

LE PATTERN STRATEGIE

Intention

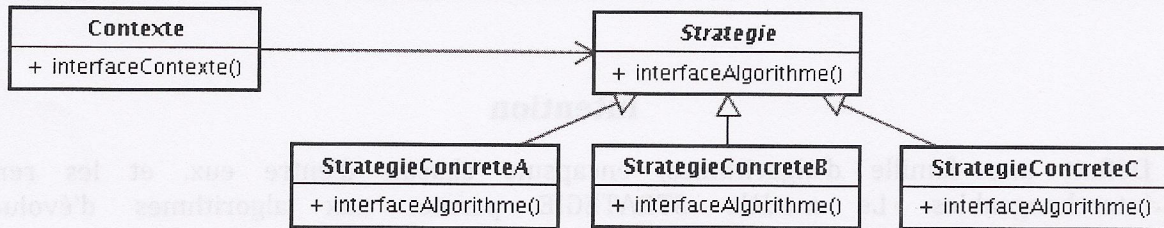
Définit une famille d'algorithmes, encapsule chacun d'entre eux, et les rend interchangeables. Le modèle STRATEGIE permet aux algorithmes d'évoluer indépendamment des clients qui les utilisent.

Indications d'utilisation

On utilisera le modèle Stratégie dans les cas où :

- Plusieurs classes apparentées ne diffèrent que par leur comportement. Les Stratégies donnent le moyen d'appareiller une classe avec un comportement parmi plusieurs autres.
- On a besoin de diverses variantes d'un algorithme. Par exemple, on peut définir des algorithmes représentant différents compromis encombrement mémoire / temps d'exécution.
- Un algorithme utilise des données que les clients n'ont pas à connaître. Utiliser la Stratégie dispense d'avoir à révéler des structures complexes de données spécifiques des algorithmes.
- Une classe définit de nombreux comportements, qui figurent dans ses opérations sous la forme de déclarations conditionnelles multiples. Plutôt que laisser subsister ces expressions conditionnelles, on déplacera les sections correspondantes aux différentes branches, dans des classes Strategie qui leur sont propres.

Structure



Constituants

Stratégie

- Déclare une interface commune à tous les algorithmes représentés. Contexte utilise cet interface pour appeler l'algorithme défini par une StrategieConcrete.

StrategieConcrete

- Implémente l'algorithme, en utilisant l'interface Stratégie

Contexte

- Est composé à l'aide d'un objet StrategieConcrete.
- Gère une référence à un objet Stratégie.
- Peut définir une interface qui permette à Stratégie d'accéder à ses données

Collaborations

Strategie et Contexte interagissent dans l'implémentation de l'algorithme choisi. Un Contexte peut fournir à la stratégie toutes les données que requiert cet algorithme, lorsqu'il est appelé. De même, le contexte peut se communiquer lui-même, en tant qu'argument, aux opérations de Strategie. C'est ce qui permet à la stratégie de se mettre en rappel sur le contexte en cas de besoin.

Un contexte transmet les requêtes de ses clients à sa stratégie. En général, les clients créent un objet StrategieConcrete, et le passent au contexte ; par la suite les clients interagissent exclusivement avec le contexte.