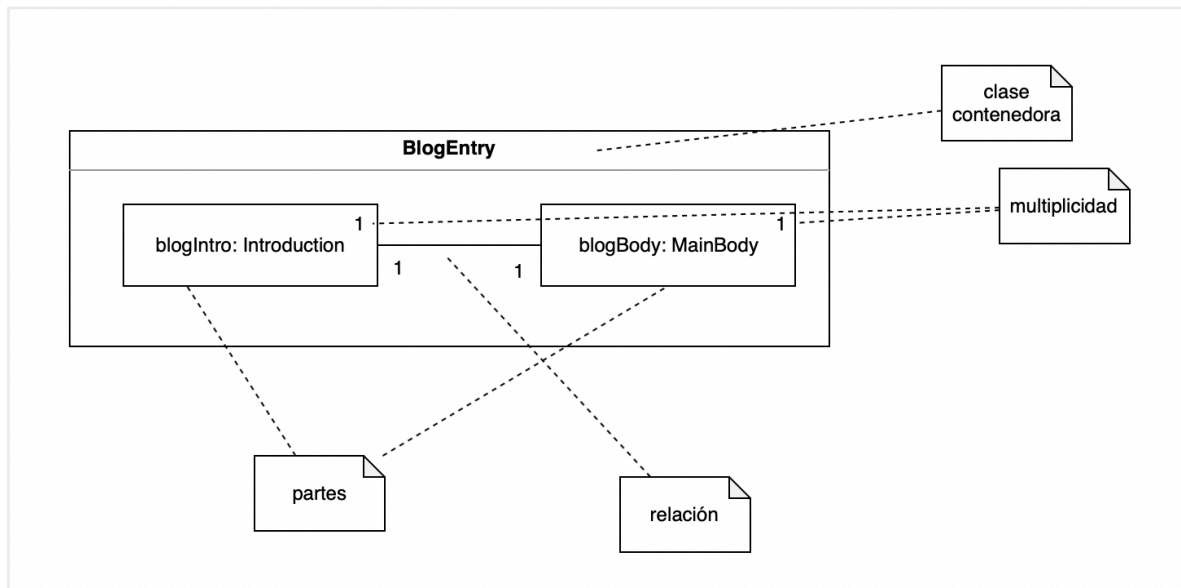


Diagrama de componentes

Este diagrama pertenece a la vista de componentes. Describe los elementos o artefactos que intervienen en el sistema, en el punto de vista de ejecución. Estos elementos van a ser los ensamblados (exe, dll).

Que son los componentes?

- Es una unidad modular de un sistema que tiene interfaces bien definidas.
- Al mismo tiempo, a consecuencia de definir una "interfaz concreta" (que pueden venir de métodos públicos de clases o métodos de interfaces definidas), los componentes son reemplazables dentro de su ambiente y son reusables.
- Juntando estos términos, podemos entender a un componente cómo: una parte física y reemplazable de un sistema que cumple o provee con un conjunto de interfaces.

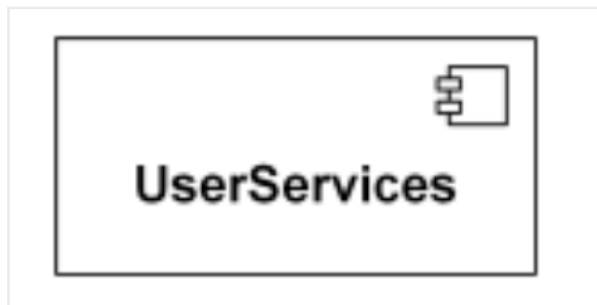
Tenemos dos vistas para un diagrama de componentes:

- La externa que se representa mediante símbolos de interfaz provista/requerida y/o relaciones de dependencias o de uso, como un componente interactúa con su entorno.
- La interna, representa como un componente (en su interior) implementa o cumple sus interfaces.

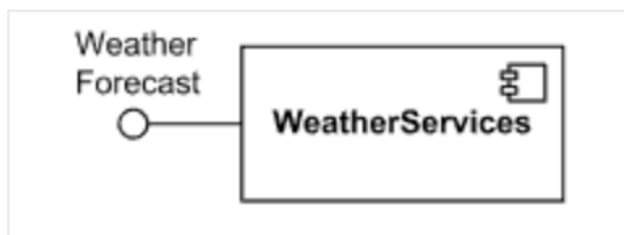
Particularmente, nosotros nos vamos a enfocar en la vista externa de un diagrama de componentes.

Notación:

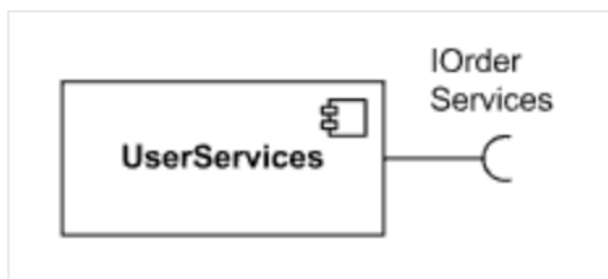
Los componentes los notamos así:



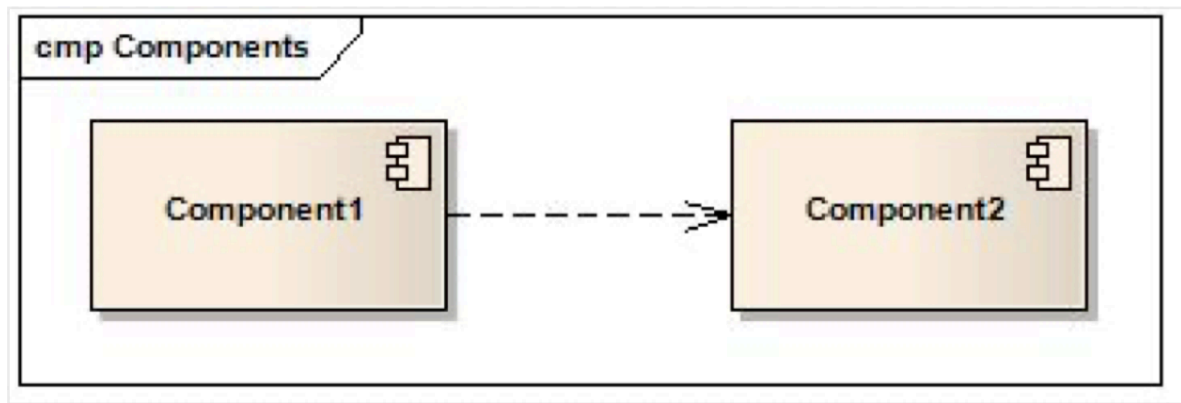
Del mismo modo un componente puede proveer una interfaz, que es implementada por el componente en si mismo (una clase que pertenece al componente e implementa dicha interfaz)



De forma análoga, un componente puede requerir interfaces, a través de tener clases que usen dicha interfaz



Por último, podemos ver como un componente usa a otro (sin acoplamiento a nivel de interfaz, sino que uno directamente usa a otro sin requerir/proveer interfaces)



Esto significa que hay clases o interfaces en el Component2 que son requeridas, realizadas o implementadas por clases del component1.

Ejemplo con obligatorio:

