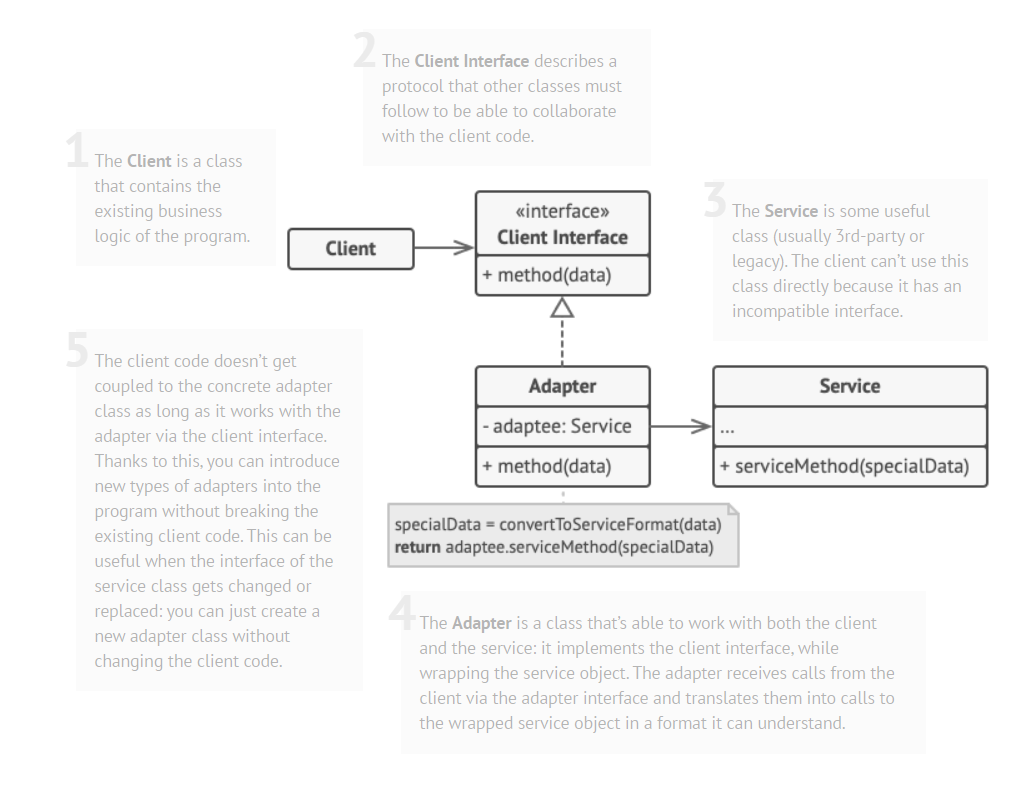
Șabloane

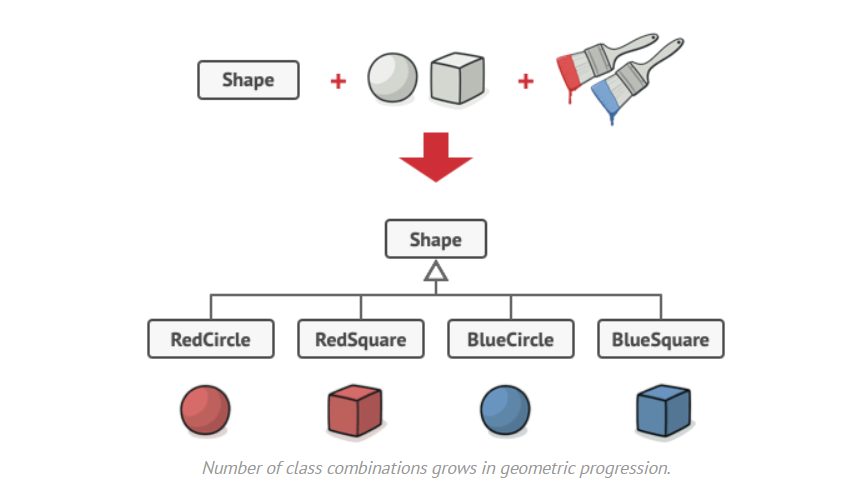
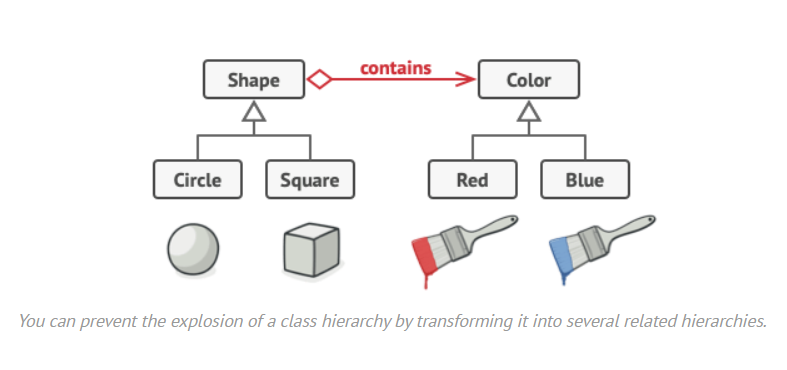
Adapter:

Sablonul Adapter este folosit pentru a permite comunicarea dintre 2 clase incompatibile.



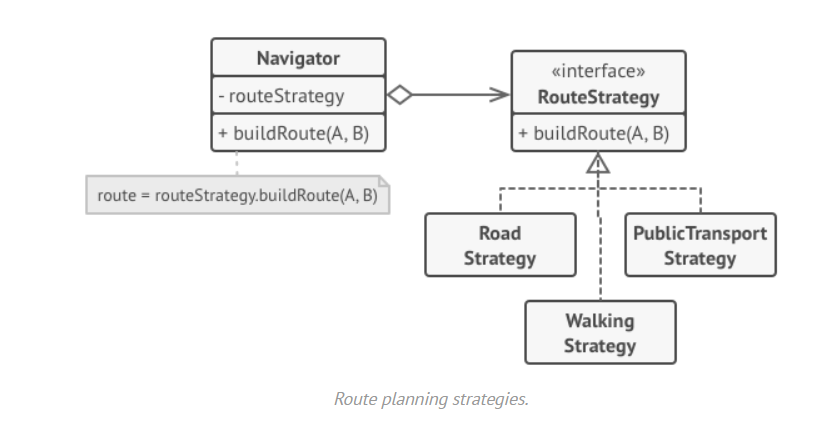
Bridge:

Sablonul Bridge separa abstractizarea fata de implementare, astfel incat cele 2 clase pot fi utilizate independent.

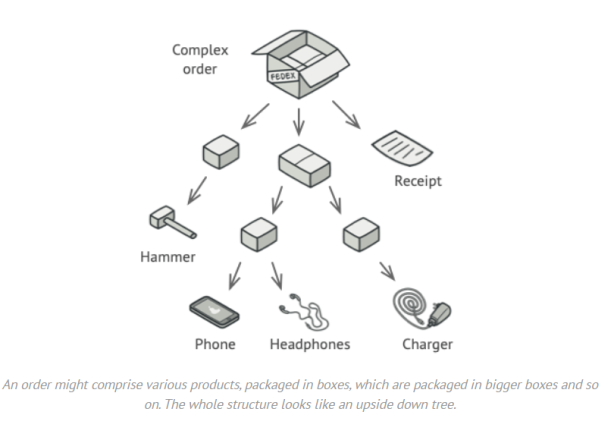
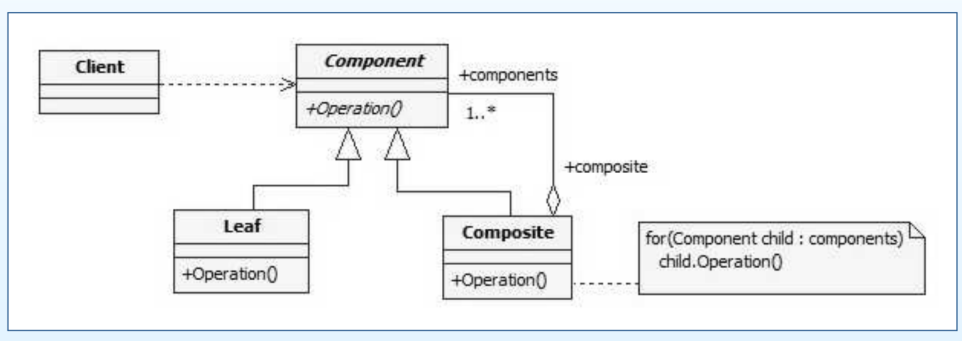
 

Strategy:

Sablonul Strategy se folosește pentru a putea usura situatia cand exista mai multi algoritmi si de asemenea pentru introducerea lor. Permite algoritmului sa varieze independent de clientii care îl utilizeaza.

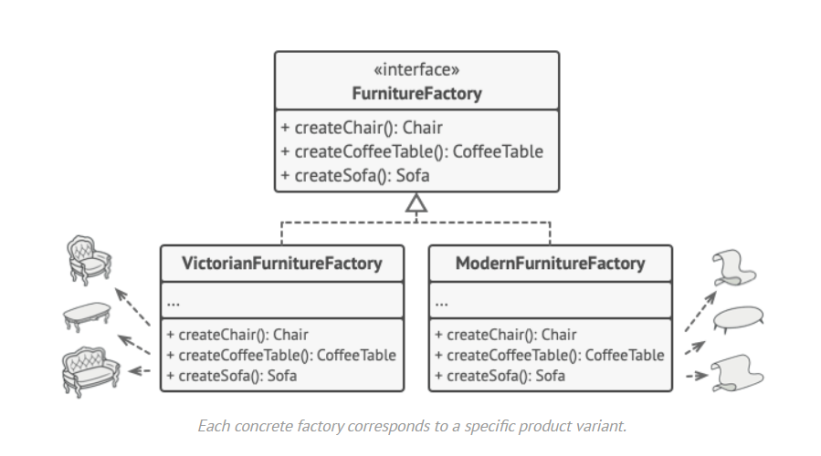
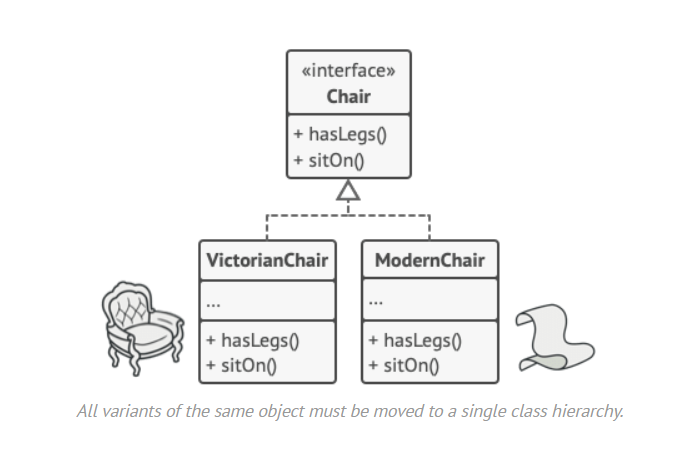


Composite:

Sablonul composite permite compunerea obiectelor ca intr-o structura de arbore, iar apoi sa poti lucra cu orice subarbore individual.

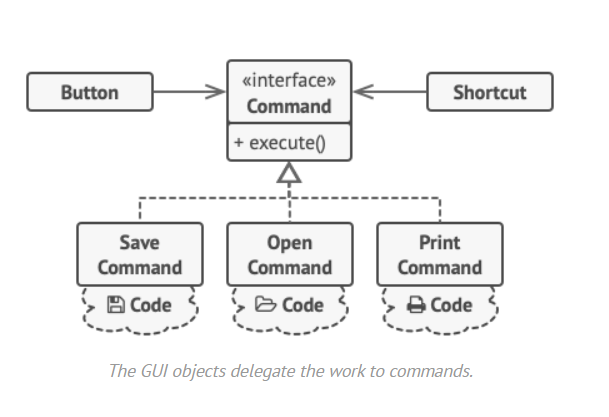
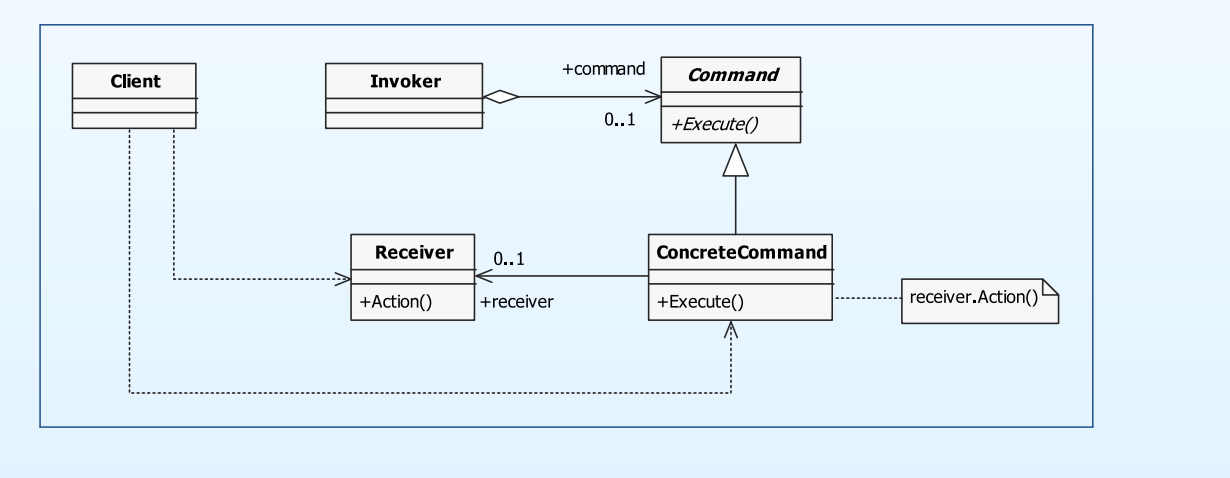
Abstract Factory:

Sablonul Abstract Factory creeaza o interfata pentru a genera familiilor de obiecte înrudite sau dependente, fara specificarea claselor lor.

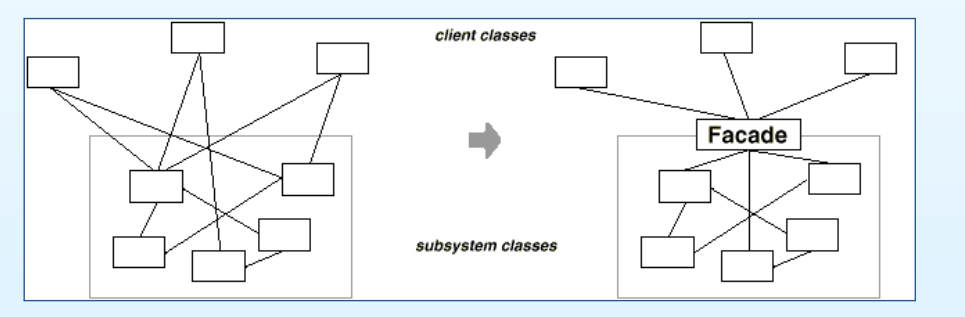


Command:

Sablonul Command se foloseste pentru a transforma o cerere intr-un obiect, astfel nu trebuie sa se creeze diferite obiecte pentru fiecare cerere. Se foloseste si pentru formarea unei cozi sau a unui registru de cereri si asigurarea suportului pentru operatiile ce pot fi anulate.



Facade:

Sablonul Facade Ofera o interfata unificata, de nivel înalt, pentru un grup de interfete ale unui subsistem, care faciliteaza utilizarea acestuia.

Proxy:

Sablonul Proxy asigura, pentru un obiect un înlocuitor, în scopul controlarii accesului la acesta.

