

Virtualna okruženja

Umrežena virtualna okruženja

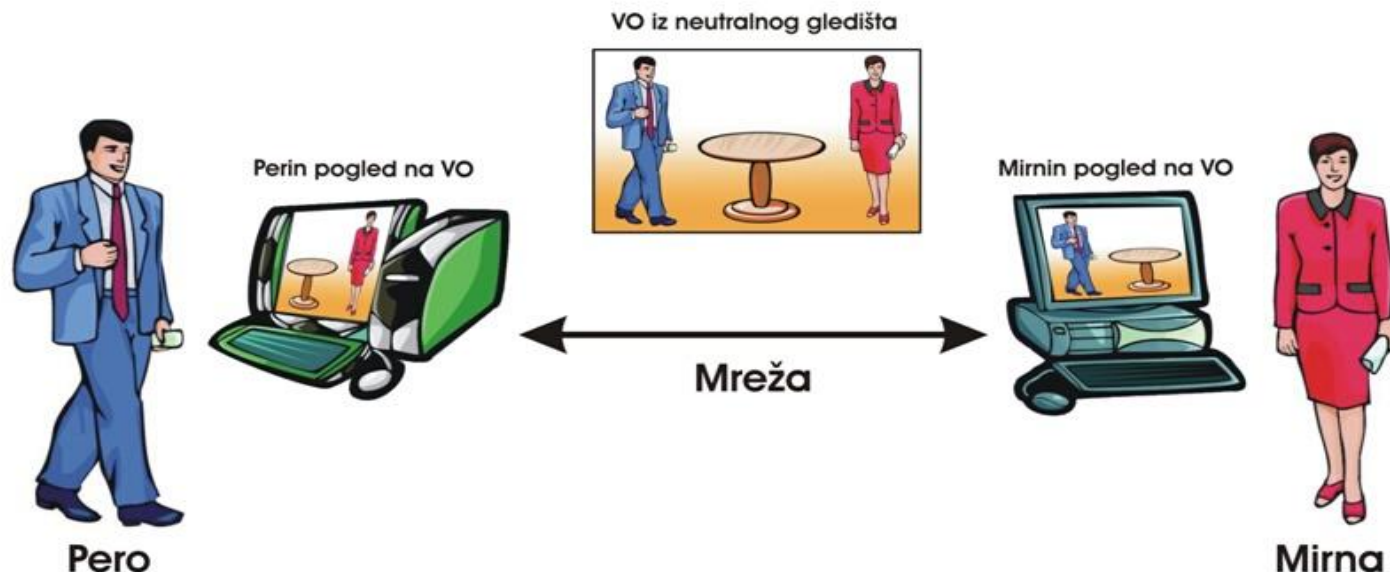
Prof. dr. sc. Igor S. Pandžić

Prof. dr. sc. Maja Matijašević

dr. sc. Mirko Sužnjević

- ◆ Uvod
 - Definicija
 - Tehnički izazovi
 - Osnovni model UVO
- ◆ Upravljanje zajedničkim dinamičkim stanjem
- ◆ Oblikovanje programskog rješenja UVO
- ◆ Virtualni svjetovi
 - Mrežni zahtjevi
 - Korisnička populacija
 - Poslovni modeli

Umrežena virtualna okruženja



- ♦ Fizički udaljeni korisnici sudjeluju u zajedničkom virtualnom okruženju
- ♦ Svako računalo ima lokalnu kopiju okruženja
- ♦ Svaki korisnik upravlja svojim 3D prikazom i okruženjem
- ♦ Sve kopije okruženja se međusobno sinkroniziraju putem mreže
- ♦ Korisnici vide jedni druge jer su grafički prikazani u okruženju

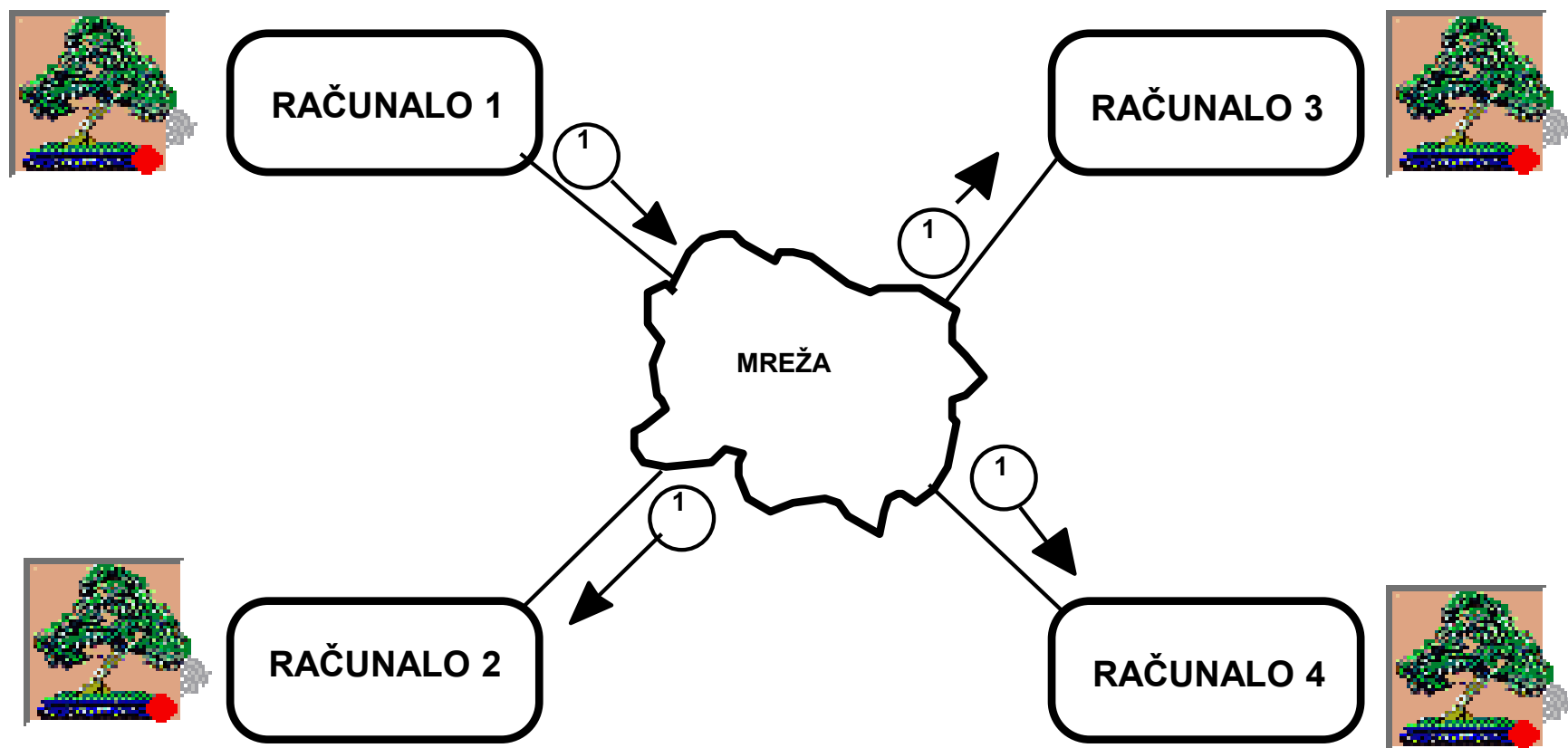
Kako korisnici doživljavaju UVO-a?



Zavod za telekomunikacije

- ◆ Doživljaj zajedničkog prostora
 - Korisnici osjećaju da su u istom (stvarnom ili zamišljenom) prostoru
- ◆ Doživljaj zajedničkog prisustva
 - Svaki od korisnika ima svoju reprezentaciju unutar virtualnog svijeta
- ◆ Doživljaj zajedničkog vremena
 - Događaji (izgledaju kao da) se događaju u isto vrijeme
- ◆ Mogućnost komunikacije
- ◆ Mogućnost interakcije
 - S virtualnim svijetom i drugim korisnicima

Kako radi umreženo virtualno okruženje



- ♦ Virtualni svjetovi – popularni u 2000-tima, danas ih ponovno popularizira virtualna stvarnost

- Druženje i društvene mreže (Facebook Horizon - <https://www.oculus.com/facebookhorizon/>)
- Virtualna telekonferencija/zajednički rad (Terf - <https://www.3dicc.com/>)
- Učenje/obuka na daljinu (Second Life - <https://secondlife.com/>)
- Virtualni svjetovi za djecu (Club Penguin - ugašen)
- ...

- ♦ Umrežene višekorisničke igre – primarna svrha je zabava



Razlike između igara i virtualnih svjetova



Zavod za telekomunikacije

- ♦ Svrha
- ♦ Perzistentnost
 - Virtualni svjetovi su uglavnom perzistentni – postoje i mijenjaju se i dok korisnik nije u njima
 - Kod igara je puno češće iznova stvaranje virtualnog svijeta iz pohranjenih podataka (primjerice svaka bitka u Call of Duty se odvija na istoj mapi koja se iznova kreira)
- ♦ Mijenjanje 3D svijeta
 - Virtualni svjetovi dopuštaju dodavanje novih slika, tekstura, zvukova, videa i 3D objekata (primjerice Second Life)
 - Igre ne dopuštaju dodavanje novih informacija u svoj klijent (iako mogu dopuštati mijenjanje svijeta primjerice Fortnite)
- ♦ Karakteristike mrežnog prometa – jako ovisne o „promjenjivosti” 3D svijeta jer ako korisnici mogu dodati nove informacije, drugi korisnici moraju te iste informacije preuzeti

- ◆ Prikaz korisnika
- ◆ Interakcija korisnika u virtualnom okruženju
- ◆ Podrška za prirodnu komunikaciju
- ◆ Karakteristike tehnologije i načina umrežavanja
- ◆ Prilagodljivost veličini (engl. *scalability*)

Prikaz korisnika

Zavod za telekomunikacije



- ◆ Lik korisnika unutar virtualnog okruženja se obično naziva avatar
- ◆ Od riječi avatāra iz sanskrita koja znači inkarnaciju ili manifestaciju nekog od bogova u tijelu čovjeka ili životinje
- ◆ Prvi put avatar se koristi u ovom značenju 1985 u igri *Ultima IV: Quest of the Avatar*
- ◆ Oblici avatara variraju od jednostavnih geometrijskih do složenih humanoidnih avatara visokih detalja
- ◆ Karakteristike avatara često definiraju i mogućnosti interakcije kao i posredno na količinu podataka koje se prenose mrežom
- ◆ Današnji avatari najčešće imaju niz gesti i izraza lica kroz koje korisnici mogu komunicirati s drugima (primjerice ples)

Interakcija korisnika u virtualnom okruženju

- ◆ Načini interakcije zadani karakterom aplikacije
- ◆ Učestali načini interakcije
 - Kretanje
 - Geste
 - Stvaranje zvukova
 - Interakcije s predmetima
 - Interakcije s drugim korisnicima
- ◆ Mehanizmi za upravljanje pravom pristupa – ako više korisnika pokušava udariti virtualnu loptu kako odrediti tko je uspio?
- ◆ Vremenski okvir interakcije?



- ◆ Većina UVO-a podržava pismenu komunikaciju
- ◆ Komunikacija putem gesti je također vrlo česta
 - Mahanje
 - Ples
 - Potvrda i negacija
 -
- ◆ U posljednje vrijeme UVO-a također donose ugrađenu podršku za glasovnu komunikaciju, a ista može biti i ograničena pozicijom avatara unutar UVO (ukoliko su avatari dva korisnika predaleko neće se čuti)

Karakteristike tehnologije i načina umrežavanja



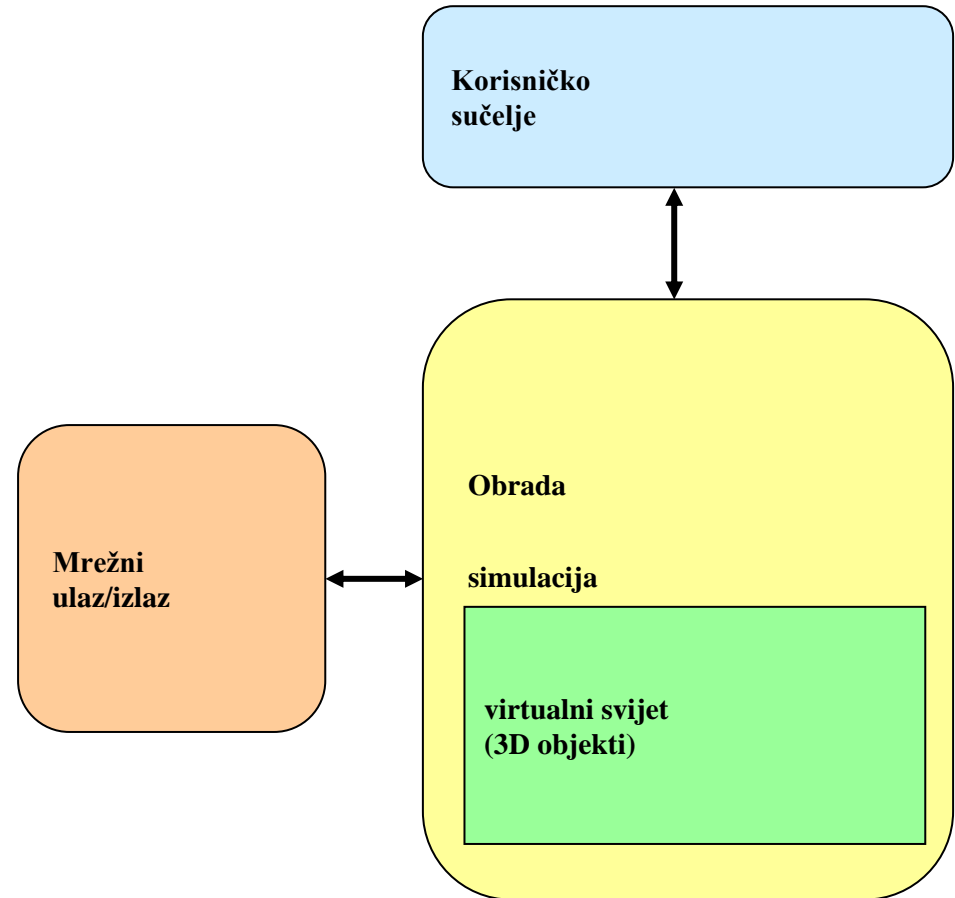
Zavod za telekomunikacije

- ◆ Korisnici istog UVO-a mogu koristiti pristupne uređaje i mreže heterogenih karakteristika
- ◆ Pristupni uređaji imaju definirane minimalne uvjete (minimalnu konfiguraciju) koja podržava pristup određenom UVO
- ◆ Karakteristike mrežnih parametara variraju ovisno o tipu UVO te o njegovim karakteristikama
- ◆ „Sakrivanje” mreže i njenih utjecaja je najveći tehnički izazov UVO-a
- ◆ Najvažniji mrežni parametri koji utječu na UVO:
 - Propusnost
 - Kašnjenje
 - Kolebanje kašnjenja
 - Gubitak paketa

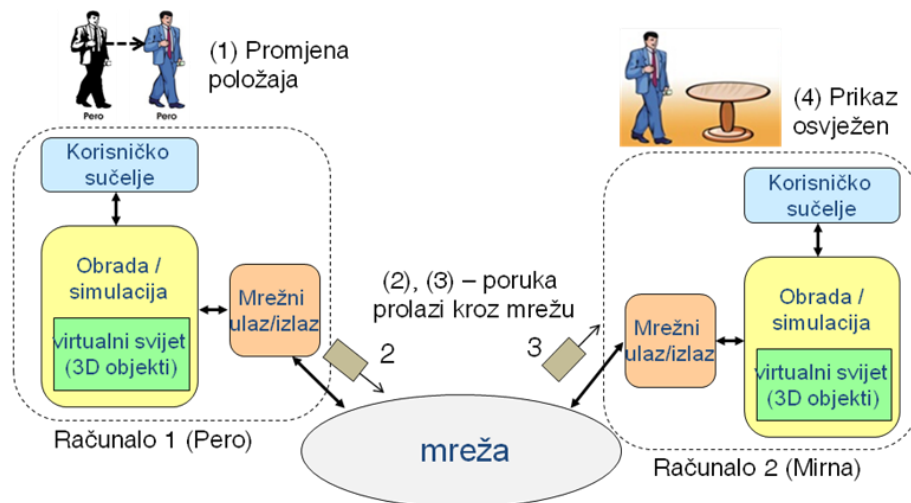
- ◆ Skalabilnost – koliko neki sustav može rasti, a da se pri tom ne naruši njegova funkcija
- ◆ UVO-a mogu biti vrlo popularna – mehanizmi za osiguranje skalabilnost su vrlo važni
 - Second Life – na vrhuncu popularnosti imao 1 milijun korisnika
 - World of Warcraft – na vrhuncu popularnosti 12 milijuna korisnika
- ◆ Parametri koji utječu na skalabilnost:
 - Složenost grafičkog prikaza svijeta i avatara
 - Količina podataka koju generira svijet
 - Obrada ulaza i izlaza
 - Odražavanje dinamičkog zajedničkog stanja

Osnovni model UVO

- ◆ Korisničko sučelje, ulazni i izlazni uređaji
- ◆ Skup 3D geometrijskih objekata (oblik, boja i sl.), i drugih komponenata (npr. svjetlo, zvuk) u računalu
- ◆ Obrada – sadrži simulaciju vezana uz primjenu (početne postavke, simulacijska petlja), obrada 3D grafike i zvuka
- ◆ Mrežna povezanost



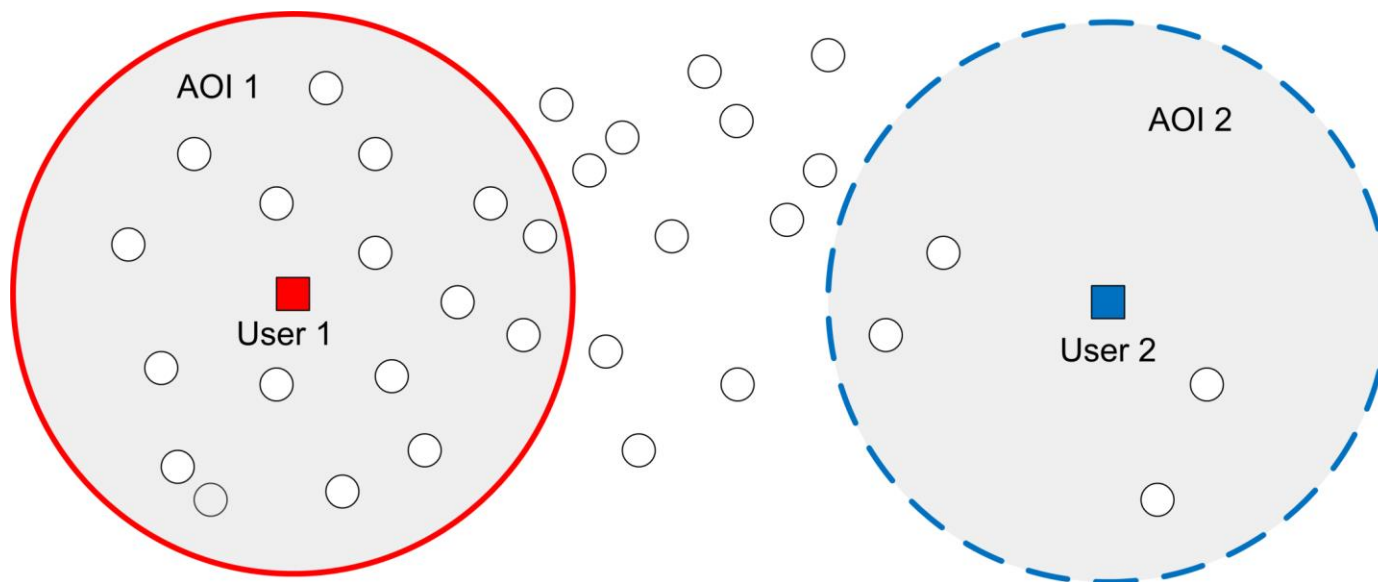
Usklađivanje instanci UVO putem mreže



- ◆ Svaki korisnik ima pokrenutu instancu VO
- ◆ Ne moramo slati elemente koji se ne mijenjaju (npr. teksture, gotovi 3D objekti)
- ◆ Problemi
 - Promet raste s brojem korisnika – skalabilnost
 - Kašnjenje nije isto za sve korisnike - nekonzistentnost

Filtriranje prema području interesa

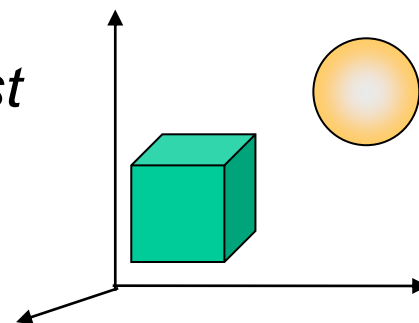
- ♦ engl. Area Of Interest Management, AOIM
- ♦ Prosljeđuju se samo relevantne poruke
- ♦ Umjesto eksponencijalnog rasta prometa s brojem korisnika, linearan rast: ključan preduvjet za postojanje modernih UVO s desecima tisuća istovremenih korisnika



Upravljanje dinamičkim zajedničkim stanjem

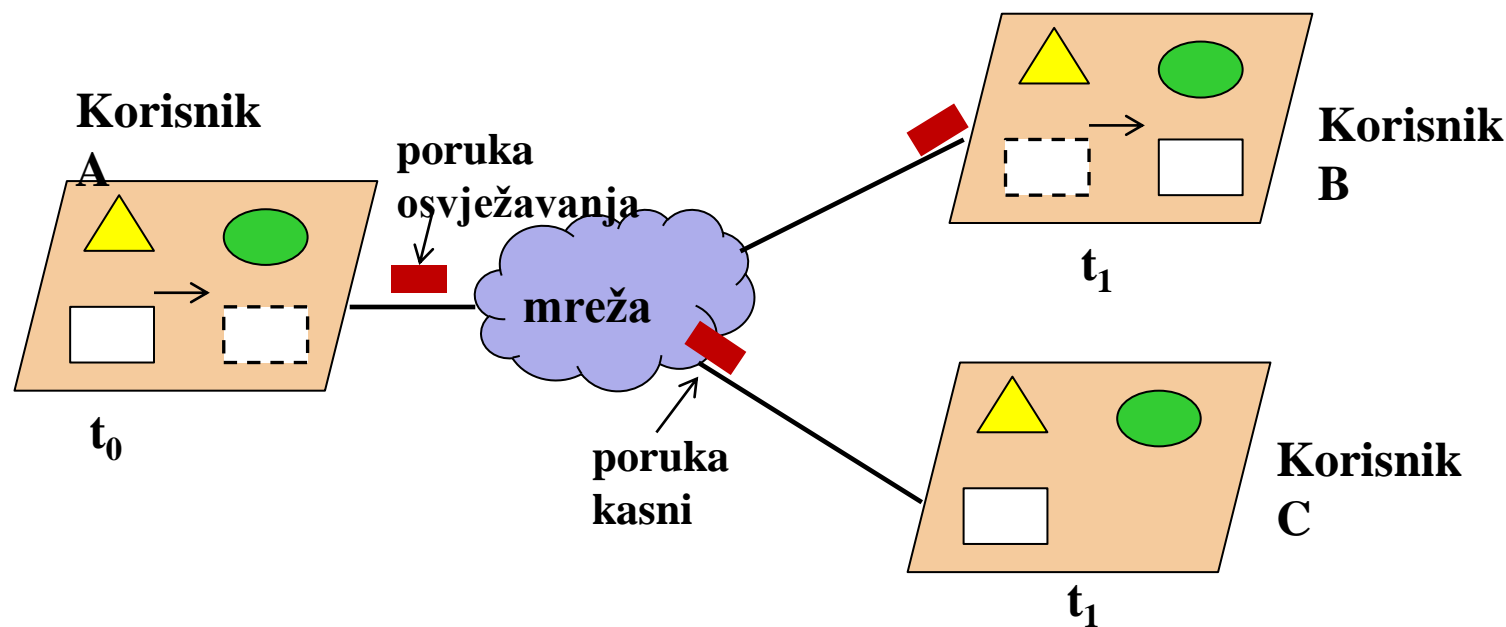
Upravljanje dinamičkim zajedničkim stanjem

- ♦ Zahtjev: održati konzistentno stanje (isto stanje na svim replikama virtualnog svijeta)
- ♦ Zajedničko stanje = skup varijabli stanja svih pojedinačnih objekata u UVO
 - varijable za jedan objekt: položaj, orijentacija, brzina, izgled, itd.
- ♦ Osnovni problem:
odnos *konzistentnost* – *propusnost*



Problem nekonzistentnosti

Zavod za komunikacije



- ♦ Ključni izazov: održavanje raspodijeljenog zajedničkog stanja konzistentnim, ili približno konzistentnim, uz promjene koje nastaju (asinkrono, dinamički i raspodijeljeno)
- ♦ Zahtjeve na konzistentnost treba uskladiti s namjenom aplikacije
- ♦ Češće slanje poruka osvježavanja skraćuje potencijalni vremenski interval u kojem može doći do odstupanja, ali više opterećuje mrežu i proces za slanje i primanje poruka

Rješenja problema upravljanja zajedničkim stanjem

- ◆ Na razini arhitekture
 - Centralizirani repozitorij informacija
 - Raspodijeljeni (distribuirani) repozitorij informacija
- ◆ Na razini protokola
 - Jaka konzistentnost - svakom trenutku stanje svake replike UVO-a je istovjetno
 - Koristi se protokol koji osigurava konzistentnost u distribuiranom okruženju
 - Protokol konzistentnosti je na aplikacijskoj razini
 - Slaba konzistentnost – stanje svake replike UVO-a može biti različito u istom trenutku
 - Razlika se pokušava minimizirati kako bi njen utjecaj bio neprimjetan
 - Koristi se učestala regeneracija (slanje) stanja
 - Koriste se tehnike predviđanja i *dead reckoning*



Arhitektura raspodijeljene aplikacije

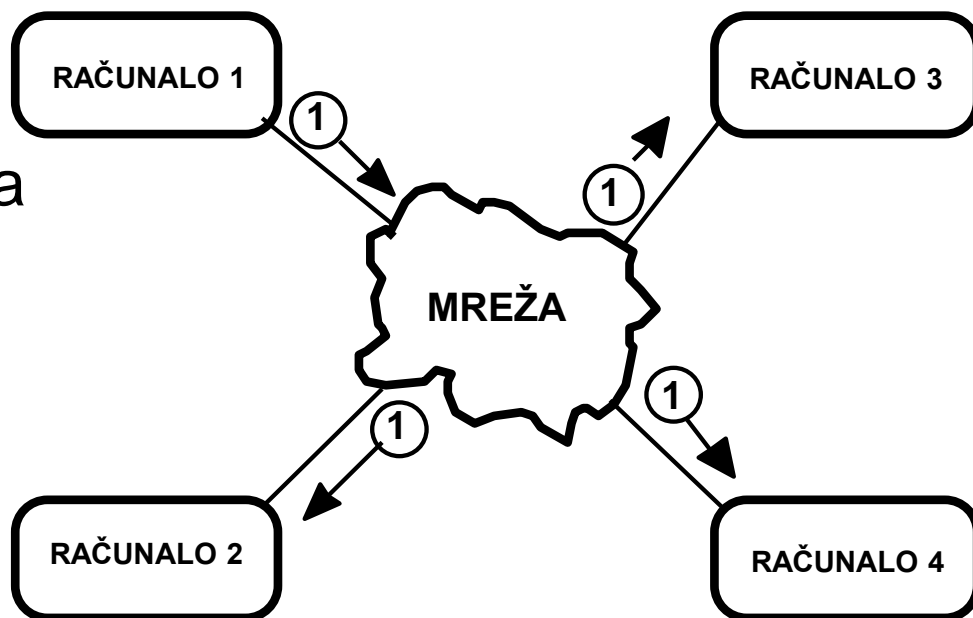
UVO

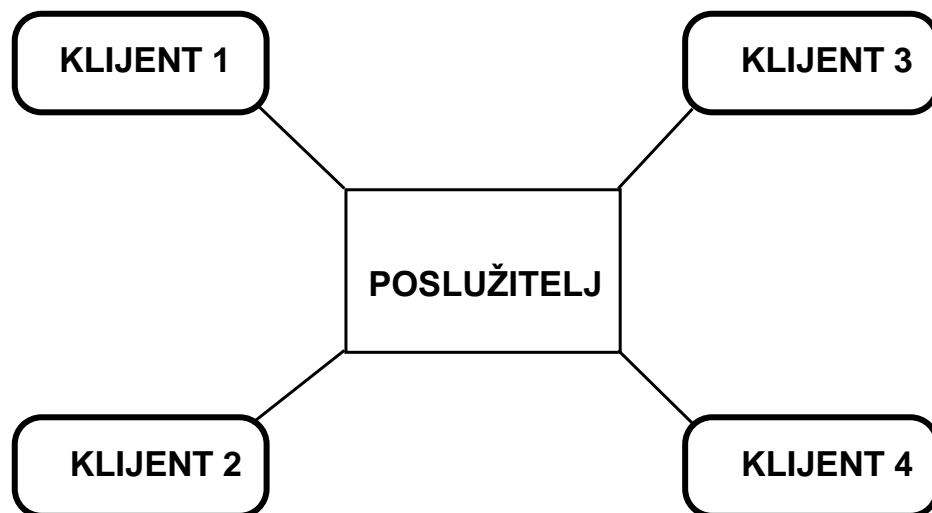
♦ Zahtjevi:

- Učinkovit prijenos poruka
- Podrška za AOIM
- Upravljanje sjednicom
- Trajni zapis stanja
- Kontrola pristupa, naplata

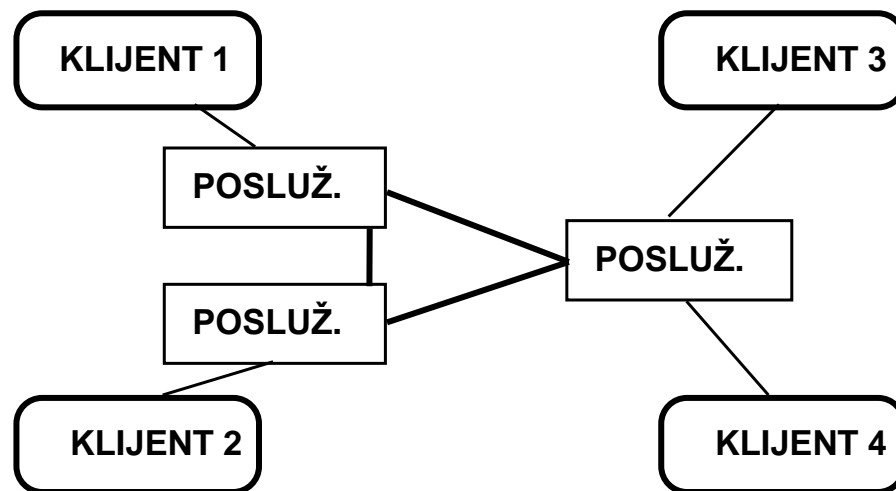
♦ Rješenja:

- Klijent/poslužitelj
- Više poslužitelja
- Ravnopravni procesi (engl. peer-to-peer)



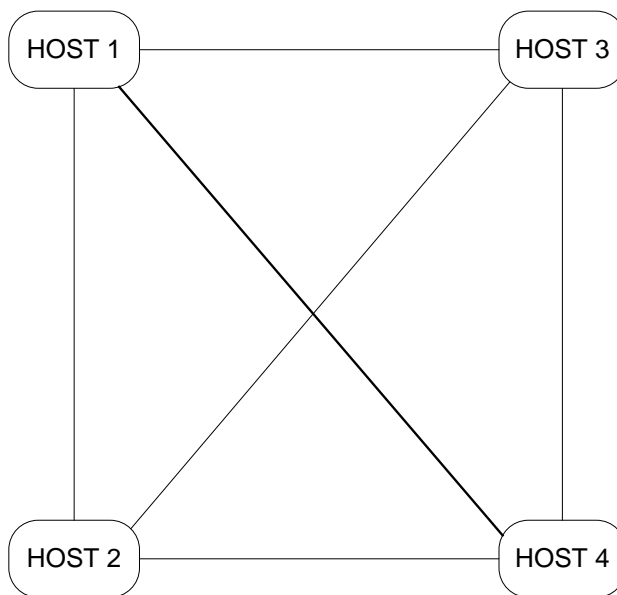


- ♦ Vrlo učinkovita arhitektura
- ♦ Filtriranje prometa, kontrola sjednice, trajni zapis, kontrola pristupa i naplata izvedeni na poslužitelju
- ♦ Poslužitelj usko grlo: ograničena prilagodljivost veličini



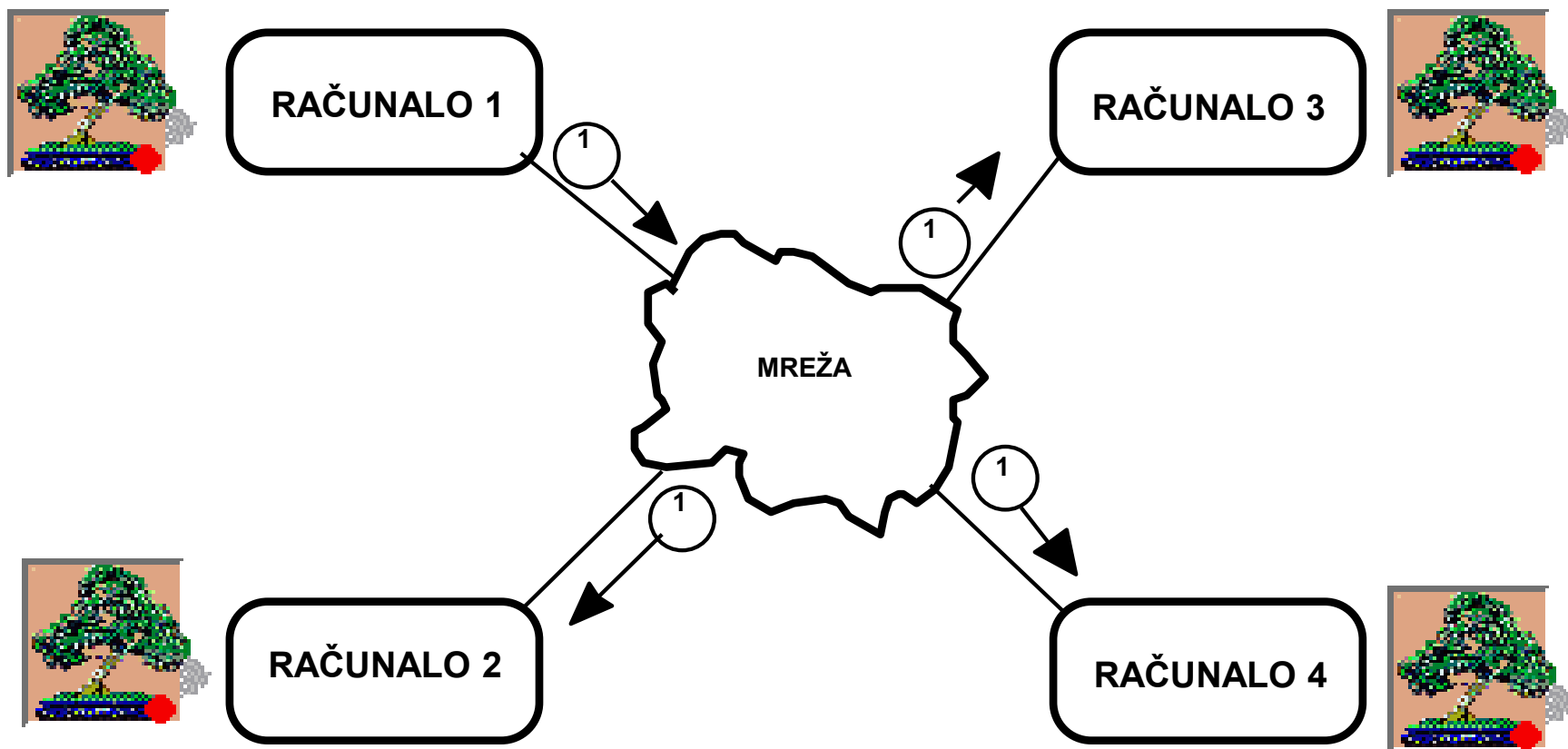
- ♦ Većina dobrih strana arhitekture klijent/poslužitelj
- ♦ Povećava se kašnjenje
 - potreba za vrlo brzom vezom između poslužitelja
- ♦ Izvedba relativno složena

Ravnopravni procesi (peer-to-peer)

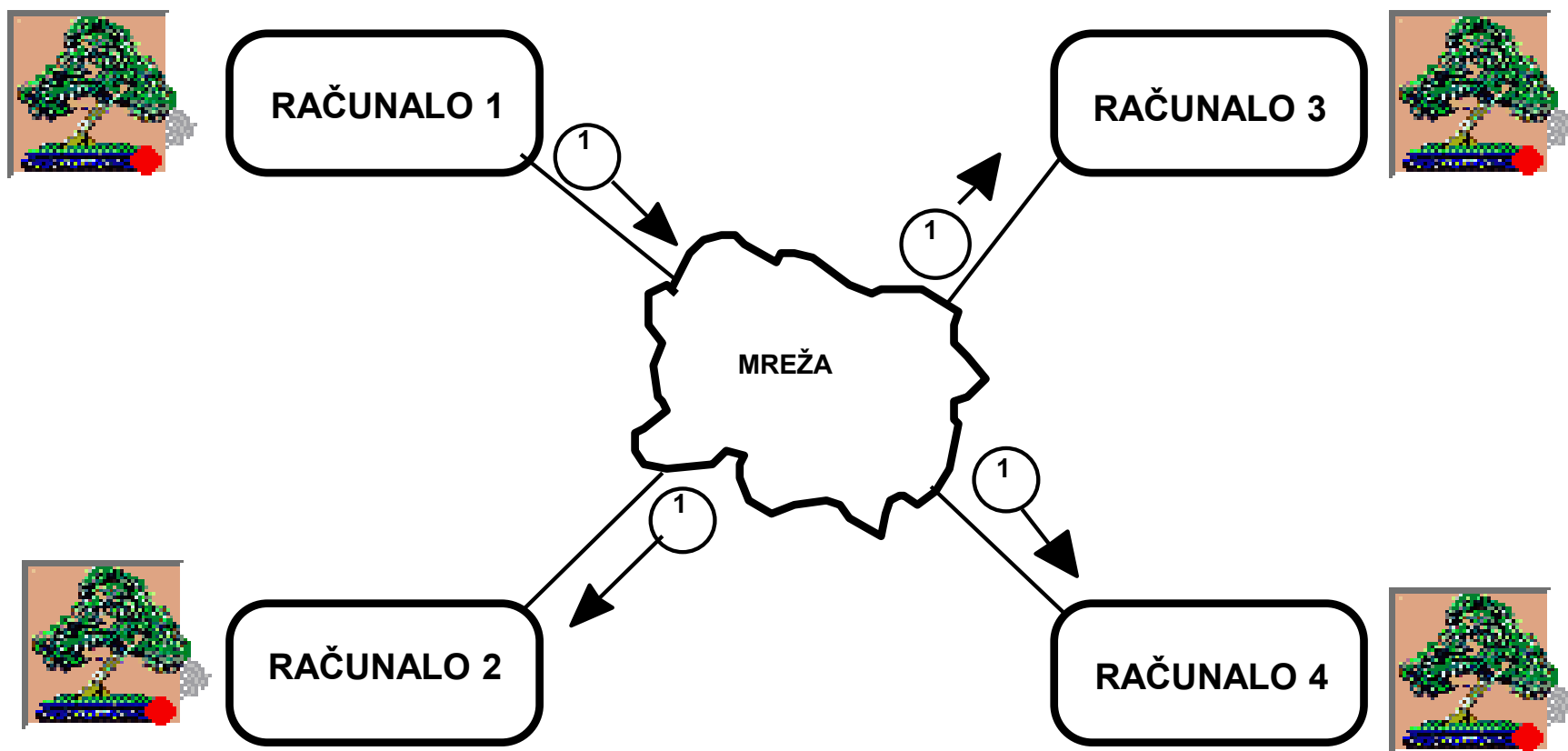


- ◆ Direktna komunikacija između svih čvorova
- ◆ Prvi sustavi UVO (npr. Doom) koristili su ovo rješenje
- ◆ Problemi: trajni zapis, kontrola pristupa, AOIM, upravljanje sjednicom, kontrola varanja, migriranje informacija u slučaju odspajanja jednog korisnika

Jaka konzistentnost



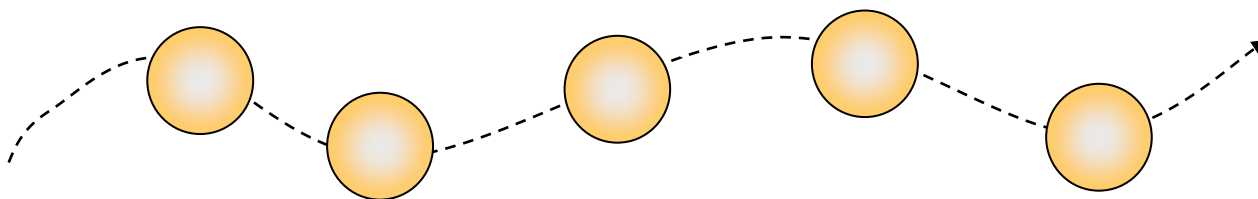
Slaba konzistenost



- ◆ Jaka konzistentnost (zajednički podaci)
 - Ograničena kašnjenjem najsporijeg klijenta – informacija o promjeni mora doći i do tog klijenta
 - Ograničena nivoom interaktivnosti samog UVO-a
 - Primjerice Time To Kill nekih oružja u Call of Duty franšizi je 200 ms - nemoguće omogućiti takvu interaktivnost u sustavu s jakom konzistentnosti
- ◆ Slaba konzistentnost (replicirani podaci)
 - Koristi učestala osvježanja – ograničena propusnošću
 - Ograničena arhitekturom (skalabilnošću) – kako primiti i poslati osvježanja za veliki broj korisnika u slučaju P2P arhitekture?

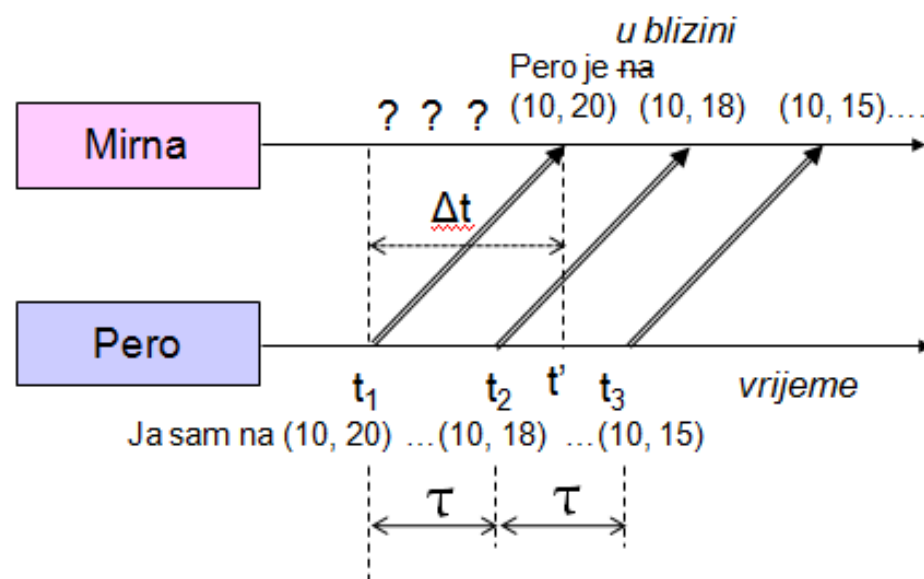
Učestala regeneracija stanja

- ◆ Vlasnik entiteta razašilje poruke osvježavanja u pravilnim vremenskim razmacima, koristeći za dostavu nepouzdan protokol



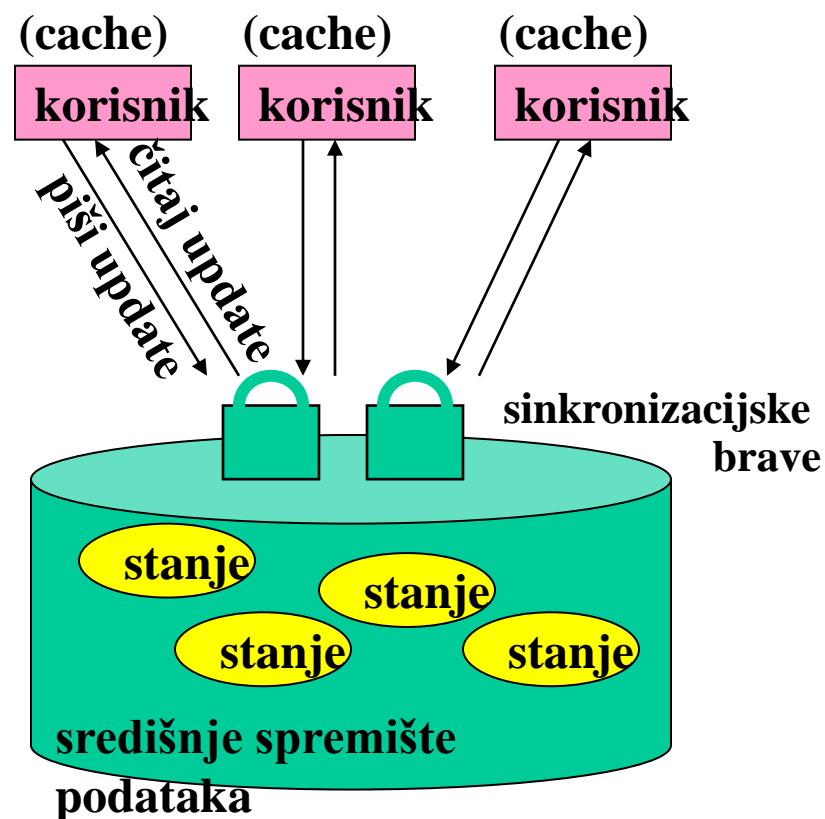
- ◆ Svaka poruka osvježavanja sadrži puni opis stanja entiteta (npr. položaj, orijentaciju, izgled itd.)
- ◆ Svaki primatelj ima lokalni spremnik
- ◆ Primatelji ne potvrđuju primitak poruka, a izgubljene poruke osvježavanja se nadomještaju novijima [primjer 1]
- ◆ Nema očuvanja globalnog redoslijeda poruka osvježavanja

- ◆ Primjer pokazuje nekonzistentnost uzrokovanu kašnjenjem
- ◆ Za maksimalnu konzistentnost, Pero bi trebao čekati da Mirna potvrdi primitak, pa tek onda slati iduću poruku: gubitak propusnosti i interaktivnosti
- ◆ Nemoguće je postići visoku dinamičnost u okruženju i visoku konzistenciju



Središnji repozitorij informacija s zajedničkim podacima

- ◆ Klijent – poslužitelj uz jaku konzistenost
- ◆ Središnji repozitorij sadrži sve podatke o zajedničkom stanju UVO
- ◆ Svi čvorovi u svakom trenutku imaju identični pogled na zajedničko stanje
- ◆ Središnji repozitorij nadzire čitanje i pisanje stanja, kao i poredak pri osvježavanju stanja



Prednosti i nedostaci središnjeg repozitorija s zajedničkim podacima



Zavod za telekomunikacije

♦ Prednosti

- Jednostavan programski model
- Garantirana konzistentnost
- Nema vlasništva nad podacima, središnji poslužitelj se brine o osvježavanju stanja i redoslijedu

♦ Nedostaci:

- Nepredvidivost što se tiče trajanja i čekanja na pristup podacima i osvježavanje
- Zamjetna količina dodatne obrade (*overhead*) – pouzdani komunikacijski protokol i/ili učestalo slanje stanja od strane poslužitelja”

Raspodijeljeni repozitorij s zajedničkim podacima



Zavod za telekomunikacije

- ◆ Peer to peer uz jaku konzistentnost
- ◆ Očuvanje konzistentnosti se sada prebacuje na protokol za očuvanje
- ◆ Sve replike su uvijek u istom stanju te imaju kompletnu informaciju o cijelom virtualnom svijetu
- ◆ Primjer korištenja: strategije u stvarnom vremenu (engl. Real Time Strategies) poput Starcraft 2
- ◆ Prednosti: bolja otpornost sustava, nema kašnjenja uzrokovano komunikacijom sa središnjim sustavom, omogućuje dobru skalabilnost s obzirom na mrežni promet (deterministička simulacija)
- ◆ Mane: najsporiji klijent usporava sustav, ne može se koristiti za sustave s visokom razinom interakcije

Središnji repozitorij s raspodijeljenim podacima



Zavod za telekomunikacije

- ◆ Klijent – poslužitelj sa slabom konzistentnošću
- ◆ Svaki klijent posjeduje repliku stanja UVO-a (najčešće samo lokalnu repliku određenu područjem interesa)
- ◆ Poslužitelj posjeduje centralnu kopiju koja ima „stvarno” stanje prema kojem se osvježavaju replike na klijentima
- ◆ Poslužitelj sljedeće dvije funkcije:
 - Osigurava jednoznačni poredak poruka osvježavanja stanja
 - Razašilje novo stanje svim zainteresiranim klijentima
- ◆ Najčešća varijanta u današnjim sustavima

Raspodijeljeni repozitorij s raspodijeljenim podacima



Zavod za telekomunikacije

- ◆ Peer to peer s slabom konzistentnošću
- ◆ Ne postoji kompletna „točna” replika
- ◆ Moguća i parcijalna replikacija virtualnog svijeta – kombinirani podaci sa svih replika daju ukupnu sliku svijeta
- ◆ Za neke primjene dodatna obrada kao kod središnjeg repozitorija može biti neprihvatljiva, ili pak ne trebaju apsolutnu konzistentnost
 - primjer: simulator leta – “glatki” pomak, odnosno prikaz promjene položaja bez trzaja ili zamrzavanja slike, je važniji od apsolutne točnosti položaja u svakom trenutku

Primjer 1: učestalost osvježavanja

Zavod za telekomunikacije

Pretpostavimo da izvor šalje 25 poruka osvježavanja svake sekunde. Na temelju toga, prikaz se može osvježiti svakih:

$$1/25 = 0.04 \text{ s} = 40 \text{ ms}$$

Jedna izgubljena poruka utječe na prikaz u trajanju od 40 ms, što za najveći broj primjena, ljudima nije uočljivo.

Dizajner aplikacije može pretpostaviti da će visoka učestalost poruka osvježavanja učiniti male nekonzistencije nevidljivima za korisnike.

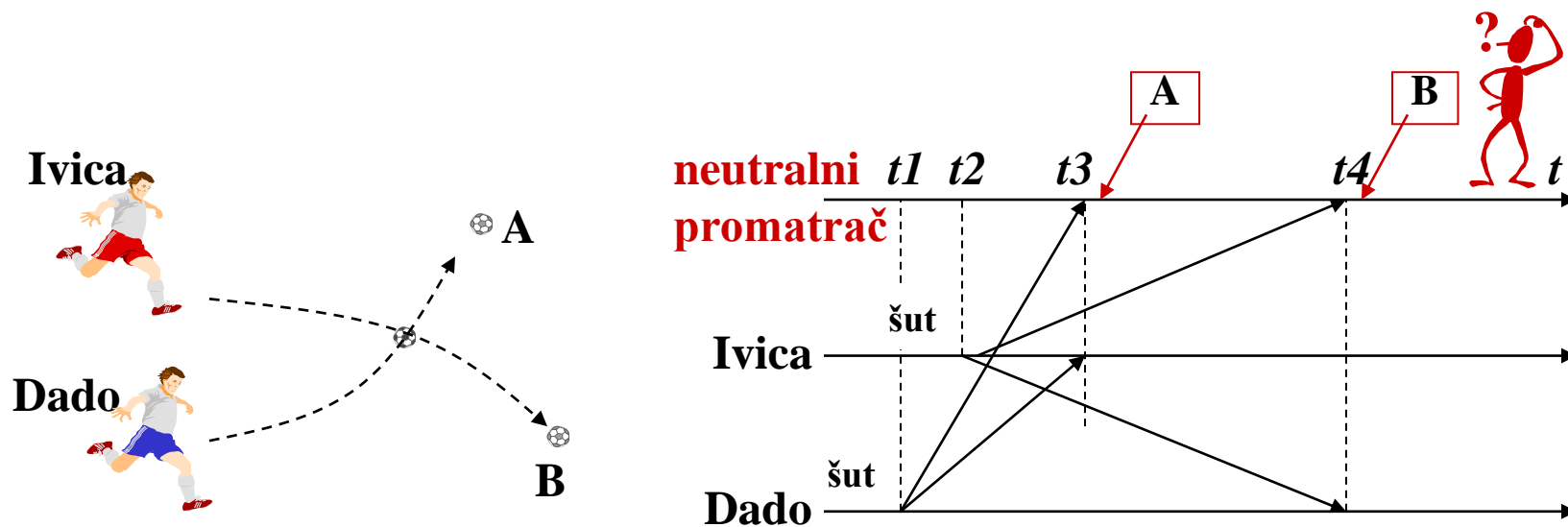
Primjeri aplikacija koje koriste učestalo osvježavanje:

- Silicon Graphics Dogfight simulator leta
- Višekorisničke igre na PC-u: Overwatch, Battlefield, World of Warcraft

- ◆ Kada nema potpune konzistentnosti dolazi do problema vlasništva nad objektima
- ◆ Treba spriječiti da više igrača istovremeno mijenja stanje entiteta [primjer 2, primjer 3]
 - Na primjer, promjena položaja je pisanje varijable stanja položaja entiteta (x, y, z)
- ◆ Uvodi se eksplicitno vlasništvo nad entitetom
 - Tipičan primjer je korisnikov *avatar*, čiji je vlasnik korisnik koji njime upravlja
 - Drugim objektima upravlja poslužitelj *lock manager*
- ◆ *Lock manager* osigurava da svaki entitet u zadanom trenutku ima samo jednog vlasnika

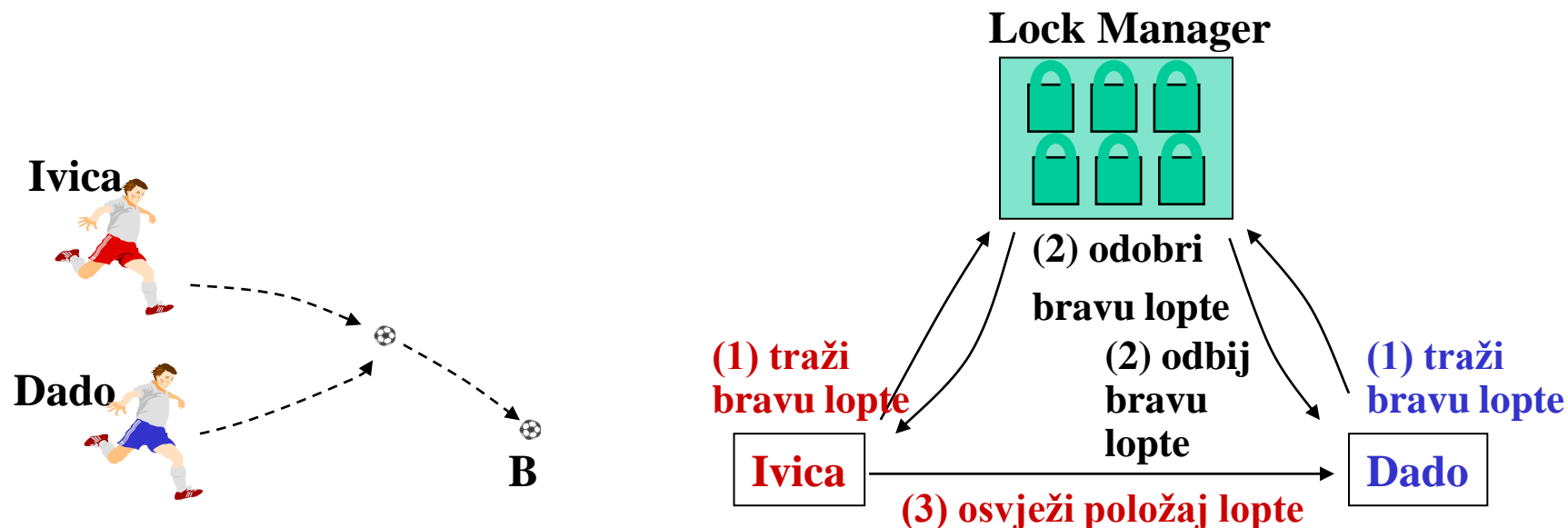
Primjer 2: nema vlasništva

Ivica i Dado igraju nogomet i pokušavaju istovremeni šutnuti loptu. Svaki host šalje poruku osvježavanja stanja lopte (položaj). Neki promatrači mogu prvo primiti Ivičinu poruku, a drugi Dadinu. Osim ako se Ivica i Dado slože oko toga tko je zapravo šutnuo loptu, svaki će nastavljati osvježavati položaj neovisno o drugome. Promatrači će vidjeti titranje lopte između novih položaja ovisno o pristizanju poruka osvježavanja od Ivice i Dade.



Primjer 3: eksplicitno vlasništvo

Ivica i Dado izdaju zahtjev za vlasništvom poslužitelju *lock manager* prije osvježavanja stanja lopte. Ovisno o politici upravljanja bravama (npr. FIFO, round-robin, prioriteti, itd.), *lock manager* daje bravu jednom od korisnika, u ovom primjeru, Ivici. Ivica onda može mijenjati položaj lopte dok god je u vlasništvu brave.



Prednosti i nedostaci distribuiranih pristupa s učestalom regeneracijom stanja



Zavod za telekomunikacije

◆ Prednosti

- Pristup jednostavan za izvedbu: nema centralnog poslužitelja, protokola konzistencije, niti (u nekim slučajevima) brava
- Može se podržati velik broj korisnika

◆ Nedostaci:

- Značajni zahtjevi na propusnost [primjer 4]
- Različita učestalost osvježavanja za različite entitete
- Posljedice mrežnog kašnjenja i kolebanja kašnjenja
 - problemi s kauzalnošću, odnosno, uzročno-posljedični odnosima u virtualnom svijetu [primjer 5]
 - povremeno “smrzavanje” slike, neujednačenost pokreta

Primjer 4: zahtjevi na propusnost

Zavod za telekomunikacije

Pretpostavimo da izvor šalje 30 poruka osvježavanja u sekundi i da je veličina poruke (PDU) 144 byte. (To je inače stvarna veličina PDU za potpuno artikulirani ljudski lik u simulaciji prema standardu Distributed Interactive Simulation).

Za svaku PDU imamo:

$$144 \text{ byte} * 8 \text{ bit/byte} * 30 \text{ 1/s} = 34\,560 \text{ bit/s}$$

Dakle, na 10 Mbit/s LAN (i zanemarujući *overhead*, koji inače nije beznačajan!), možemo imati najviše 289 virtualnih ljudi.

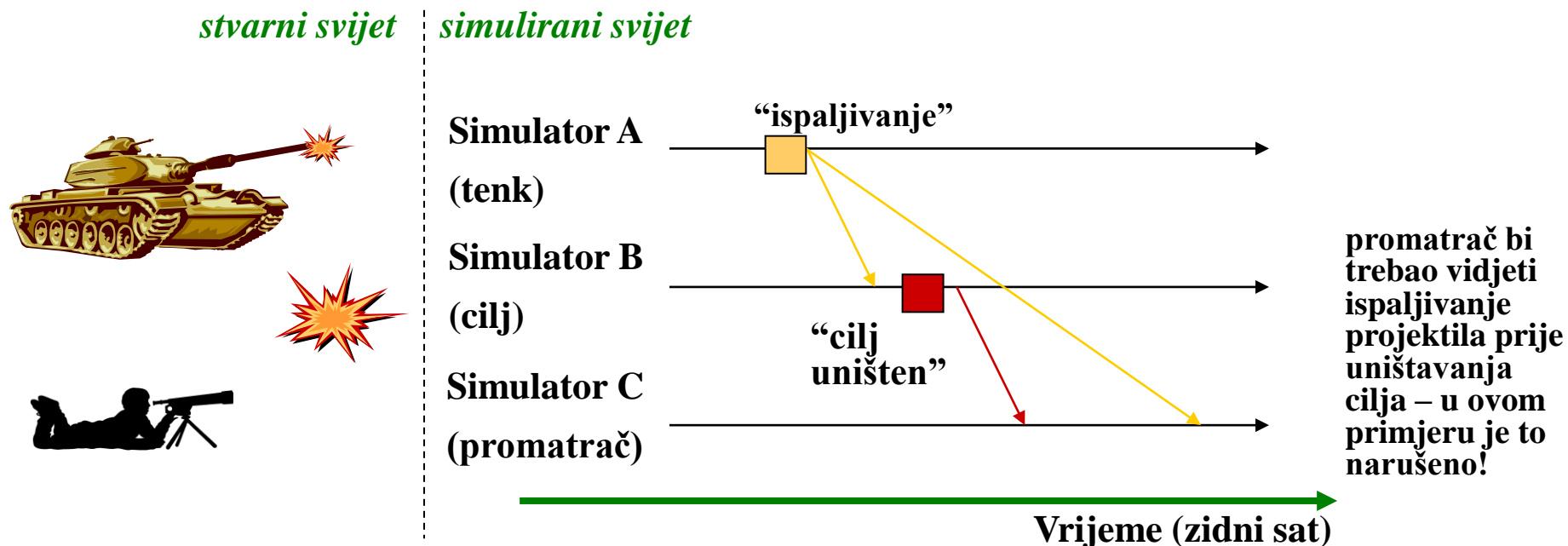
Ako koristimo modemsku vezu s 56 kbps, možemo imati samo 1 virtualni lik.

Da bi se moglo podržati više korisnika, koristi se smanjena učestalost poruka osvježavanja, kao i kompresija poruka. Vidljivo je da se uz osvježavanje “naslijepo” ne mogu podržati UVO s velikim brojem entiteta.

Primjer 5: problem s kauzalnošću

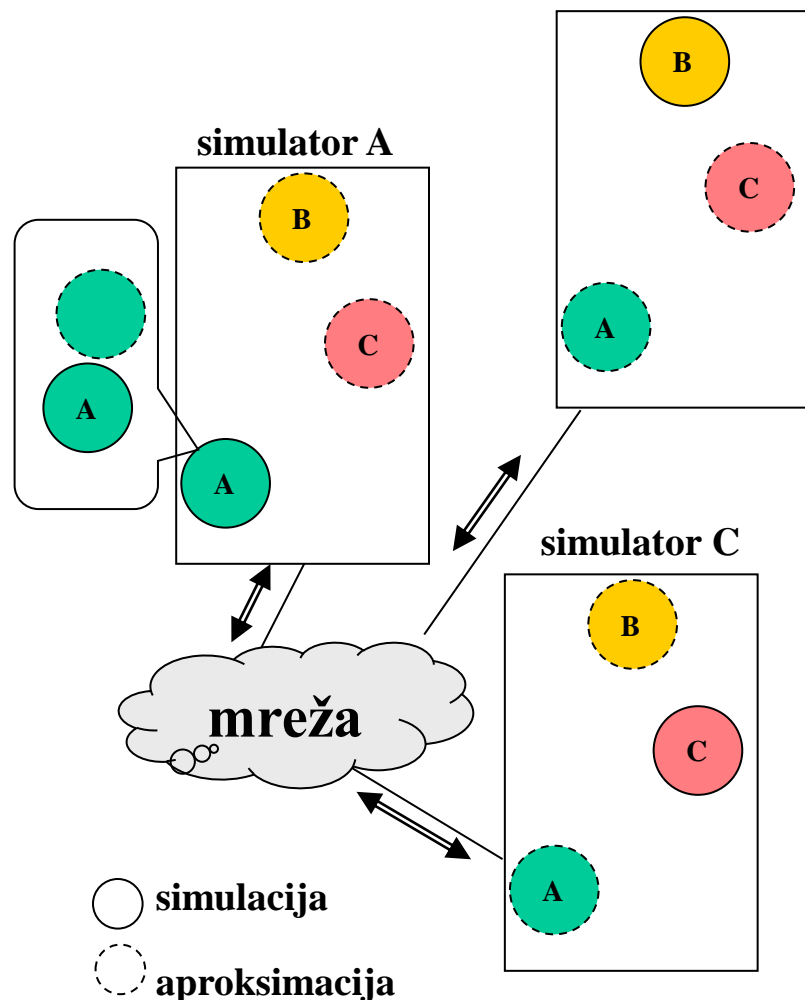
U UVO, kao i u stvarnom svijetu, korisnici očekuju da vrijedi kauzalnost, odnosno, uzročno-posljedični odnosi.

U UVO se taj odnos ne može baš uvijek očuvati zbog nepredvidivog kašnjenja u mreži.



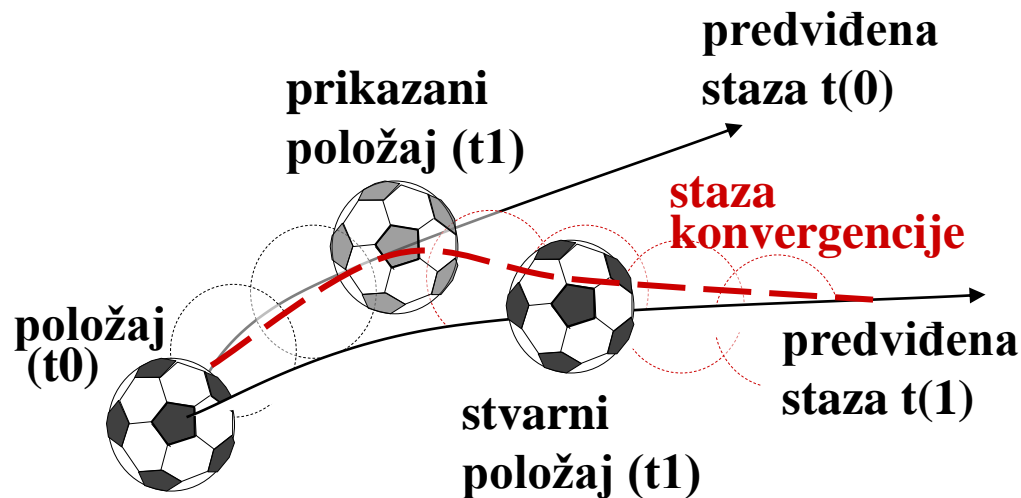
Mrtva procjena (Dead reckoning) 1/2

- ♦ Na strani svakog čvora koristi se i simulacija i aproksimacija trenutnog stanja
- ♦ Aproksimacija je jednostavnija za računanje
- ♦ Svaki čvor za vlastite entitete šalje poruke osvježavanja samo ako razlika između simulacije i aproksimacije prijeđe zadani prag
- ♦ Svaki čvor radi predikciju (predviđanje) stanja udaljenih entiteta, npr. položaja i orijentacije, na temelju lokalno pohranjene informacije



- ◆ Pristup je pogodan za UVO s velikim brojem sudionika
- ◆ Algoritam mrtve procjene sastoji se od predikcije i konvergencije
- ◆ Uvjet za primjenu metode je (barem djelomična) predvidivost kretanja entiteta kako bi se mogla primijeniti predikcija

- ♦ **Predikcija:**
izračunavanje
sadašnjeg stanja na
temelju prethodno
primljenih poruka
osvježavanja
- ♦ **Konvergencija:**
korekcija staze
dobivene predikcijom
na temelju novo-
primljenih poruka
osvježavanja
("izgladivanje")



Prednosti i nedostaci primjene mrtve procjene



Zavod za telekomunikacije

♦ Prednosti:

- Smanjen promet i zahtjevi na propusnost
- Svaki čvor računa algoritam mrtve procjene neovisno o drugima
- Moguće je podržati velik broj korisnika
- Otpornost na gubitke paketa

♦ Nedostaci:

- Slaba konzistentnost – nema garancije da će svi čvorovi imati identično stanje istog entiteta
- Distribuirane simulacije su složene za izvedbu, održavanje i vrednovanje

Oblikovanje programskog rješenja UVO

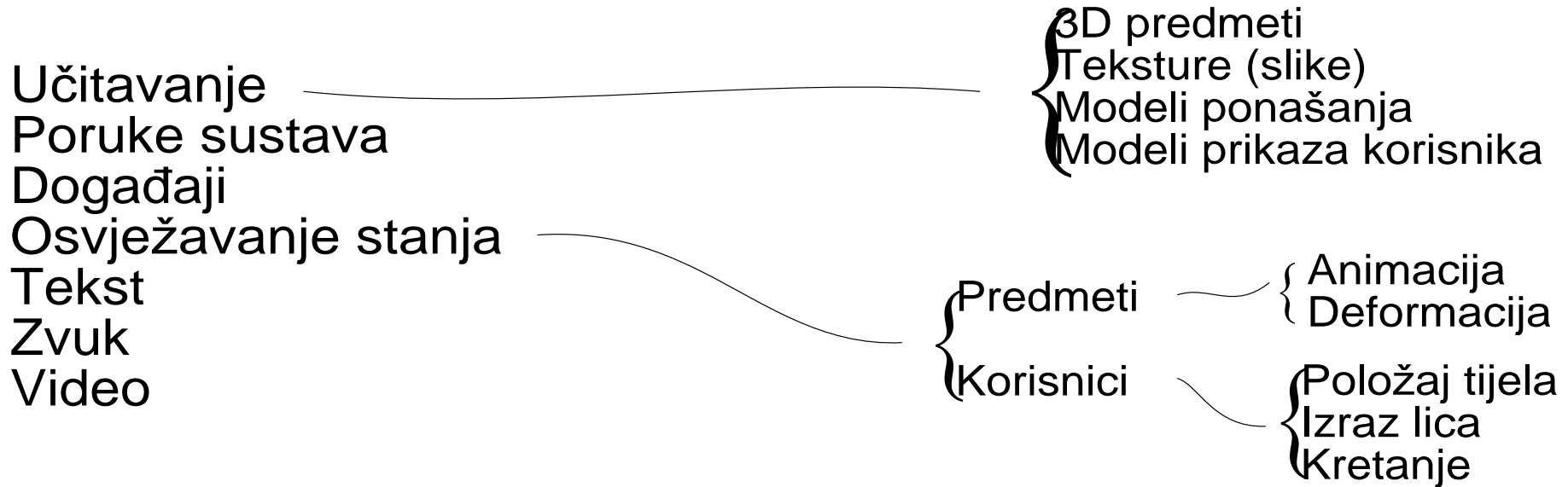
- ◆ Strukturiranje virtualnog prostora
- ◆ Arhitektura raspodijeljene aplikacije UVO
- ◆ Vrste mrežnog prometa i komunikacijski protokoli
- ◆ Način distribucije poruka
- ◆ Izvedba simulacijske petlje UVO

Strukturiranje virtualnog prostora

Zavod za telekomunikacije

- ◆ Podijeliti okruženje na jedinice prihvatljive po:
 - broju korisnika u svakoj jedinici
 - složenosti geometrije svake jedinice
 - preciznosti koordinata
- ◆ Primjer: World of Warcraft
 - Korisnici podijeljeni na veći broj poslužitelja, po nekoliko tisuća korisnika po poslužitelju (engl. žargon *shard*)
 - Nema interakcije s korisnicima na drugom poslužitelju
- ◆ Primjer: Second Life
 - Virtualni svijet podijeljen u ćelije 256 x 256 m
 - Za svaku ćeliju odgovoran zaseban poslužitelj

Vrste mrežnog prometa u UVO



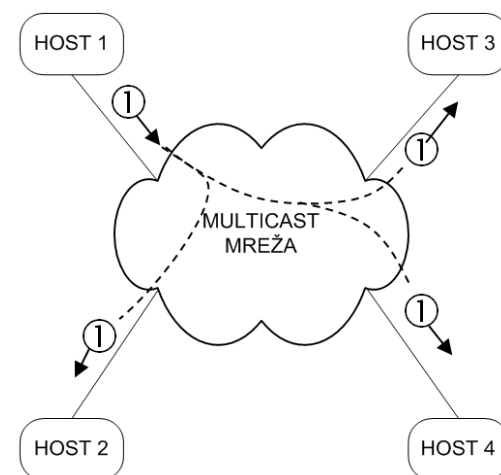
Mrežni zahtjevi i protokoli za pojedine tipove podataka



Zavod za telekomunikacije

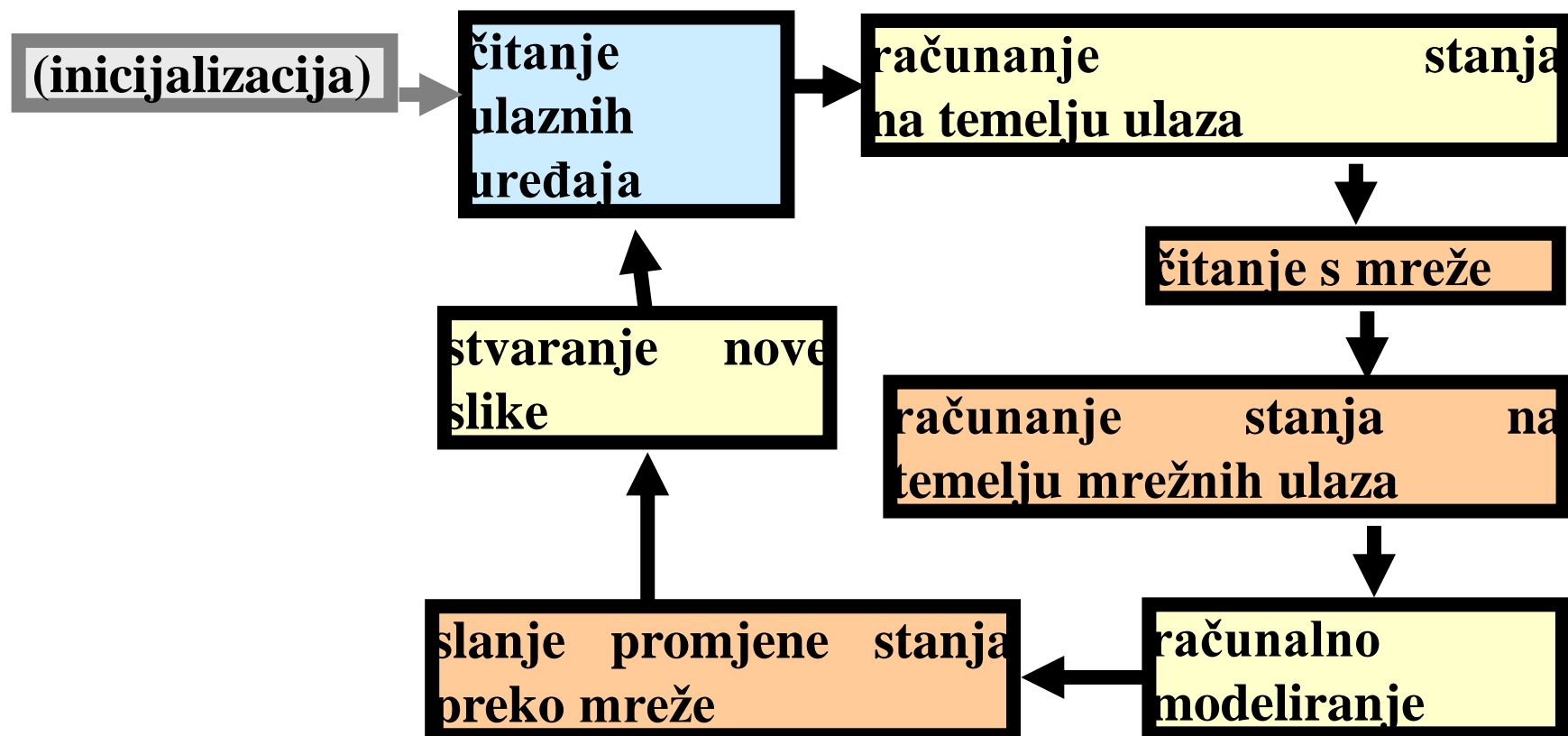
Vrsta prometa	Količina podataka	Potrebna pouzdanost	Protokol
Učitavanje			TCP
Poruke sustava			TCP
Događaji			TCP
Osvježavanje stanja			UDP
Tekst			TCP
Zvuk			UDP (RTP)
Video			UDP (RTP)

- ◆ Jednoodredišna komunikacija
 - Najčešći način povezivanja
 - Neučinkovitost se u praksi kompenzira izvedbom AOIM na poslužitelju
- ◆ Višeodredišno razašiljanje
 - Učinkovitije, ali složenija izvedba

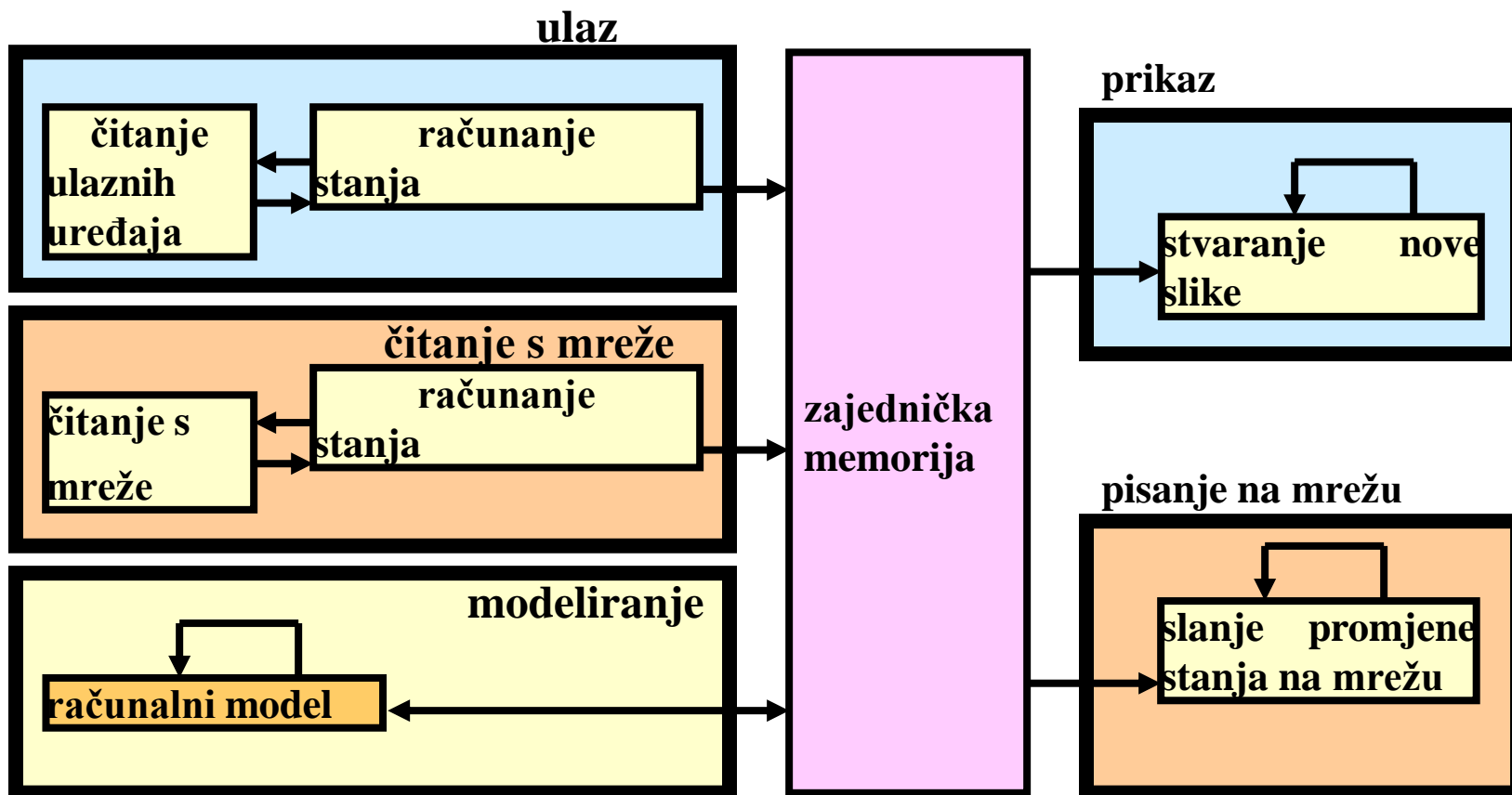


- ◆ Zahtjev:
 - Omogućiti nesmetano i efikasno izvođenje većeg broja modula
- ◆ Rješenja:
 - ~~■ Jednolitno programsko rješenje~~
 - Višenitno programsko rješenje

Jednonitno programsko rješenje



Višenitno programsko rješenje

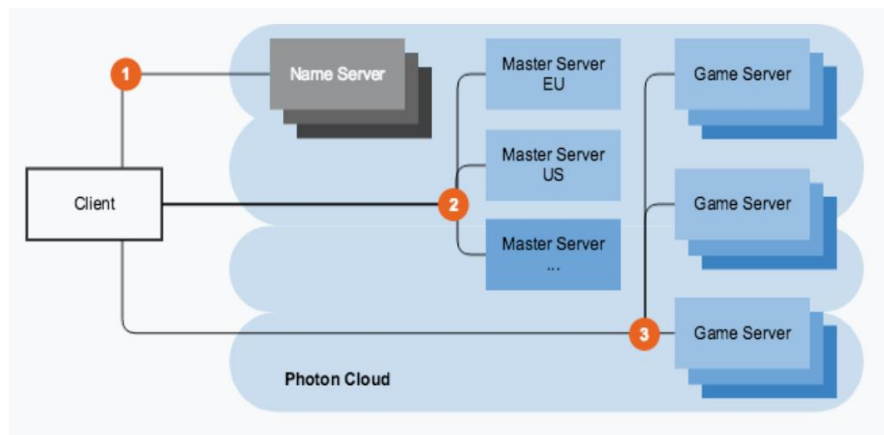


Izrada višekorisničke aplikacije u Unity sustavu za izradu igara



Zavod za telekomunikacije

- ◆ Trenutno unutar Unity postoji sustav pod nazivom Unet koji je zastario te je u razvoju novi sustav
- ◆ Jednostavno se može zamijeniti postojećim bibliotekama
 - Photon Unity Networking (PUN)
 - Mirror Networking (otvorenog koda)
- ◆ PUN omogućuje
 - Sinkronizaciju objekata kroz dodavanje komponente PhotonView objektu koji se sinkronizira
 - Poziv udaljenih procedura
 - Podizanje događaja
- ◆ Detaljnije u okviru zadaće



Virtualni svjetovi

- ◆ engl. Virtual Worlds



Virtualni svjetovi - primjeri



- ◆ Virtualnim novcem kupuju se dobra i usluge
- ◆ Primjeri usluga: kampiranje, rad u trgovini, zabava (često “za odrasle”), izrada sadržaja
- ◆ Primjeri dobara: zgrade, vozila, nakit, odjeća, animacije, biljke, životinje, umjetnine
- ◆ Virtualni novac se kupuje i prodaje
 - Primjer: SL LindeX: 12.12.2018. tečaj 253,90 L\$ za 1\$
 - Pomoću dane valute trgovci unutar Second Lifea mogu zarađivati kreirajući sadržaje (odjeću, obuću, kuće itd.)

- ◆ Pored ranije opisanih osnovnih svojstava, karakteristike specifične za virtualne svjetove:
- ◆ Zajednički, perzistentni svijet za sve korisnike
- ◆ Korisnici stvaraju svijet!
- ◆ Goleme količine sadržaja koji se stalno mijenja
 - Korisnici kupuju “zemlju”, grade, trguju, programiraju
- ◆ Kada se korisnik kreće kroz svijet, mora stalno učitavati nove ili osvježene sadržaje

- ◆ Osvježavanje stanja korisnika: slično kao u igrama
- ◆ Daleko veći promet generira učitavanje modela svijeta/predmeta i njihovo osvježavanje prilikom promjena
- ◆ Načini na koje se smanjuje promet:
 - Lokalno spremanje (engl. *cache*) smanjuje promet pri drugoj i idućim posjetama istom prostoru
 - UDP za sve osim kritičnih poruka, npr. autentikacija
 - Kompresija (JPEG, zlib)
 - Primjer: Second Life – koristi sve navedene tehnike

Primjer: Second Life

- ◆ Mjerenja prilikom posjete Veleposlanstvu Švedske
 - Prva posjeta: blokiranje, dugo čekanje na učitavanje predmeta
 - Druga posjeta: normalan rad, prosjek prometa na klijentu 640 kbit/s

Smjer poruka	Tip prometa	Promet (kb), 1. posjeta	Promet (kb), 2. posjeta
Poslužitelj - klijent	Učitavanje predmeta, tekstura...	11563	2792
	Osvježavanje stanja, događaji	91	50
	Poruke sustava i ostalo	276	151
Klijent -poslužitelj	Učitavanje (zahtjevi)	71	14
	Osvježavanje stanja, događaji	101	87
	Poruke sustava i ostalo	112	69

- ◆ Zbog mrežnih i računalnih zahtjeva jedan poslužitelj služi oko 40 klijenata
 - Usporedba s igrama: stotine ili tisuće klijenata po poslužitelju

- ◆ Virtualni svjetovi su vrhunac popularnosti imali u 2000-tima
 - Club Penguin – čak 200 milijuna registriranih računa – ugašen 2017., a nasljednik Club Penguin Island nije uspio te je najavljeno gašenje
 - Second Life – milijun korisnika u 2007 godini – još uvijek radi, ali s vrlo malim brojem korisnika (oko 45 000)
 - Današnja popularnost puno manja, ali još uvijek postoji veliki broj virtualnih svjetova
- ◆ Minecraft – igra ili virtualni svijet?
 - Prosinac 2017 – 74 milijuna aktivnih korisnika
- ◆ Virtualna i proširena stvarnost donose nove oblike virtualnih svjetova
 - Facebook Horizons
 - Holoportation

Lista virtualnih svjetova (nepotpuna)



Zavod za telekomunikacije

- | | | | |
|---------------------|-------------------|----------------------|--------------------|
| ◆ Active Worlds | ◆ Ever, Jane | ◆ Myst Online: Uru | ◆ There.com |
| ◆ AltspaceVR | ◆ Facebook Spaces | ◆ Live | ◆ TheWaveVR |
| ◆ Anyland | ◆ Geekzonia | ◆ NeosVR | ◆ Twinity |
| ◆ Avakin Life | ◆ Guru Gedara | ◆ Neutrans | ◆ Center |
| ◆ Bigscreen | ◆ High Fidelity | ◆ Occupy White Walls | ◆ VirBELA |
| ◆ Ceek | ◆ Hyperfair VR | ◆ Oculus Rooms | ◆ Virtual Paradise |
| ◆ Cisco Spark in VR | ◆ IMVU | ◆ Pluto VR | ◆ Virtual Universe |
| ◆ Community Garden | ◆ Inlight Spark | ◆ Rec Room | ◆ VRChat |
| ◆ Cryptovoxels | ◆ JanusVR | ◆ Rumii | ◆ vTime |
| ◆ DiveReal | ◆ Kiss or Kill | ◆ Sansar | ◆ Worldopoly |
| ◆ Edorable | ◆ Kitely | ◆ Second Life | ◆ Worlds Adrift |
| ◆ EmbodyMe | ◆ LivCloser | ◆ Sinespace | |
| ◆ Endless Riff | ◆ LiveLike | ◆ Somnium Space | |
| ◆ Engage | ◆ Mark Space | ◆ STYLY | |
| ◆ Eventual VR | ◆ Mozilla Hubs | ◆ SurrealVR | |
| | | ◆ Topik | |

- ◆ Većina virtualnih svjetova koristi kombinacije triju modela
 - Oglašavanje, kanal za prodaju
 - Pretplate
 - Često besplatno osnovno članstvo, naplata za “premium” članstvo
 - Prodaja virtualnih dobara
 - Virtualni novac koji se može trošiti unutar svijeta
 - Dodaci za avatare
- ◆ Ograničenja poslovnog modela s oglašavanjem
 - Svjetovi za djecu – mnogi roditelji nisu skloni izlaganju djece reklamama
 - Tvrtke nisu sklone oglašavanju u nekontroliranom okruženju
 - Troškovi: stalna prisutnost npr. u SL zahtijeva zaposlenika

Primjeri poslovnih modela



Zavod za telekomunikacije

Svijet	Vlasnik	Ciljani korisnici	Područje	Poslovni model	Korisnici (2008)	Rast 07-08
BarbieGirls	Mattel	Djevojčice	SAD, svijet	Prodaja igračaka, pretplate	3,3M	26%
ClubPenguin	Disney	Djeca	SAD, svijet	Pretplate	10,8M	23%
CyWorld	SK Telecom	Tinejdžeri, 20te	Koreja	Virtualna dobra	13,7M	-1%
Habbo Hotel	Sulake, FI	Tinejdžeri	Europa, svijet	Oglašavanje, virtualni novac	6,3M	52%
NeoPets	Viacom	Djeca	Svijet	Premium članstvo, oglašavanje, virtualna dobra	6,1M	-16%
ActiveWorlds		Odrasli	SAD, svijet	Pretplate, hosting	0,1M	181%
Lively	Google	Odrasli	Svijet	Oglašavanje	0,6M	Ugašen
IMVU		Odrasli	SAD, svijet	Premium članstvo, oglašavanje	4,8M	22%
Second Life	Linden Labs	Odrasli	Svijet	Pretplate, virtualni novac i dobra, oglašavanje	1,6M	-45%

- ♦ S. Singhal & M. Zyda, *Networked Virtual Environments: Design and Implementation*, Addison-Wesley, 1999 (poglavlja 4, 5)
- ♦ Abdelkhalek, A.; Bilas, A. and Moshovos, A. (2001), Behavior and Performance of Interactive Multiplayer Game Servers, Proc. Int. IEEE Symposium on the Performance Analysis of Systems and Software.
- ♦ Gao Huang, Meng Ye, Long Cheng, “Modeling System Performance in MMORPG”, in IEEE CSG Workshops, 2004, pp. 512-518.
- ♦ Seay, A. F., Jerome, W. J., Lee, K. S., and Kraut, R. E. 2004. Project massive: a study of online gaming communities. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems* (Vienna, Austria, April 24 - 29, 2004). CHI '04. ACM, New York, NY, 1421-1424.
- ♦ Kumar, Chhugani, Kim, Kim, Nguyen, Dubey: “Second Life and the New Generation of Virtual Worlds”, IEEE Computer, 2008
- ♦ D-Lib Magazine, December 2005, Volume 11, Nr. 12 (John Kirriemuir)
- ♦ ESA: Entertainment Software Association, <http://www.theesa.com/>
- ♦ DFC Intelligence, <http://www.dfciint.com>
- ♦ “Telepresence report”, Howard S. Lichtman, Human Productivity Lab
- ♦ Virtual Worlds Management, <http://www.virtualworldsmanagement.com/>
- ♦ JP Morgan Global Equity Research, 2009