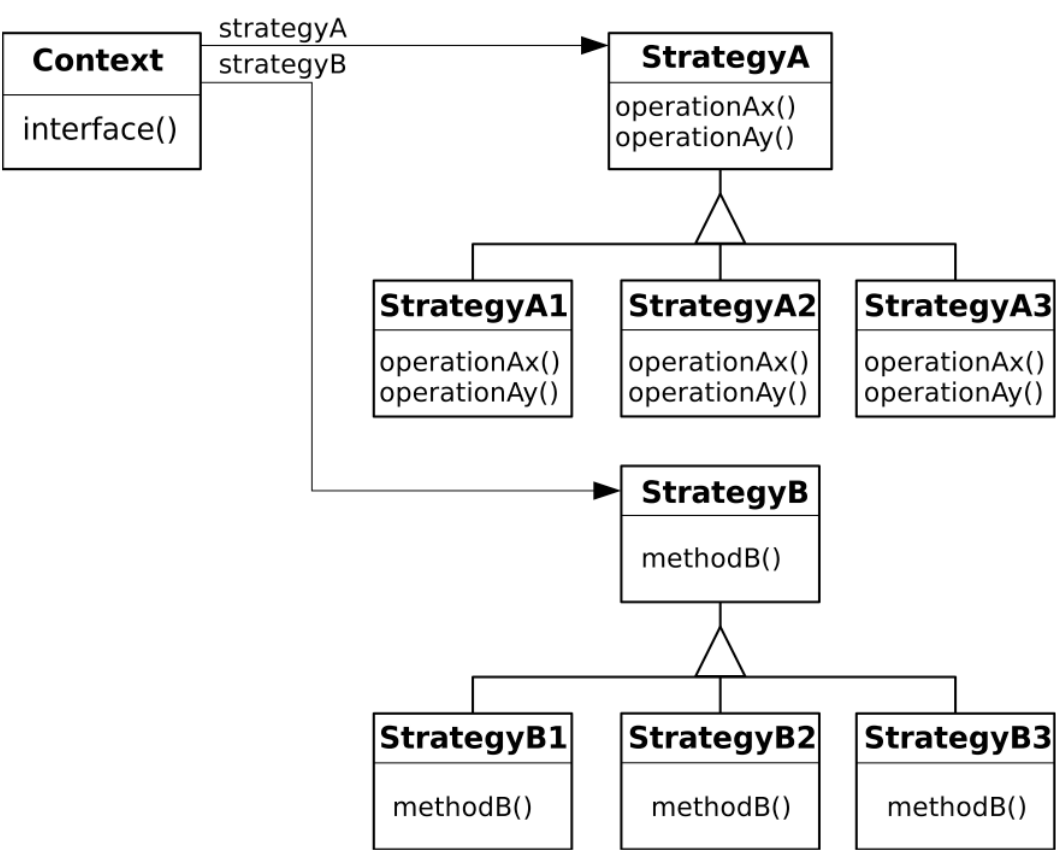


STRATEGIJA: STRUKTURNI DIJAGRAM

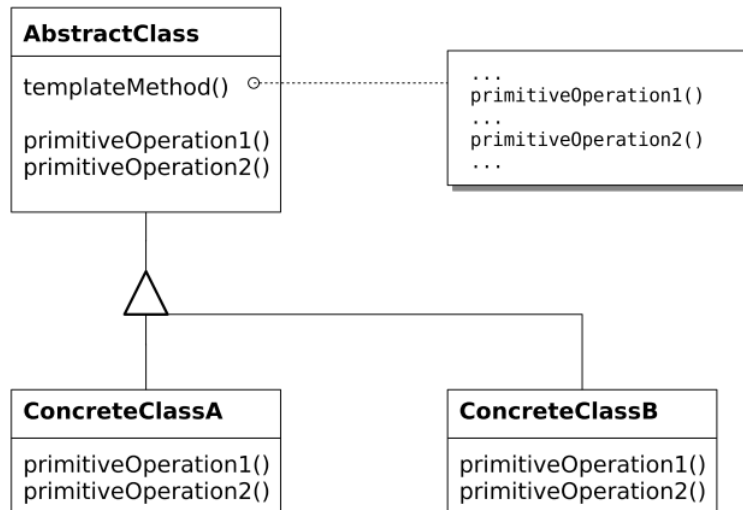


OKVIRNA METODA: STRUKTURNI DIJAGRAM

Okvirna metoda (template method, self-delegation):

ponašajni obrazac u domeni **razreda**

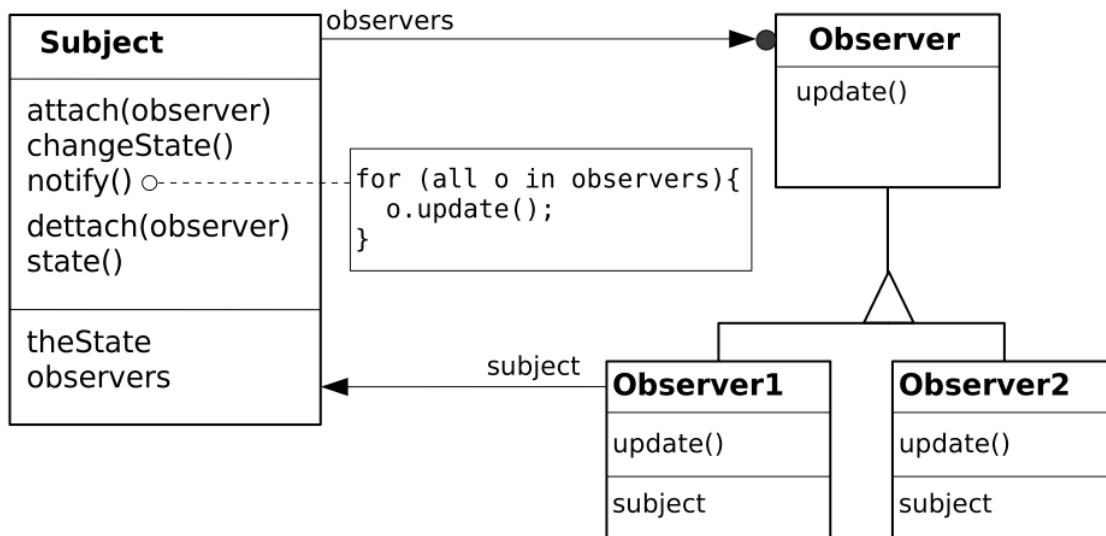
Prilagodljivost se ostvaruje nasljeđivanjem



PROMATRAČ: STRUKTURNI DIJAGRAM

Promatrač: **ponašajni** obrazac u domeni **objekata**

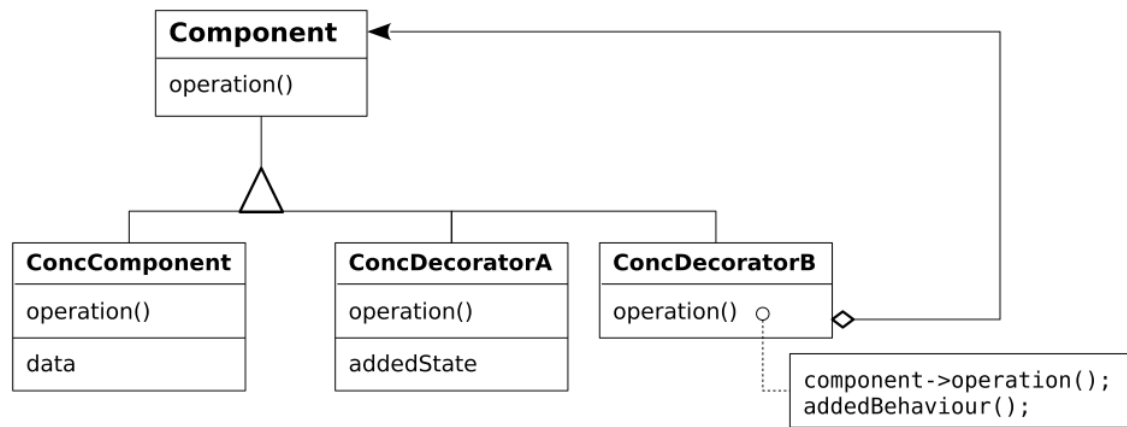
Fleksibilnost se ostvaruje kombinacijom povjeravanja i nasljeđivanja



DEKORATOR: STRUKTURNI DIJAGRAM

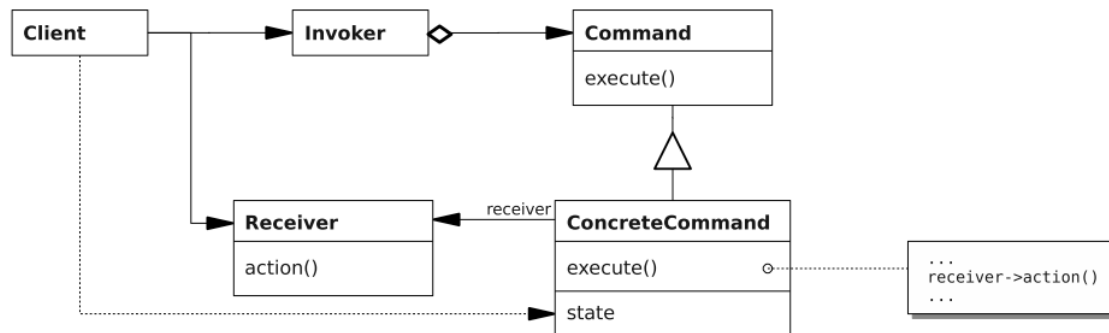
Dekorator: **strukturni** obrazac u domeni **objekata**

Fleksibilnost se ostvaruje rekurzivnom kompozicijom (handle-body)



NAREDBA: STRUKTURNI DIJAGRAM

Naredba: **ponašajni** obrazac u domeni **objekata** (akcija, transakcija)

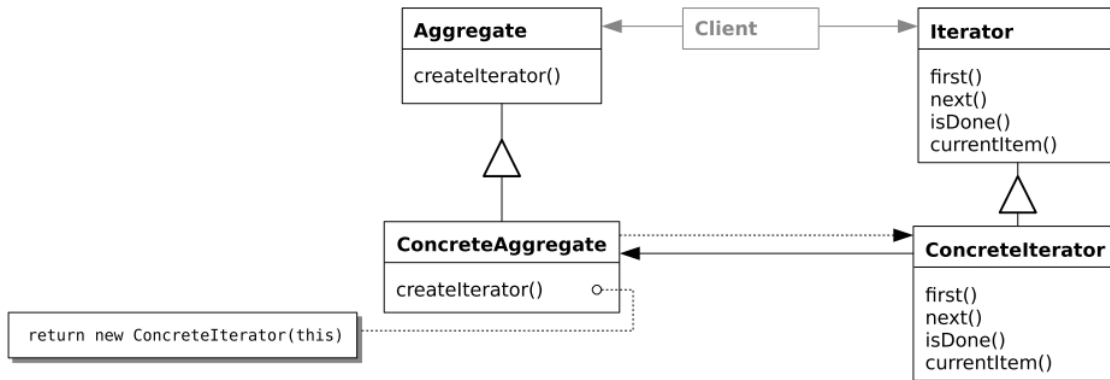


Analogija iz fizičkog svijeta (restoran brze prehrane):

- Klijent: gost (naručuje jelo)
- Naredba: narudžba (predefinirani izbor: ćevapi, lepinja, luk)
- Pozivatelj: konobar (nosi listić s narudžbom kuharu)
- Primatelj: kuhar (sprema narudžbu i izvršava je kad dođe na red)

ITERATOR: STRUKTURNI DIJAGRAM

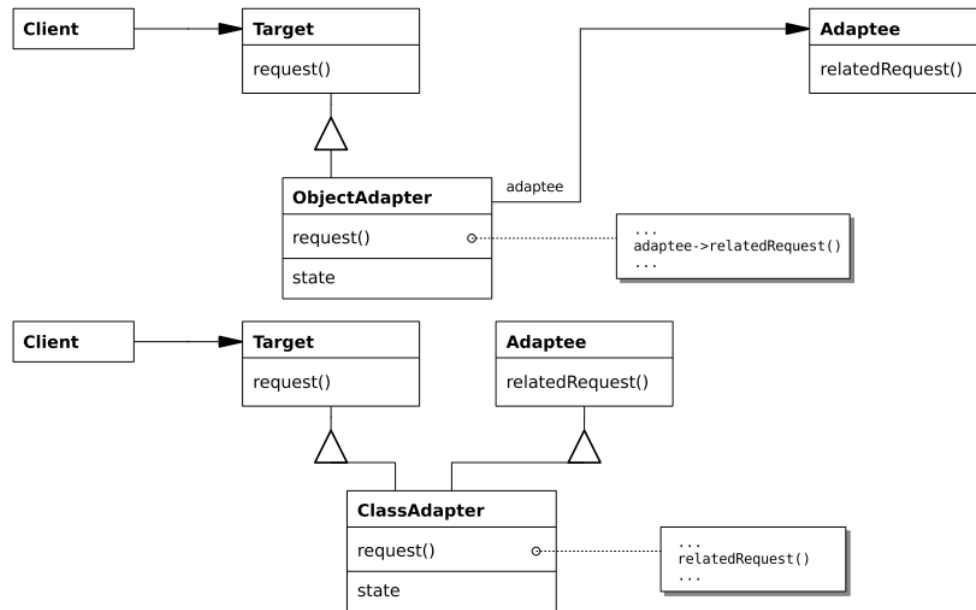
Iterator (Cursor): **ponašajni** obrazac (iterator referencira kolekciju)



PRILAGODNIK: STRUKTURNI DIJAGRAM

Prilagodnik (adapter, wrapper):

- prilagodnik **umata** vanjsku komponentu → **strukturni** obrazac
- prilagodljivost se ostvaruje povjeravanjem (ili nasljeđivanjem)



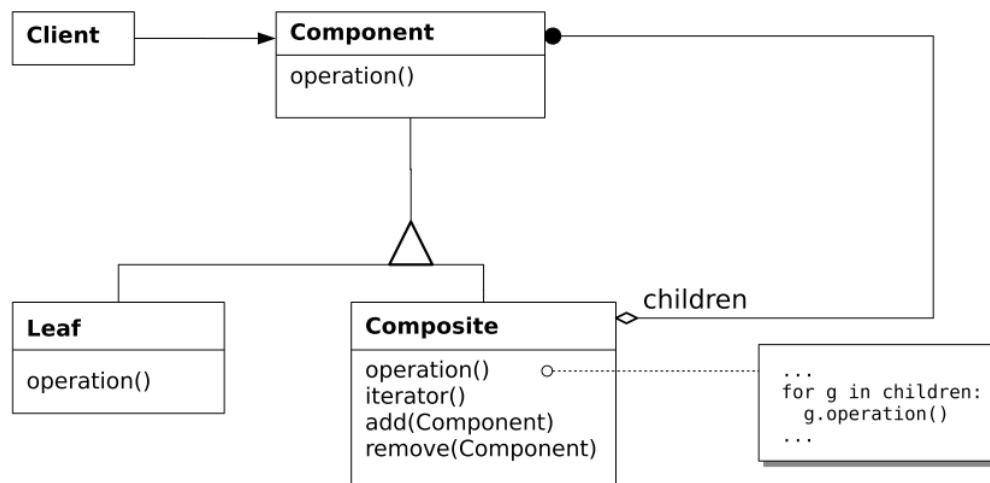
KOMPOZIT. PRIMJENLJIVOST I STRUKTURA

Primjenljivost -- Kompozit se koristi za:

- predstavljanje hijerarhije cjelina i pojedinačnih dijelova
- transparentno operiranje nad elementima takvih hijerarhija

Kompozit (composite): **strukturni** obrazac (kompozit sadrži komponente)

temeljna ideja -- rekurzivna kompozicija



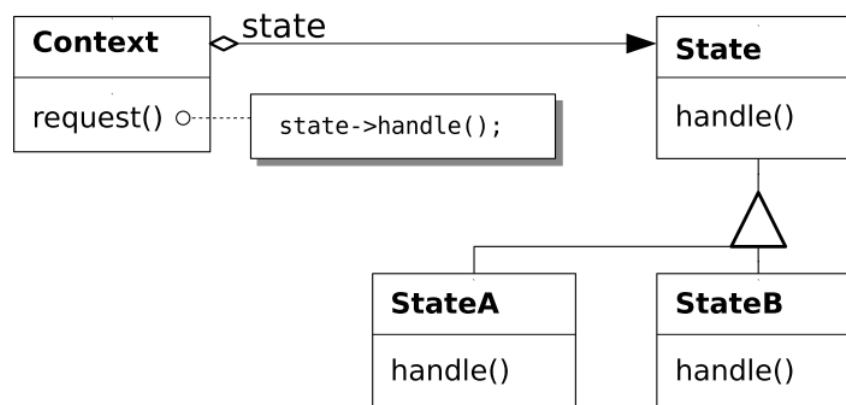
STANJE: PRIMJENLJIVOST I STRUKTURA

Primjenljivost -- Stanje se koristi kad:

- ponašanje objekta mora se dinamički uskladiti s tekućim stanjem
- metode imaju istovrsne uvjetne izraze koji ispituju stanje objekta
- obrazac smješta odgovarajuće uvjetne grane u izdvojeni razred, i time omogućava dinamičku konfiguraciju ponašanja delegacijom

Strukturni dijagram:

- stanje (state): **ponašajni** obrazac (referenciranje preko apstraktnog sučelja)

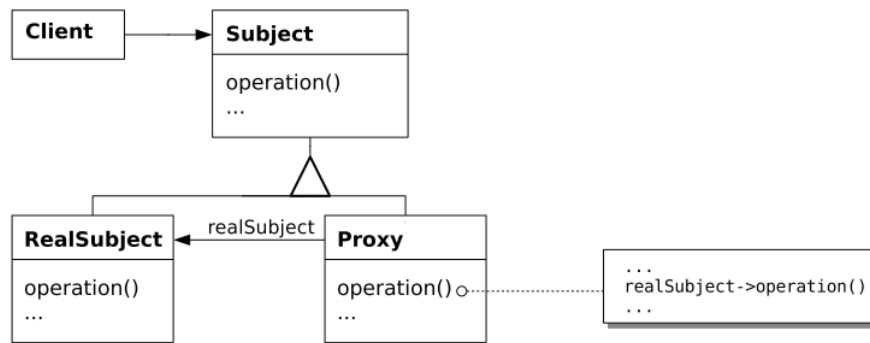


PROXY: PRIMJENLJIVOST I STRUKTURA

Primjenljivost -- podatan i/ili sofisticiran pristup resursu

- **Udaljeni** proxy: ciljni subjekt je u različitom adresnom prostoru
- **Virtualni** proxy: kreira teške subjekte na zahtjev (s odgodom)
- **Sigurnosni** proxy: omogućava pozive samo nekim klijentima
- Cacheirajući, sinkronizacijski, podstrukturni, ..., pametni pokazivač

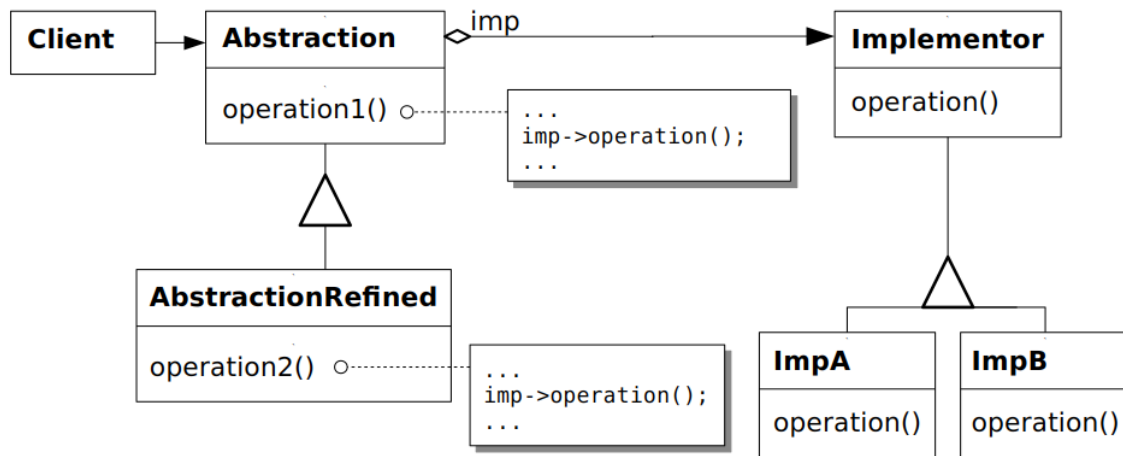
Proxy (surogat, posrednik): **strukturni** obrazac (proxy umata stvarni subjekt)



MOST. STRUKTURA

Most (bridge): **strukturni** obrazac

- lijeva hijerarhija *sadrži* izvođača iz desne hijerarhije



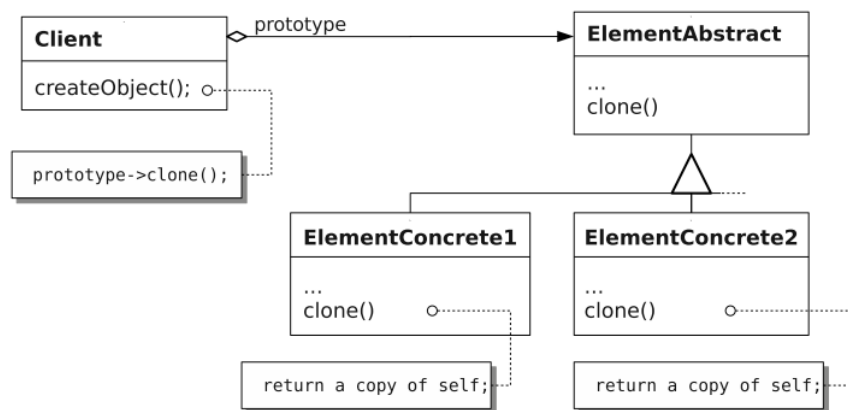
PROTOTIP: PRIMJENLJIVOST I STRUKTURA

Primjenljivost -- kad sustav mora biti neovisan o tome kako se elementi predstavljaju, komponiraju i **kreiraju**

- kada razredi koje treba instancirati nisu poznati u vrijeme pisanja programa (npr. mogu biti dinamički povezani)
- kada korisnik treba vidjeti objekt prije stvaranja

Prototip: **kreacijski** obrazac

- prototip *stvara* nove primjerke svog razreda

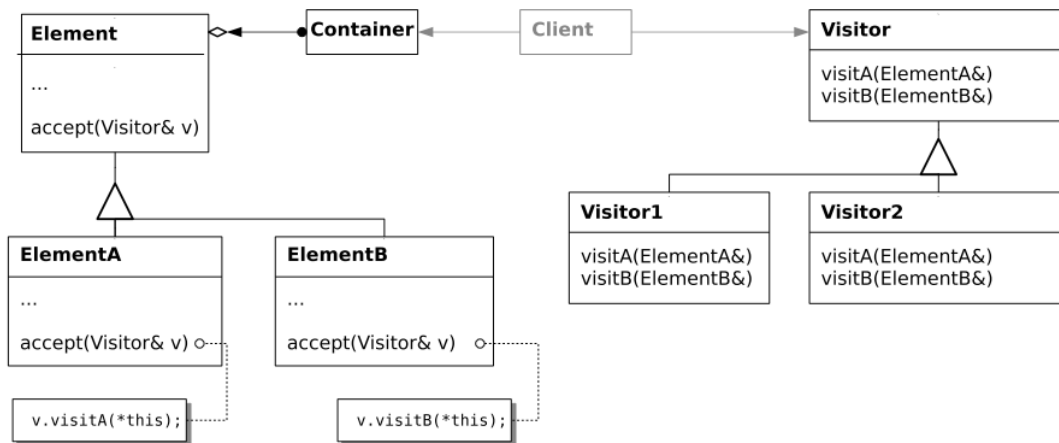


POSJETITELJ: PRIMJENLJIVOST I STRUKTURA

Primjenljivost -- polimorfni poziv koji ovisi o dva konkretna razreda

- operacije nad polimorfnim objektima ovise o konkretnim tipovima
- operacija ima mnogo, ne želimo ih uguravati u sučelje elemenata
- razredi elemenata su stabilni, operacije se mijenjaju i množe

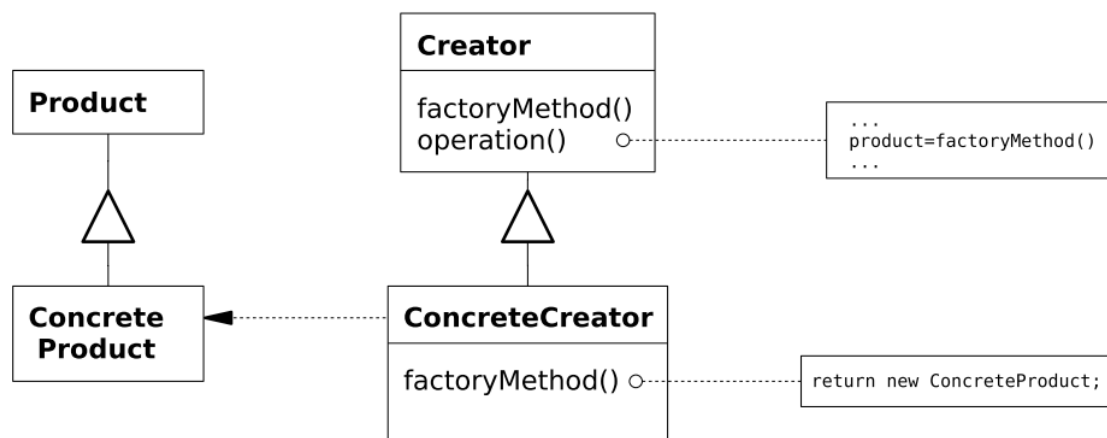
Posjetitelj: **ponašajni** obrazac (lijeva porodica prima objekt desne porodice)



METODA TVORNICA: STRUKTURNI DIJAGRAM

Metoda tvornica: **kreacijski** obrazac (kreator stvara prikladne proizvode)

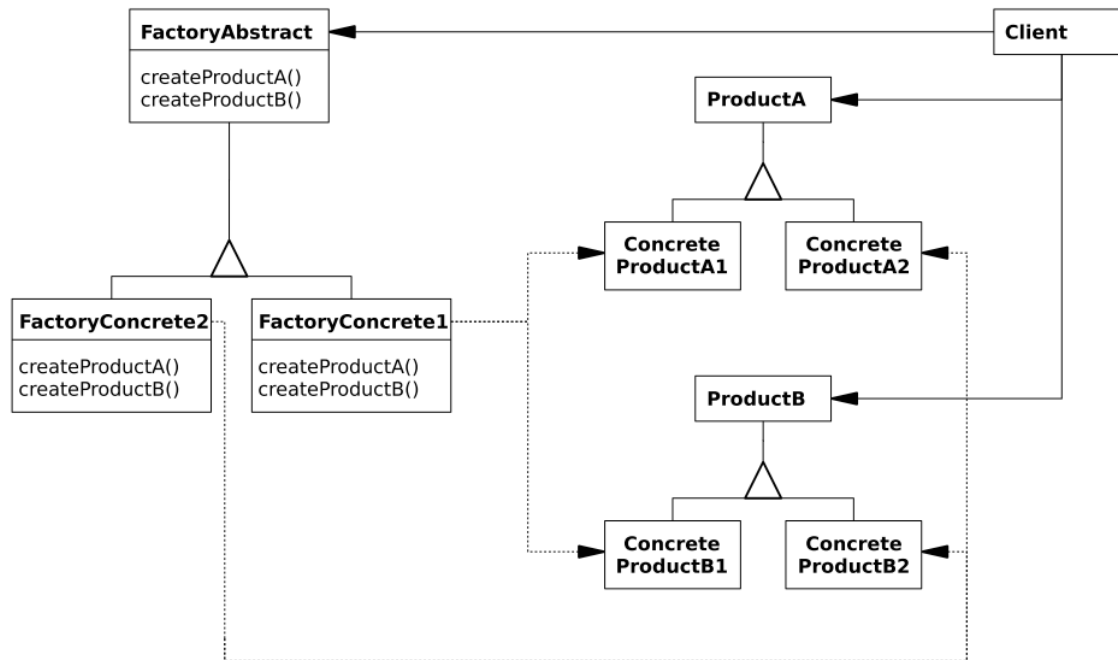
Fleksibilnost se ostvaruje nasljeđivanjem



APSTRAKтна TVORNICA: STRUKTURNI DIJAGRAM

Apstraktna tvornica: **kreacijski** obrazac (tvornica stvara objekte prikladne porodice)

Fleksibilnost se ostvaruje povjeravanjem i nasljeđivanjem



MVC: PRIMJENLJIVOST I STRUKTURA

Primjenljivost -- arhitektonski programski obrazac

- kad želimo međusobno izolirati logiku programa, korisničko sučelje i prikaz, s ciljem lakšeg održavanje programa
- prikladan i za složenije GUIjske aplikacije
- često kombinacija Promatrača, Strategije i Kompozita

MVC: **arhitektonski** obrazac, kombinacija Promatrača, Strategije, ...

