

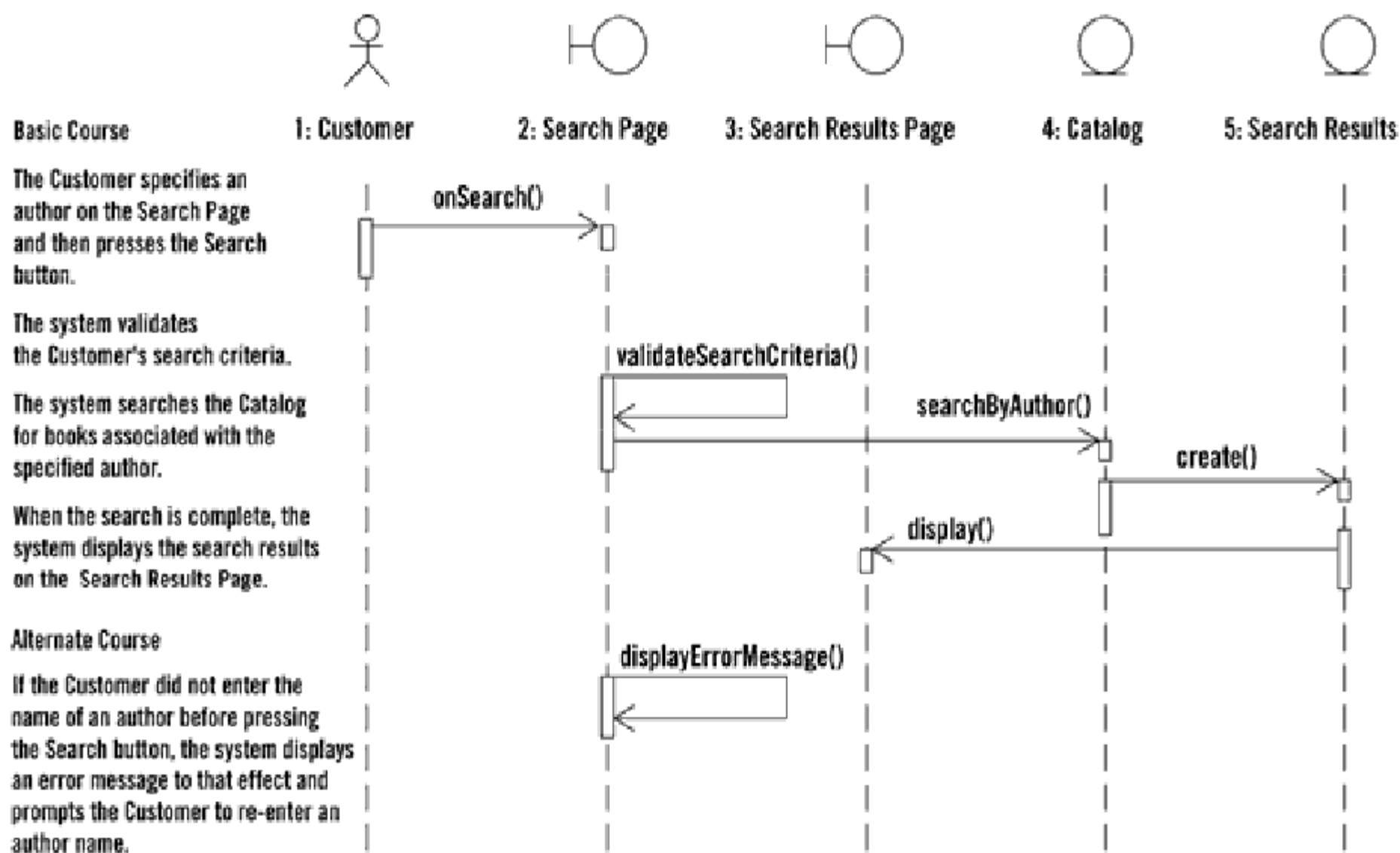
UML – dijagrami sekvence

Dijagram sekvence

- Dijagram sekvence prikazuje interakciju koja modeluje jedan scenario koji se dešava u sistemu
 - Uz dijagrame klase, najčešće korišćeni UML dijagrami
- Dijagrami sekvence proizilaze iz use case dijagrama

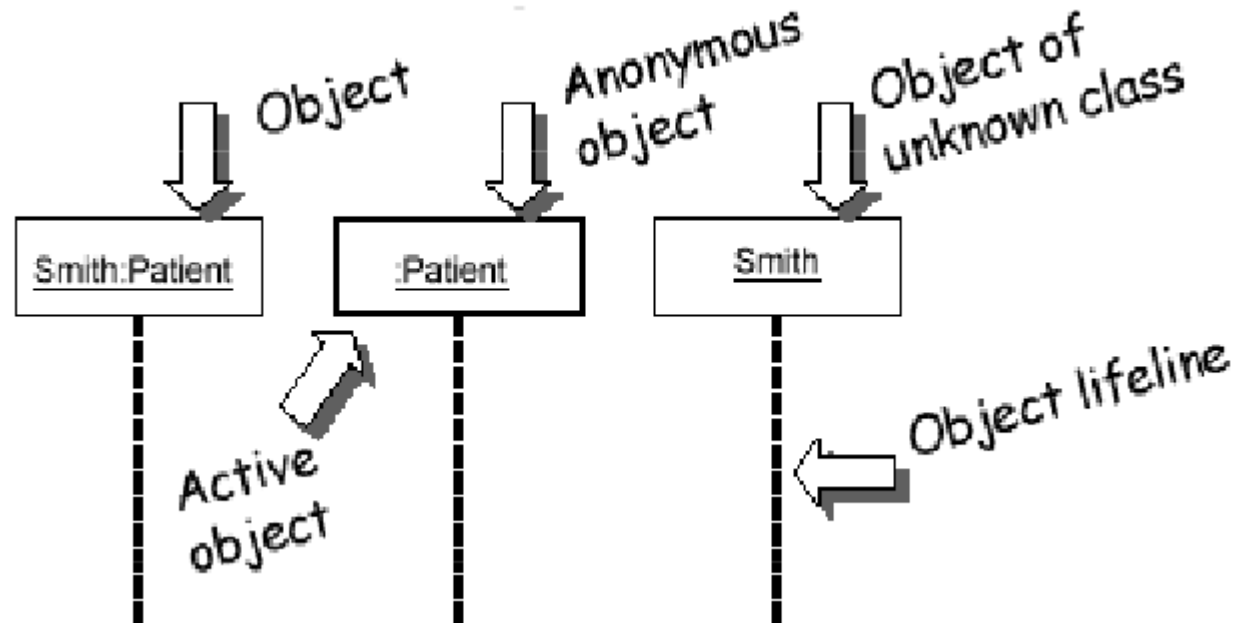
Osnovni elementi

- *Učesnici (participants)*: objekti koji učestvuju u interakciji predstavljenoj dijagramom
 - Dijagram sekvence počinje sa „found message“
- *Poruka*: komunikacija između objekata
- Ose
 - Horizontalna osa predstavlja objekte (koji od njih je aktivan)
 - Vertikalna osa prikazuju vrijeme, vrijeme teče na dolje



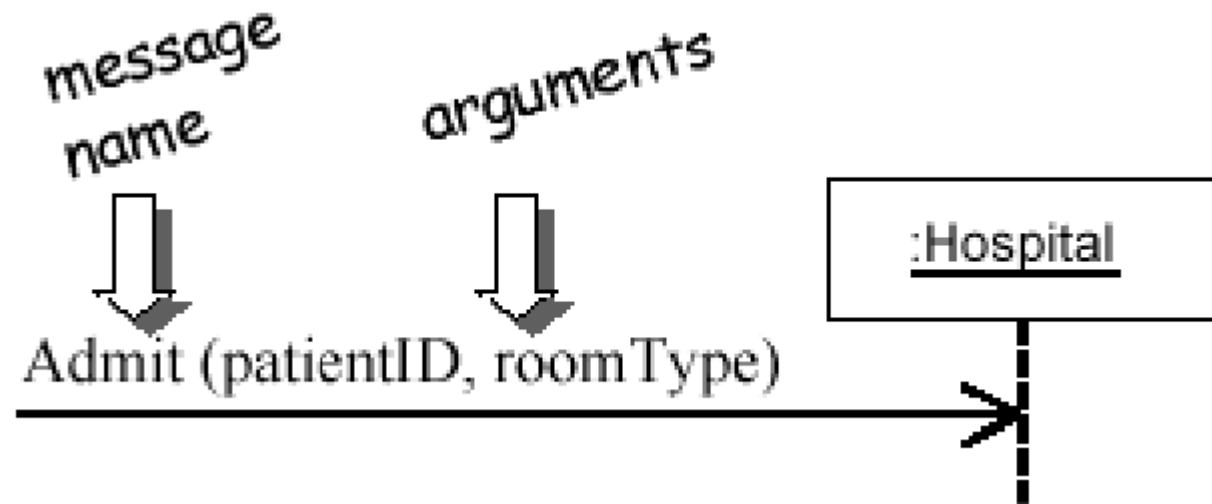
Predstavljanje objekata

- Sintaksa: *objectname:classname*
- Grafička reprezentacija je pravougaonik sa nazivom klase, opciono i nazivom objekta



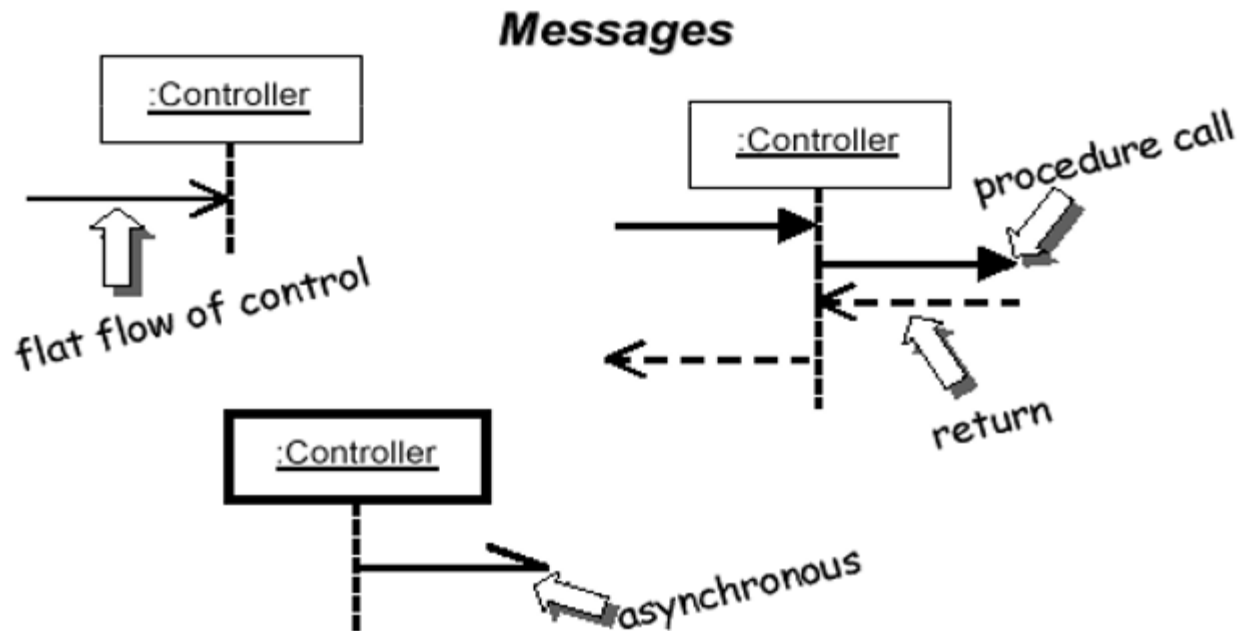
Predstavljanje poruka

- Poruke su zapravo pozivi metoda
- Predstavljaju se horizontalnim linijama



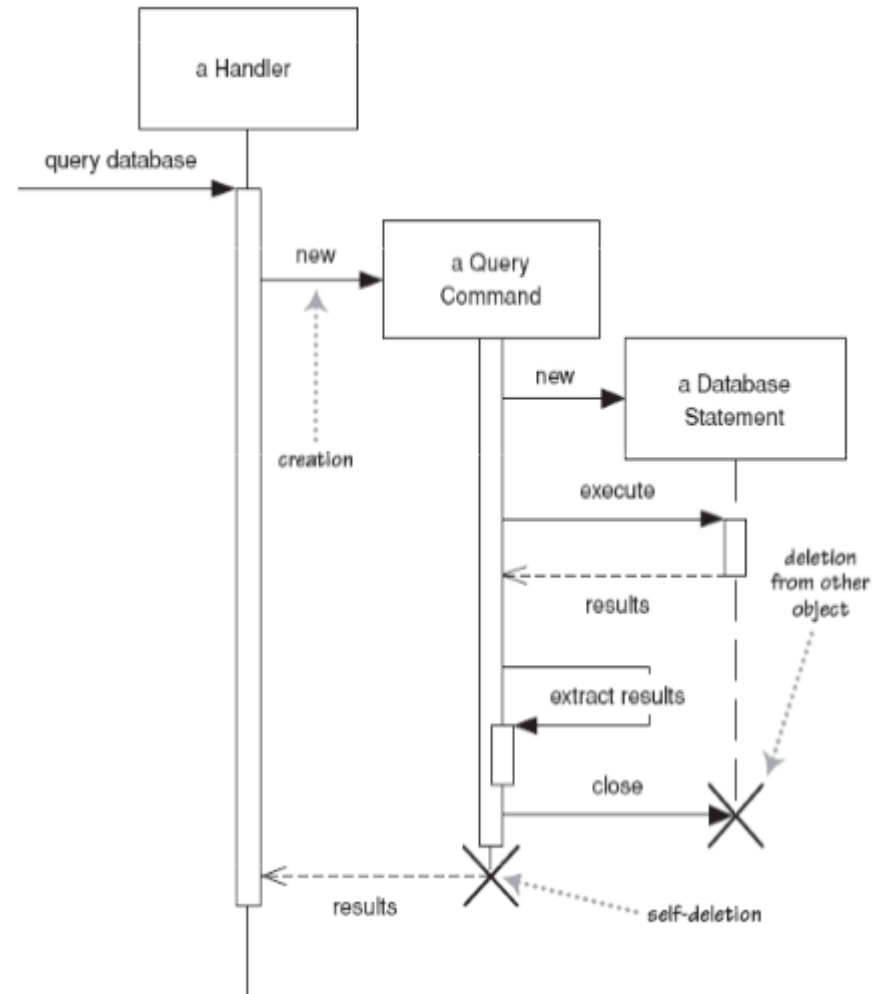
Predstavljanje poruka 2

- Poruka
 - Naziv poruke sa lista argumenata
 - Isprekidana linija „unazad“ – return
 - Sinhrono i asinhrono poruke



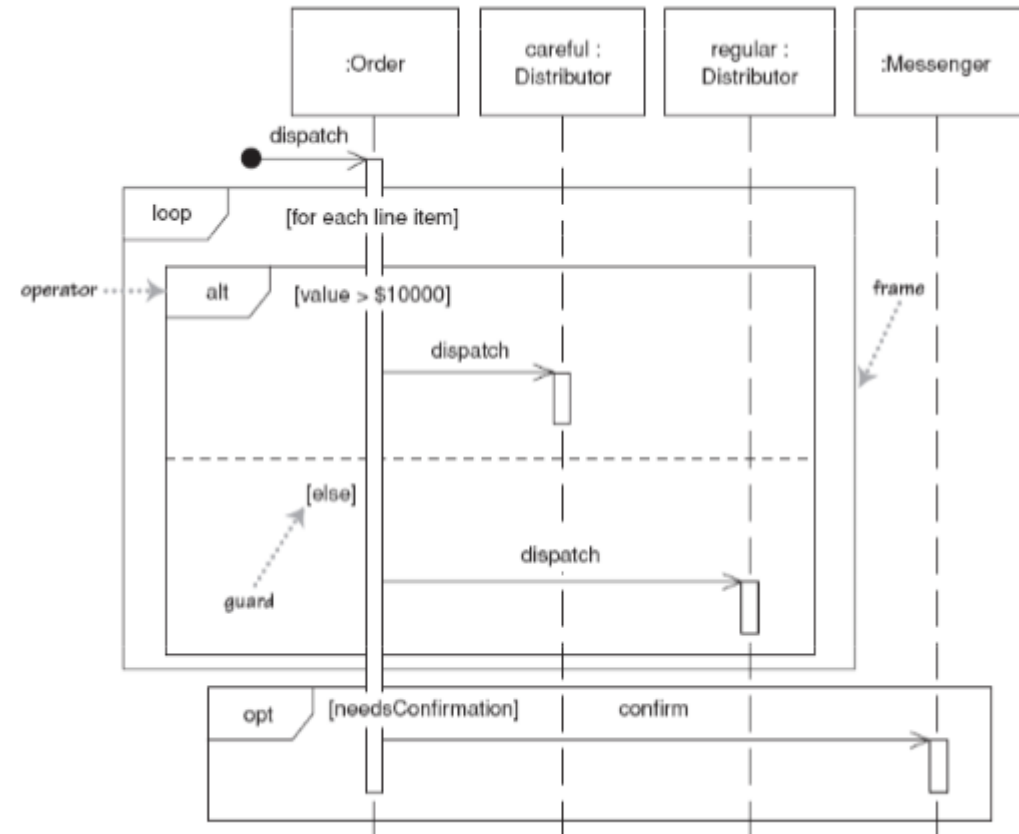
Životni vijek objekta

- Kreiranje objekta: poruka sa deskriptorom „new“
 - Objekti koji nastaju u toku interakcije prikazuju se „niže“ na vremenskoj osi
- Brisanje (uništavanje) objekta: oznaka X na kraju životne linije
 - Neki programski jezici koriste garbage collector



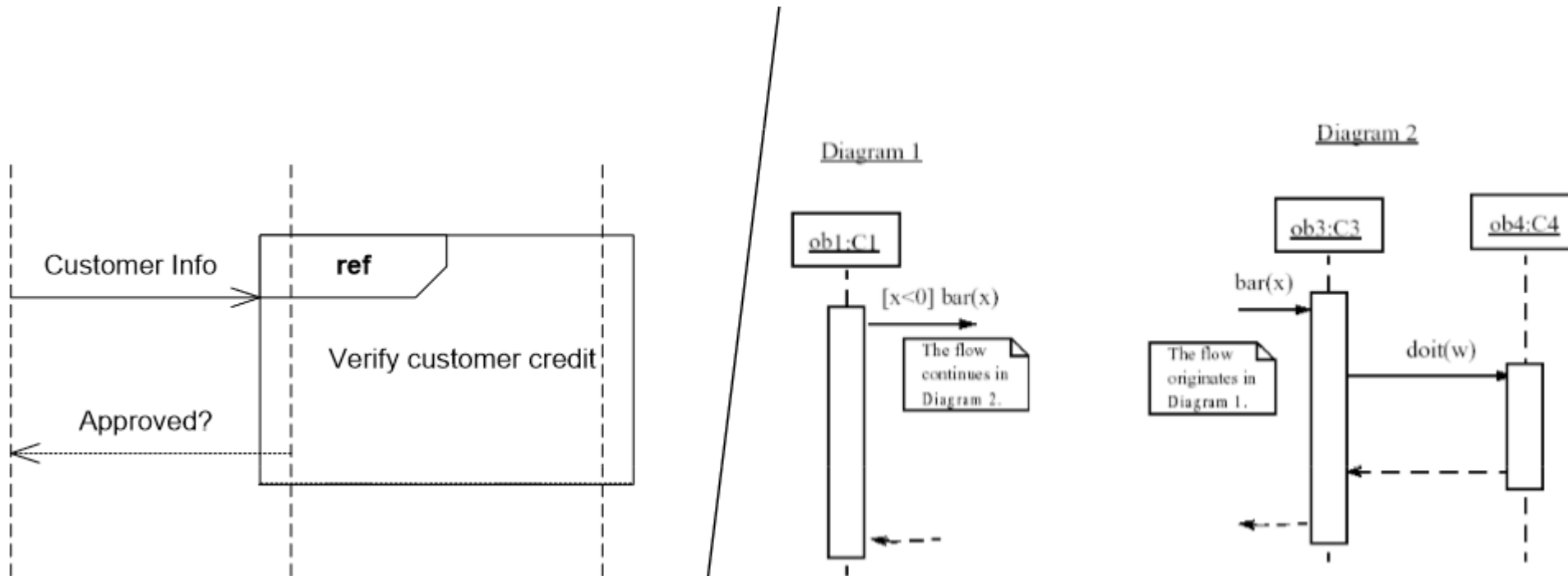
Konstrukcije if i loop

- Okvir (frame) na dijagramu sekvence prikazuje if ili loop
- if: opt [condition]
- if/else: alt [condition]
 - razdvojeni horizontalnom isprekidanom linijom
- loop: loop [condition or items to loop over]

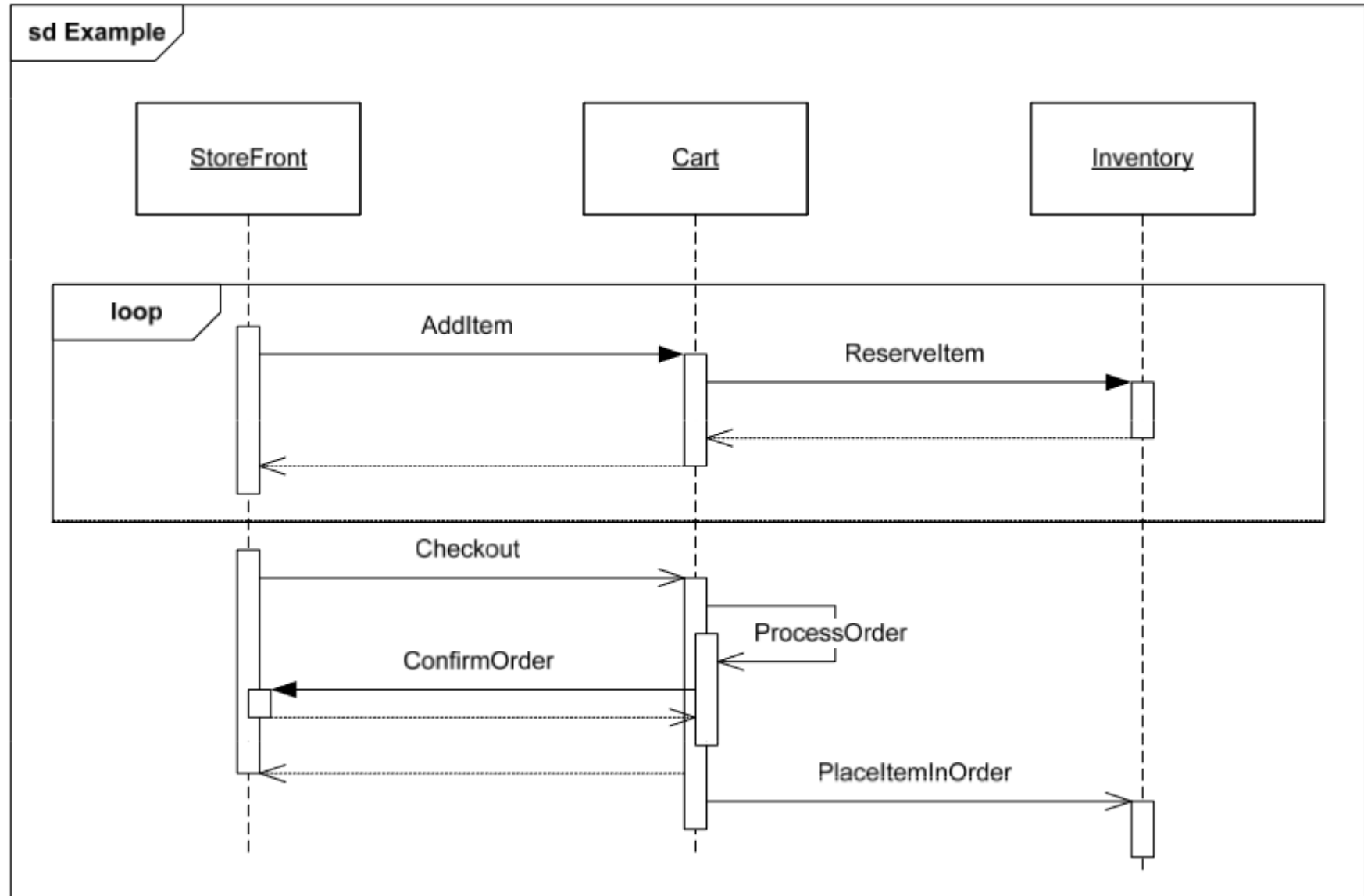


Povezivanje više dijagrama

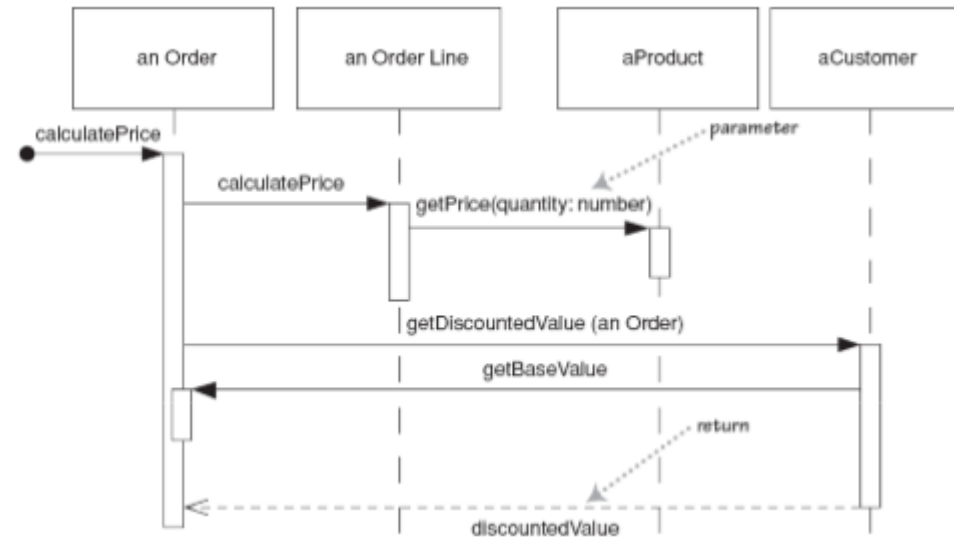
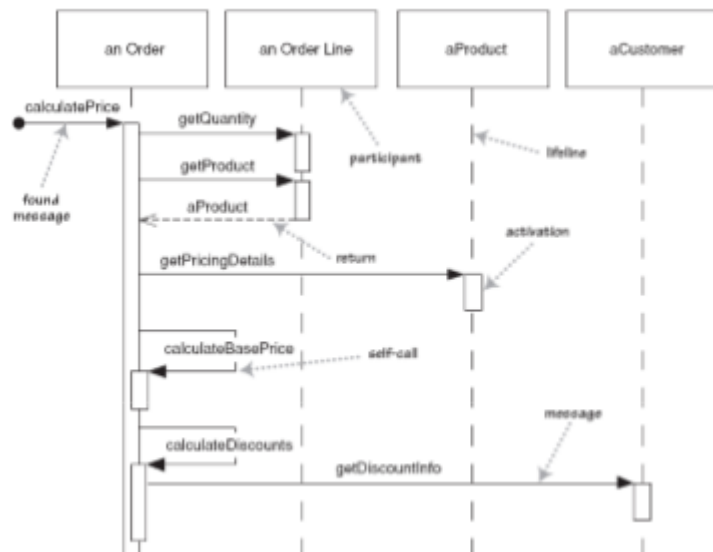
- Kada je dijagram sekvence preveliki ili referencira drugi dijagram
 - Nedovršene strelice
 - Frame „ref“



Primjer



Primjer, centralizovani vs distribuirani



Zašto dijagrami sekvence?

- Dijagrami sekvence su po detaljnosti “blizu” izvornom programskom kodu, ali
 - Dobar dijagram sekvence je ipak uopšteniji od programskog koda
 - Ne zavise od programskog jezika
 - Mogu da ih kreiraju i razumiju ljudi koji nijesu programeri
 - Daju pregledan prikaz više objekata/klasa koji čine neku interakciju

Zadatak 1

- Nacrtati dijagram sekvence za use case Start New Poker Round:
 - Scenario započinje kada igrač kroz UI pokreće novu rundu. Sistem pita da li ima novih igrača za ovu rundu, ako ima, kroz UI novi igrači se dodaju
 - Karte se promiješaju. Igrač lijevo od djelatelja postavlja početni ulog. Dije se dvije karte svakom igraču po metodi round-robin
 - Kada igrač lijevo od djelatelja nema novca za ulog biva isključen iz igre. Sljedeći igrač postavlja ulog. Ovaj postupak se ponavlja sve dok se ne nađe igrač koji može postaviti ulog, ili dok svi igrači ne budu isključeni iz igre

Zadatak 2

- Nacrtati dijagram sekvence za use case Add calendar appointment
 - Scenario započinje kada korisnik kroz UI pokreće komandu za kreiranje novog događaja u kalendaru. Sistem prepoznaje koji dio kalendara je „aktivan“ i prikazuje formu za kreiranje događaja za odgovarajući dan
 - Korisnik unosi naziv događaja, lokaciju, vrijeme početka i završetka. Sistem validira unos, nije moguće unijeti događaj bez naziva ili sa negativnim trajanjem (vrijeme početka > vrijeme završetka). Sistem evidentira novi događaj u listi za taj dan. Korisnik selektuje tip podsjetnika (reminder). Podsjetnik se dodaje u odgovarajuću listu
 - U slučaju konflikta (u specifikovano vrijeme već postoji drugi događaj), sistem prikazuje poruku i traži od korisnika da promijeni vrijeme ili poništi prethodno upisani događaj
 - U slučaju da korisnik upiše događaj istog naziva i trajanja kao postojeći zajednički događaj (meeting, više prijavljenih korisnika) sistem dodaje korisnika u listu učesnika tog sastanka