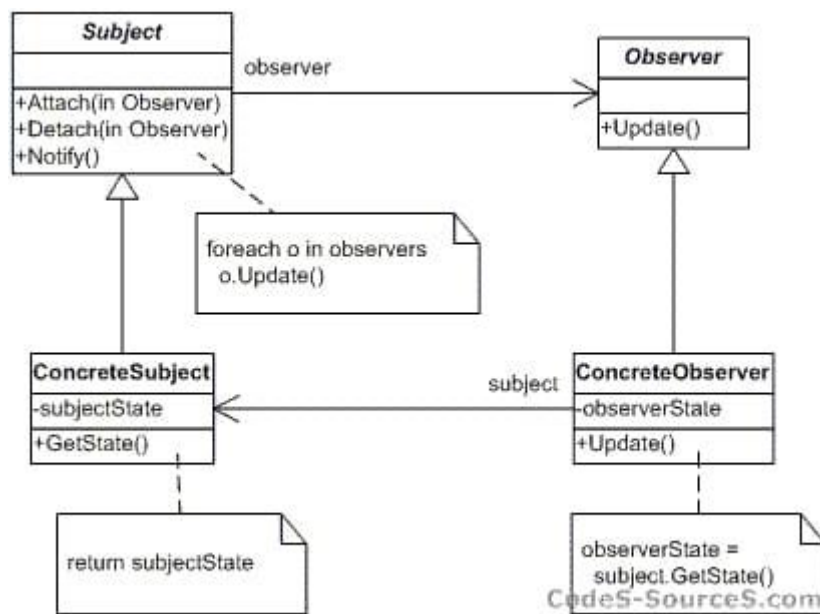


Observer

Le design pattern observer est utilisé lorsqu'il existe une relation un-à-plusieurs entre des objets, par exemple si un objet est modifié, ses objets dépendants doivent être notifiés automatiquement. Le modèle observer relève de la catégorie des modèles de comportement.



Le modèle observer utilise trois classes d'acteurs. Sujet, Observateur et Client. Le sujet est un objet ayant des méthodes pour attacher et détacher des observateurs à un objet client. Nous avons créé une classe abstraite Observer et une classe concrète Subject qui étend la classe Observer

- Le client configure le nombre et le type d'Observateurs.
- Différencier les fonctionnalités primaires et optionnelles.
- Les observateurs s'enregistrent auprès du Sujet.
- Modeler les fonctions indépendantes avec une abstraction « subject ».
- Modeler les fonctions dépendantes avec une abstraction « observer ».
- Le sujet n'est couplé qu'à la classe de base Observer.
- Le Sujet diffuse des événements à tous les Observateurs enregistrés.
- Le Sujet peut « pousser » des informations vers les Observateurs, ou les Observateurs peuvent « tirer » les informations dont ils ont besoin auprès du Sujet.

- Le client configure le nombre et le type d'observers.
- Le sujet diffuse les événements à tous les observers enregistrés.
- Le sujet peut « pousser » les informations vers les observers ou les observers peuvent « tirer » les informations dont ils ont besoin du sujet