



# Programimi i distribuar

Pjesa 5 – Thirrja nga larg (remote invocation)

Prof. Asoc. Dr. Ermir Rogova

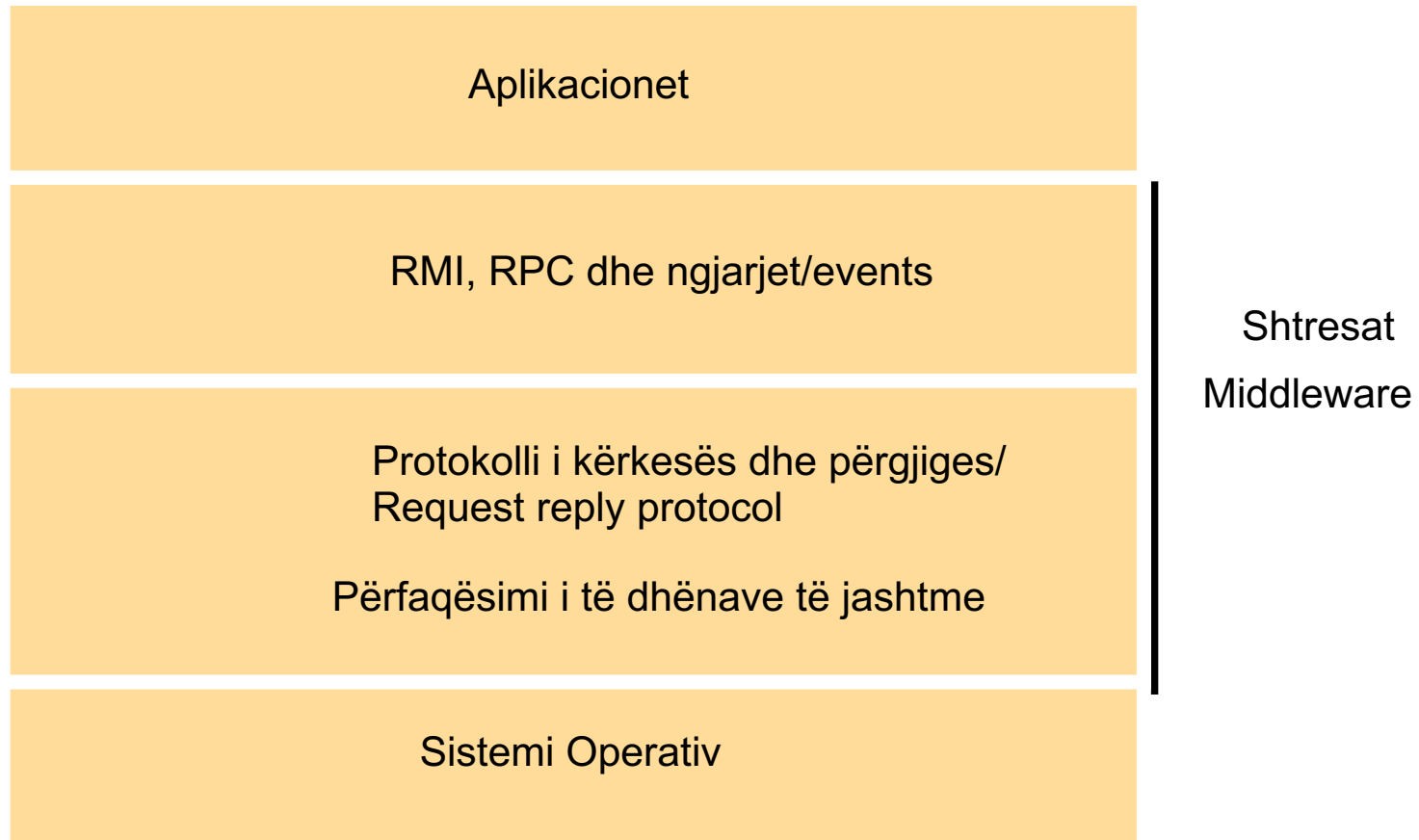
# Përmbajtja

- Hyrje
- Objektet e Shpërndara
- RMI – Remote Method Invocation
- Semantika e thirrjes - Invocation Semantics
- Proxy, Skeleton, Dispatcher
- Mbledhja e mbeturinave - Garbage Collection
- Ngjarjet dhe njoftimet
- Ngjarjet dhe njoftimet e shpërndara

# Middleware

- Softueri që lejon një nivel të programimit përtej proceseve dhe pasimit të mesazhit quhet middleware.
- Shtresat middleware janë të bazuara në protokollet dhe programimin e ndërfaqeve të aplikacioneve.
- Middleware luan rolin e urës ndërmjet sistemit operativ apo databases dhe aplikacioneve, sidomos në rrjeta.
- Disa shtresa middleware janë paraqitur në sllajdin e ardhshëm.

# Shtresat e Middleware-it



# Përfitimet e middleware

- Transparenca e lokacionit:
  - Objektet në largësi/remote duken sikur janë në të njëjtën makinë me klientin.
- Protokollet e komunikimit:
  - Klient / Server nuk ka nevojë të di nëse protokoli themelor që përdoret nga middleware është UDP ose TCP
- Computer Hardware/ Operating System:
  - Fsheh dallimet në përfaqësimin e të dhënave të shkaktuara nga pajisje të ndryshme kompjuterike ose të sistemit operativ
- Programming Languages:
  - Lejon programet e klientit dhe serverit të shkruhen në gjuhë të ndryshme.

# Interfaces / Ndërfaqet

- Gjuhët e programimit momentalisht lejojnë që programet të zhvillohen si një grup i moduleve që komunikojnë me njëri-tjetrin.
- Ndërveprimet e lejuara ndërmjet moduleve janë të përcaktuara nga ndërfaqet.
- Një ndërfaqe e specifikuar mund të implementohet nga module të ndryshme.

# Ndërfaqet në Sistemet e Shpërndara

- Kur modulet janë në procese të ndryshme ose në host-e të ndryshme ka kufizime mbi ndërveprimet që mund të ndodhin. Vetëm veprimet me parametra të specifikuara plotësisht dhe që janë kuptuar mund të komunikojnë në mënyrë efektive për të kërkuar ose ofruar shërbime për module në një tjetër proces.
- Një Shërbim i Ndërfaqes/Interface-it përcakton procedurat që janë në dispozicion të klientëve në një server. Kjo nga ana tjetër e lejon klientin të kërkojë një thirrje të një procedure të veçantë që do të performohen apo kryhen në server.
- Një ndërfaqe në largësi apo Remote Interface përcakton objektet në një server, dhe secilin prej hyrjeve dhe daljeve të argumenteve të metodave të objekteve që janë në dispozicion të klientëve.
  - Objektet në largësi mund të kthejnë objektet si argumente përsëri tek klienti.
  - Objektet në largësi mund të kthejnë referenca në objekte në largësi tek klienti.

# Ndërfaqet e objektit

- Një ndërfaqe përcakton një sërë metodash, duke përfshirë argumentet, llojet e argumentit, vlerat kthimit dhe përjashtimet.
  - Detajet e zbatimit apo implementimit nuk janë të përfshira në një ndërfaqe/interface.
  - Një klasë mund të zbatojë apo implementojë një ndërfaqe duke specifikuar sjelljen për çdo metodë në ndërfaqe/interface.
  - Ndërfaqet/Interfaces nuk kanë konstruktorë apo ndërtues.

# Gjuhët e Definimit të Ndërfaqes (IDL)

- Gjuha e definimit të ndërfaqes, ose IDL, është një specifikim formal që definon një modul në largësi:
  - Parametrat
  - Argumentet
  - Tipet
- IDL-të sigurojnë definim të objekteve në largësi, që mund të përdoren me ndonjë gjuhë programimi.
- Ndërfaqet duhet të jenë në përputhje me një IDL të përbashkët me qëllim të krijimit të një mekanizmi kërkesë-përgjigje apo request-reply.

# Modeli i Objektit

- Një program i orientuar në objekte (object-oriented) përbëhet nga një koleksion i bashkëveprimeve të objekteve.
- Objektet përbëhet nga:
  - Informacioni - një grup i të dhënave
  - Sjellja - një grup i metodave
- Objektet shpesh përfaqësojnë një koncept në botën e reale, të tilla si një libër ose një person, por gjithashtu lejon më shumë koncepte arbitrare.

# Referencat e objektit

- Objektet mund qasen nga referencat e objekteve.
- Referencat e objektit mund të:
  - Të caktohen për variablat
  - Të kalojnë apo pasohen si argumente
  - Të kthehen si rezultat i një metode
  - Specifikojë një metodë për të thirrur apo inicuar atë objekt
- Metodatat zakonisht thirren apo inicohen në lidhje me objektin me të cilin ato janë lidhur.

# Aksionet apo veprimet

- Në programimin e orientuar në objekte aksionet apo veprimet kryhen nga objekte që thirrin metoda në objekte të tjera.
- Një thirrje apo inicim mund të përfshijë informata shtesë si argumente për të kryer sjelljen të përcaktuar nga metoda.

# Efektet e metodës së thirrur

- Gjendja e objektit pranues mund të ndryshohet
- Një objekt i ri mund të instancizohet
- Thirrja e mëtejshme në metoda mund të ndodhin edhe në objekte tjera.
- Përjashtimet mund të gjenerohen në qoftë se është hasur ndonjë problem.

# Përrjashtimet/Exeptions

- Për të thjeshtuar kodin, dhe për të shmangur nevojën për të shkruar testime për të gjitha problemet e mundshme potenciale, dukuritë e papritura mund të trajtohen si përjashtime/exeptions.
- Një bllok i kodit mund të përcaktohet për të hedhur/throw një përjashtim kur gabimet ose kushtet e papritura ndodhin.
- Atëherë kontrolli kalon në kod që të kap/catches përjashtimin/exeption.
- Çdo përjashtim i kapur pastaj mund të trajtohet mirë, në vend se të shkaktoj një gabim fatal që do të shkaktonte crash apo rënie të programit.

# Garbage Collection

- Një mënyrë duhet të sigurohet për të liruar hapësirën e përdorur nga objektet kur ata nuk janë më të nevojshme.
- Kjo quhet mbledhja e mbeturinave/ garbage collection.
  - Java automatikisht zbulon dhe liron memorjen që nuk është më në përdorim.
  - C ++ i kërkon programerit për ta bërë këtë, e cila është burim i shumë gabimeve dhe dështimeve.

# Objektet e shpërndara

- Me zhvillimin e kompjuterëve, shpejtësia e procesimit, memorjes dhe ruajtjes së vazhdueshme, programet rriten përtej madhësisë që mund të menaxhohen me metodat e mëparshme.
- Modeli i objektit është zhvilluar për të trajtuar rritjen e kompleksitetit, kryesisht duke izoluar funksionalitetin dhe informacione të ngjajshme në pjesë të vogla.
- Me rritjen e aplikacioneve përtej një CPU të vetme për zgjidhje të ndërrmarjes të rrjetëzuar, praktika e orientuar në objekte e ndarjes së aplikacioneve në komponentë u bë modeli më praktik.

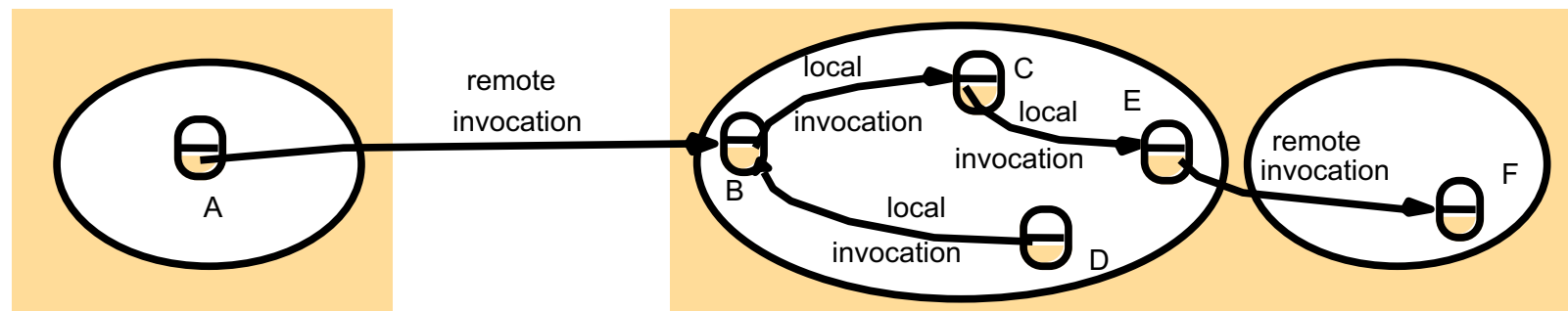
# Metoda e thirrjes në largësi RMI

- Objektet që mund të marrin apo pranojnë kërkesa në largësi për shërbime janë quajtur remote objects.
- Objektet Remote duhet të ketë një mënyrë për tju qasur apo aksesuar nëpërmjet një remote reference.
- Për ta thirrur/invoke një metodë, parametrat e saj duhet të përcaktohen përmes një ndërfaqeje në largësi apo remote interface.
- Së bashku, këto teknologji janë quajtur metoda të thirrjes në largësi remote method invocation.

# Java RMI

- Nuk duhet ngatërruar Java Remote Method Invocation me Remote Method Invocation.
  - Java RMI është një mënyrë për të thirrur metoda në largësi/invoke remote methods që është specifik për gjuhën Java.
  - RMI është emri i dhënë për thirrje të objekteve në largësi/ invoke remote methods pa specifikuar gjuhën në veçanti.

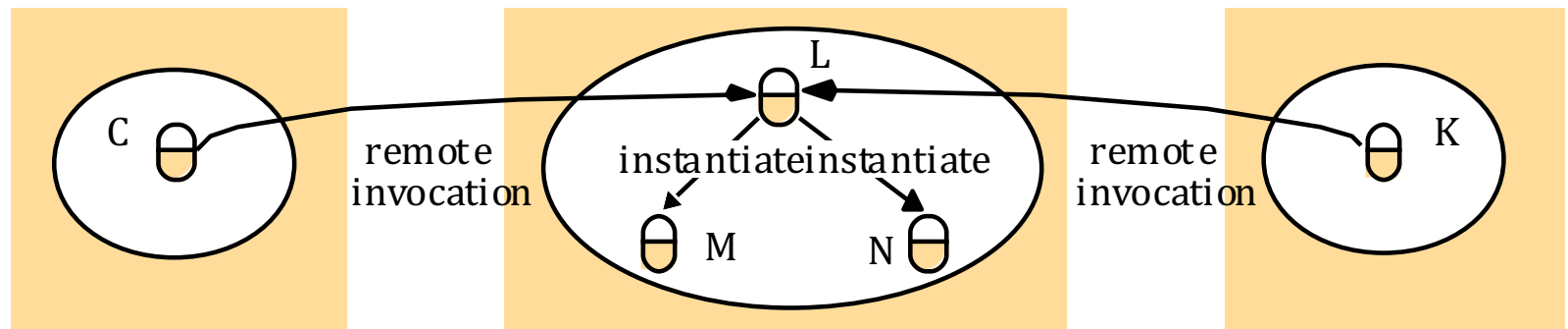
# Metoda të thirrjeve lokale dhe në largësi



# Instancizimi i Objekteve në largësi

- Duhet të mbajmë mend se një klasë është një përkufizim apo definim, ndërkohë që një objekt është një instancë e një klase, zakonisht me informacion të lidhur specifik.
- Prej kur që një objekt i largët ekziston në një proces në hapësirë tjetër nga thirrja e objektit, objekti në largësi mund të kenë nevojë të krijohet ose ngarkohet/loaded (instantiated).
- Një proces i ri gjithashtu mund të duhet të fillojë të përmbajë objekt.
- Një sistem i objektit në largësi ka nevojë për një mënyrë të instancizimit të objektit në largësi që nuk ekziston në kohën që thirret.

# Instancizimi i Objektit në Largësi



# Toleranca e gabimeve të RMI

- Një sistem i objektit në largësi gjithashtu do ti duhet një mënyrë për të zbuluar dhe trajtuar gabimet.
- Gabimet mund të ndodhin në sistemin që thirret, sistemin e thirrur, ose në rrjet, kështu që kjo është më komplekse se trajtimi i gabimit lokal.
- Egzistojnë disa masa për të siguruar tolerancën ndaj gabimeve.

# Masat për tolerancën e gabimeve

- **Retry Request Message**
  - Kontrollon a duhet të duhet të ritransmetojë një kërkesë derisa të merret një përgjigje ose të supozohet që server ka dështuar.
- **Duplicate Filtering**
  - Kontrollon kur janë përdorur ritransmetimet dhe nëse duhet filtruar kërkesat duplicate.
- **Retransmission of Results**
  - Kontrollon nëse rezultatet nga kërkesat duhet të ruhen dhe të ridërgohen në vend të ri-ekzekutimit të një kërkesë.

# Semantikat e thirrjes

## Call semantics

| <i>Fault tolerance measures</i>   |                            |                                                 | <i>Call semantics</i> |
|-----------------------------------|----------------------------|-------------------------------------------------|-----------------------|
| <i>Retransmit request message</i> | <i>Duplicate filtering</i> | <i>Re-execute procedure or retransmit reply</i> |                       |
| No                                | Not applicable             | Not applicable                                  | <i>Maybe</i>          |
| Yes                               | No                         | Re-execute procedure                            | <i>At-least-once</i>  |
| Yes                               | Yes                        | Retransmit reply                                | <i>At-most-once</i>   |

# Ndoshta

- Semantika e thirrjes ndoshta.
  - Kërkesat e metodave në largësi mund të ekzekutohen një herë ose asnjëherë.
  - Ndodh kur nuk janë aplikuar masat për tolerance ndaj gabimeve.
  - Sistemi i kërkuar nuk mund të tregojë nëse serveri ka ekzekutuar kërkesën para se të ndodhë gabimi.
  - Përdoren aty ku dështimet e rralla janë të pranueshme.

# Të paktën një herë (at least once).

- Sistemi i kërkuar merr/pranon ose një rezultat ose një përjashtim/exeption
- Serveri do të ri-ekzekutojë një kërkesë për çdo ri-kërkesë të mesazhit të marrë/pranuar
- Nuk është i përshtatshëm kur ekzekuton metodën e njëjtë shumë herë sepse do të ndryshojë rezultatin
- Shembull:
  - Një kërkesë për të tërhequr 10 \$ nga një llogari bankare nuk mund të përpunohen apo procesohen brenda afatit timeout sepse serveri është i zënë.
  - Klienti ridërgon mesazhet e kërkesës 5 herë para se serveri të fillojë përpunimin e kërkesave të reja.
  - Serveri do të ekzekutojë tërheqjen e kërkesën 5 herë.
  - Rezultati është se në vend që të tërhiqen 10 \$, 50 \$ do të mirren nga llogaria.
  - Kjo nuk është një rezultat i pranueshëm!

# Të shumtën një here (at most once)

- Sistemi i Kërkuar ose merr një rezultat ose një përjashtim
- Nëse sistemi kërkues merr një rezultat, kjo është për shkak se serveri ekzekuton rezultatin saktësisht një herë
- Përdoret kur kërkesat e mesazheve të kopjuara janë filtruar.

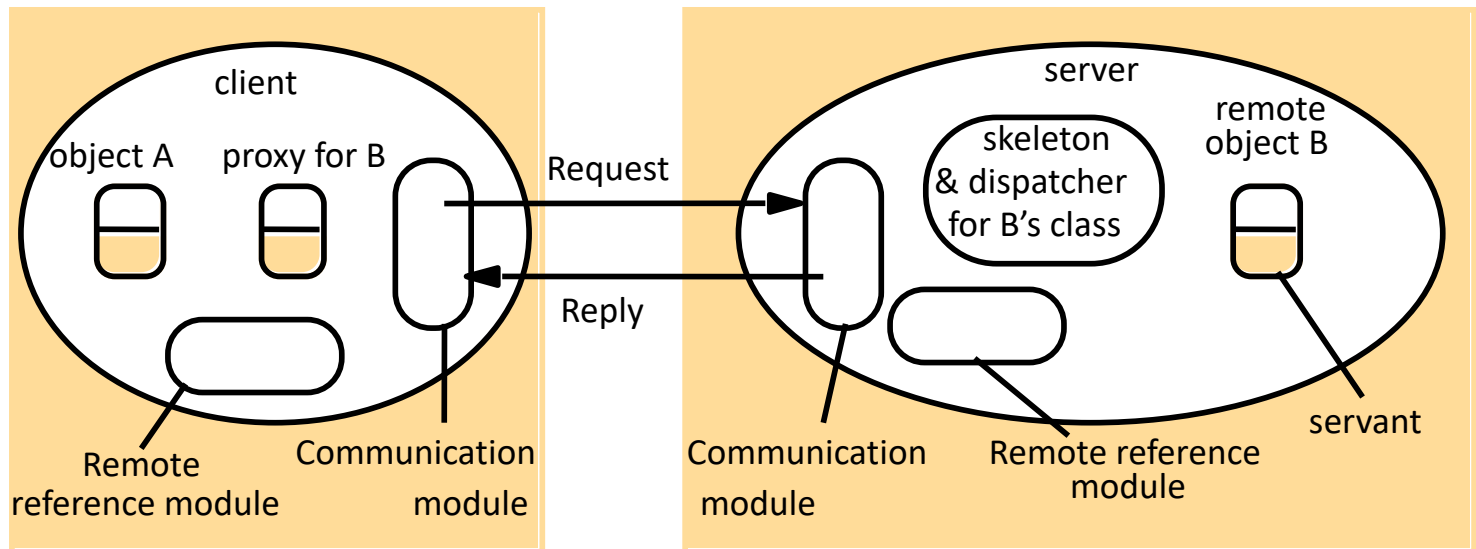
# Komunikimi dhe Modulet me referencë në largësi

- Moduli për komunikim
  - Transmeton kërkesa dhe kthen mesazhe
  - Merret me Tipin e Mesazhit(Message Type), Id e kërkesës (requestId), dhe referencën e objektit në largësi
  - Specifikon Semantikën e Thirrjes (Psh: Të-paktën-njëherë / At-Most-Once)
  - Zgjedh apo selekton dispeçer (Dispatcher) për klasën e objektit që do të thirret
- Moduli për referencë nga larg
  - Është përgjegjës për përkthimin e referencave locale dhe remote (nga larg)
  - Krijon referenca të reja të objektit në largësi
  - Mirëmban lokalen në referencat e objekteve në largësi në Tabelën e Objektit në Largësi

# Proxy, Dispatcher, & Skeleton

- Një element lokal që qëndron në një objekt të largët, si një mënyrë për tu qasur apo aksesuar quhet një proxy.
- Një Dispatcher/dispeçer merr/pranon një mesazh të kërkuar nga Moduli i Komunikimit dhe përdor `methodId` mbi kërkesën për të zgjedhur metodën e nevojshme.
- Një element në largësi që pranon një thirrje në rrjet nga një proxy dhe lidh atë në një objekt të largët është quajtur një skeleton.

# Proxy dhe skeleton në RMI



# Binder

- Aplikacionet e klientit kanë nevojë për një mënyrë për të marrë të paktën një referencë në objekte të largëta që janë në dispozicion në server.
- Një Binder është një shërbim që është i ndarë edhe nga aplikacionet e klientit edhe nga aplikacionet e serverit që ofrojnë një hartim/mapim midis emrave të objekteve në largësi që klienti duhet të specifikojë dhe referencave të objekteve në largësi që janë në dispozicion në server.
- Për shembull, në Java binder është RMIregistry

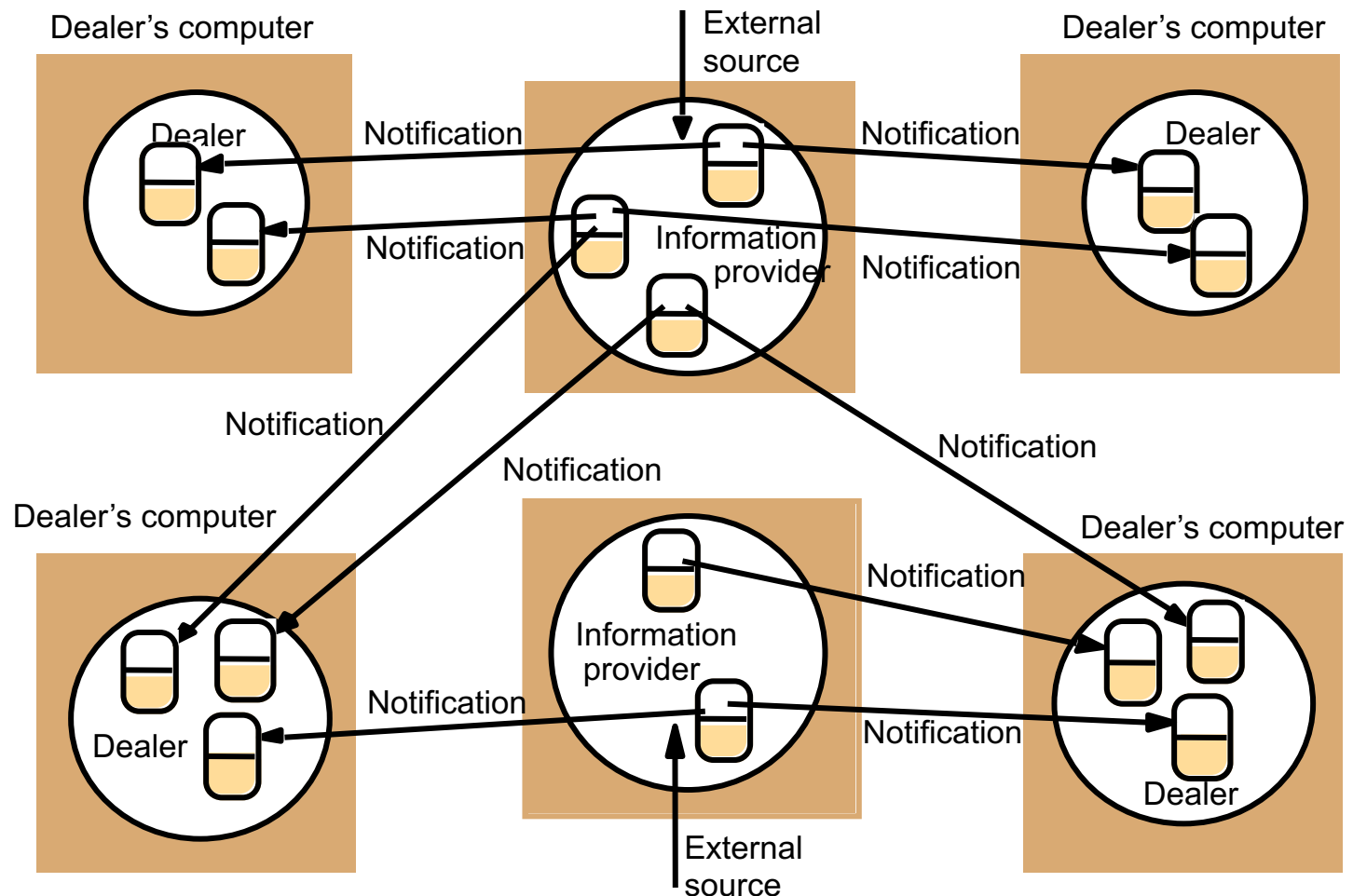
# Ngjarjet dhe njoftimet

- Manipulimi i një objekti që rezulton në një ndryshim të gjendjes është quajtur një ngjarje apo Event.
- Objektet që shfaqin një pamje të gjendjes aktuale iu dërgohen njoftime apo Notifications kur ndodh një ngjarje.
- Shembull: Një përdorues duke klikuar një buton paraqet një ngjarje që shkakton/triggers një njoftim/notification në butonin e objektit.

# Ngjarjet dhe njoftimet e shpërndara

- Sistemet e shpërndara përdorin Ngjarje dhe Njoftime në metodologjinë Publish-Subscribe
  - Klientët mund të bëjnë Subscribe në server në mënyrë që të dëgjojnë ngjarjet që mund të ndodhin në serverin e objektëve në largësi.
  - Kur një ngjarje në server mund të ndodhë, serveri mundet të Publikojë ngjarjen të gjithë klientëve që kanë bërë subscribe.
  - Shpërndarësi apo botuesi specifikon çfarë lloj i tipit të ngjarjes zë vend si pjesë e mesazhit që është dërguar. Kjo lejon klientët që të regjistrohen apo bëjnë subscribe për ngjarje specifike apo të veçanta në të cilat janë të interesuar, dhe injorojnë ngjarjet e tjera.

# Dealing room system



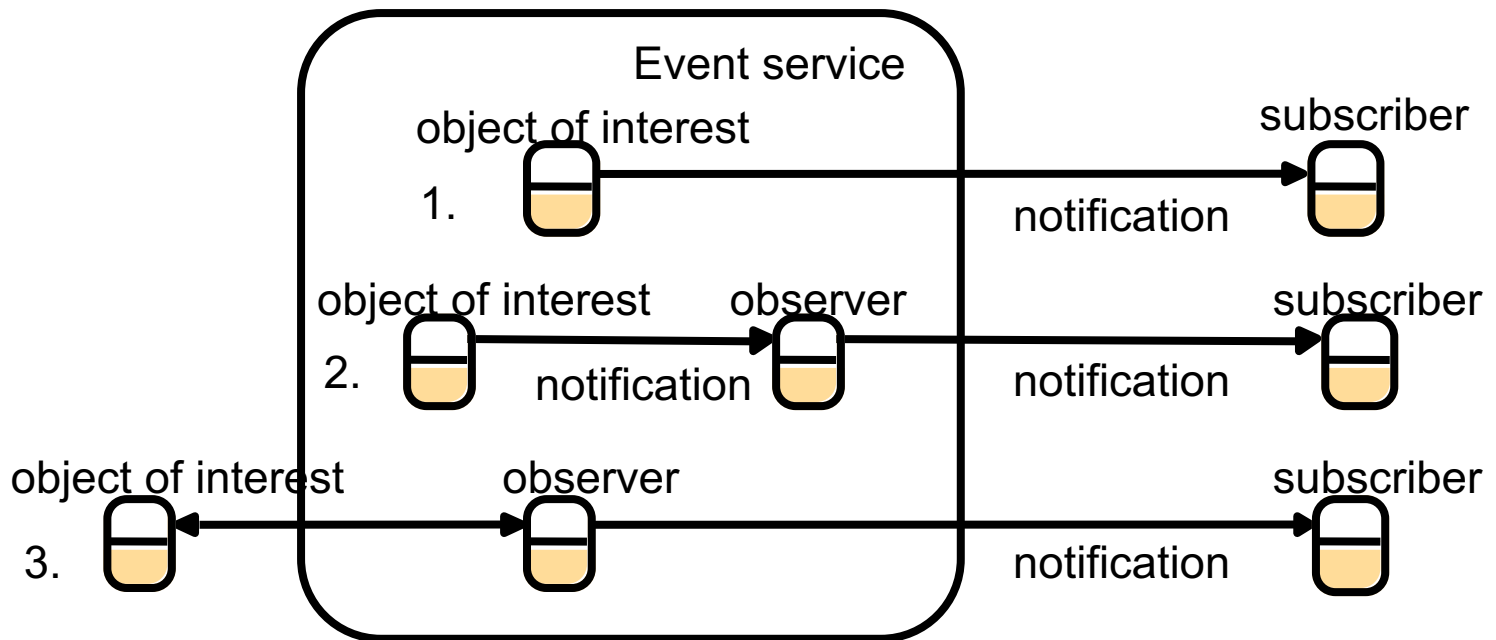
# Vëzhguesit/Observers

- Është e mundur që të dërgohen njoftime direkt nga një objekt i interesit të një regjistruesi/subscribers, por kjo mund ti bëjë teknikat e avancuara të publikimit vështirë të menaxhueshme.
- Një vëzhgues është një ose më shumë objekte që qëndron në mes të një objekti të interesit dhe të regjistruesëve që mund të filtrohen dhe njoftoj ngjarjet me mesazh para se të arrijnë regjistruesit.
- Vëzhguesit mund të sigurojnë:
  - Forwarding: Objektet e interesit dërgojnë një njoftim të vetëm për një vëzhgues. Vëzhguesi trajton në pjesë më të madhe dërgimin e njoftimit, duke lënë objektin e interesit të vazhdojë përpunimin apo procesimin pa vonesë.
  - Filtrim: vëzhguesi mund të përdor logjikën vetëm për të dërguar një pjesë të vogël të njoftimit të regjistruesve.
  - Kutitë postare të Njoftimit / Notification Mailboxes : Vëzhguesit mund të ruajnë njoftimin deri sa regjistruesit të jenë gatshëm të konsumojnë mesazhet përderisa objekt i interesit mund të vazhdojë me përpunimin apo procesimin e pa pengesa.

# Shembull i Njoftimit të Ngjarjes

- Sllajdi i ardhshëm demonstroi skemat e mundshme të njoftimit të ngjarjes:
  - Objekti i interesit publikon njoftimet direkt për regjistruarit/subscribers.
  - Objekti i interesit boton apo publikon në një vëzhgues. Vëzhguesi trajton publikimin për regjistruarit.
  - Objekt i interesit është jashtë shërbimit lokal të ngjarjes. Një vëzhgues periodikisht pyet objektin e interesit për të parë nëse një ngjarje ka ndodhur. Vëzhguesi do të publikojë njoftime për regjistruarit nëse ajo gjen një ngjarje/event.

# Njoftimet e ngjarjes të shpërndara





# Pyetje???