Diseño y Arquitectura de Software Patrón de Diseño Observer

- Observer
- •Es un patrón de diseño de software que define una dependencia uno a muchos, de manera que cuando uno de los objetos cambia su estado, las demás dependencias deberán ser notificadas del cambio de estado, este patrón también es conocido como como patrón de publicacióninscripción o modelo-patrón.
- Este patrón tiene relación con los patrones:
- Singleton.
- •Mediator.
- Problema
- •Un diseño grande no se adapta bien a medida que se imponen nuevos requisitos de gráficos o monitoreo.

- Solución
- Sujeto:
- •El sujeto proporciona una interfaz para agregar y eliminar observadores, el sujeto conoce a todos sus observadores.
- Observador:
- •Define el método que usa el sujeto para notificar cambios en estado (update/notify).
- Sujeto concreto:
- •Mantiene el estado de interés para los observadores concretos y los notifica cuando cambia su estado, no tienen porque ser elementos de la misma jerarquía.
- Observador concreto:
- •Mantiene una referencia al sujeto concreto e implementa la interfaz de actualización, guardan la referencia del objeto que observan así en caso de ser notificados de algún cambio, estos pueden preguntar sobre este cambio.

Consecuencias

- •Con este patrón se desconocen las consecuencias de una actualización, lo cual, dependiendo del problema, puede afectar en menor o mayor medida como por ejemplo al rendimiento.
- •Por otra parte abstrae el acoplamiento entre el sujeto y observador lo cual es un beneficio ya que se consigue una mayor independencia y además el sujeto no necesita especificar los observadores afectados por un cambio.

