Sugerencia: implementar solo 2 tipos de mensaje.

Mensajes posibles

- El cazador se encuentra en: (x, y) lugar
- He detectado monstruos de otra clase
- He detectado trampas de cazador en la cueva (z)
- La cueva (z) conecta con (g)

Por implementar:

- Métodos que **generan** los mensajes.
- Métodos que imprimen el mensaje **recibido** (cada monstruo al recibirlo).
- Creación del **mediador** que recibe y envía los mensajes.
- En UML: **agregar** roles: << colega >> etc.

- Conformación de **manadas dinámicas**
  - Se crean N cantidad de manadas.
  - Se crean aleatoriamente N cantidad de monstruos para cada manada.

- Hacer un ejemplo en el main:
  - Un monstruo que envía mensaje a otro de la **misma** manada
  - Ese mismo monstruo que envíe mensaje a otro de **otra** manada (no debería permitirse)
  - Ese mismo monstruo que envíe mensaje a otro monstruo **solitario** (no debería permitirse)
  - Que se “desagregue” de la manada
  - Y vuelva a enviar el mensaje a la manada que se acaba de desagregar (no debería permitirse)

Con esto se pasa la prueba de concepto

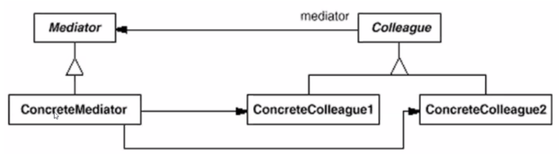
Mis notas (materia)

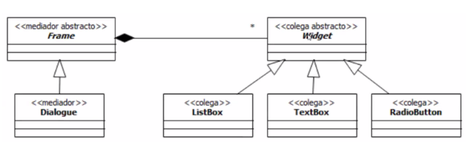
Patrón mediador.

Hay interdependencia en dos sentidos (entre objetos).
Responde a eventos (casi siempre concurrentemente)

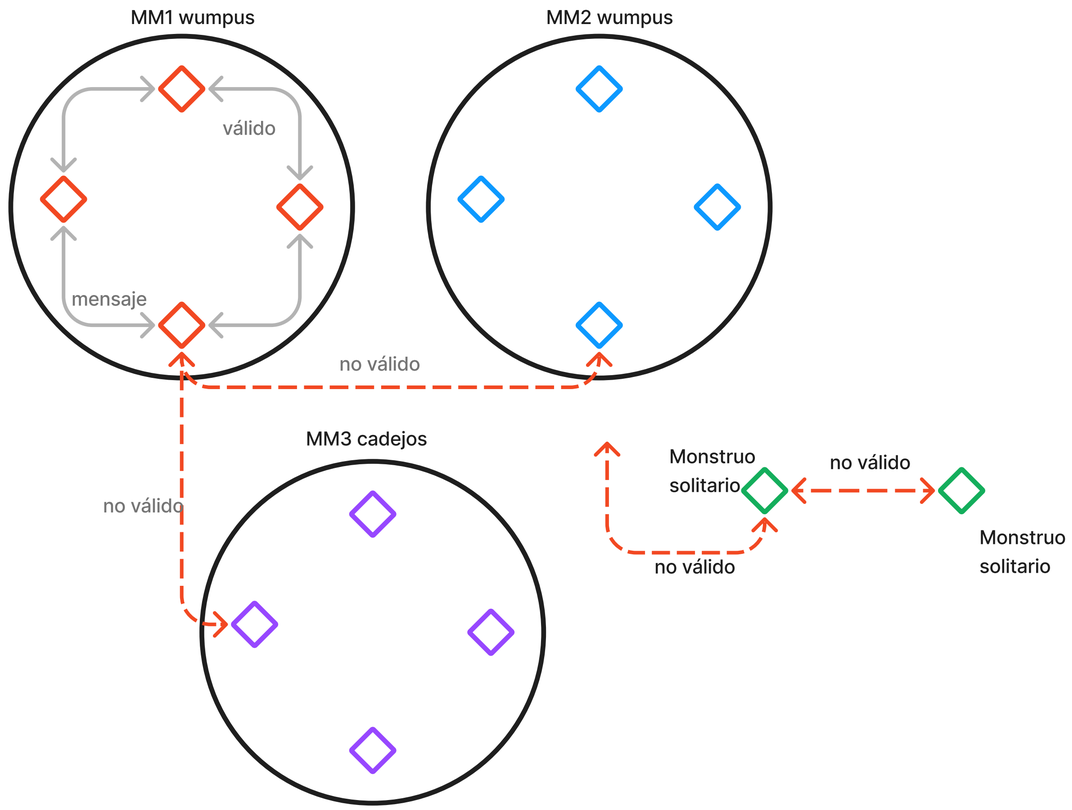
De 3 a 4 roles máximo.

Colegas: componentes que **dependen** entre sí.
Rol concreto de colega: representa un elemento donde interdependen con los demás.
Rol abstracto de colega: tiene al menos un método que asocia a un colega con el mediador. Es **opcional**.
Rol mediador abstracto / concreto: pueden ir combinados.
Facilita la comunicación entre los colegas. El colega le comunica al mediador y el mediador sabe a cuales otros colegas les interesa el evento.
El mediador recibe un evento y comunica a otros métodos de los otros colegas.





Funcionamiento



Links

https://www.oodesign.com/mediator-pattern.html

https://www.tutorialspoint.com/design_pattern/mediator_pattern.htm Ejemplo en código

https://sourcemaking.com/design_patterns/mediator/cpp/1 Otro ejemplo en código

https://www.dofactory.com/net/mediator-design-pattern Otro ejemplo en código (C#). Se ve como Gollo...