

Programiranje mrežnih računalnih igara

Izradio: Mislav Svetina, 3.f

Sadržaj

Uvod.....	2
Osnove mrežnog programiranja za računalne igre	2
Arhitekture mrežnih igara	2
Izazovi i rješenja u programiranju mrežnih igara	3
Primjeri popularnih mrežnih računalnih igara	3
Zaključak.....	4
Literatura.....	4

Uvod

U ovom seminarsku radu obraditi ću složen proces programiranja mrežnih računalnih igara i to kroz osnove programiranja mrežnih igara, arhitekture, izazove koji se javljaju programerima, rješenja koja omogućavaju uzbudljiva i dinamična online iskustva.

Samo programiranje mrežnih računalnih igara je jedan složen proces koji omogućava igračima diljem svijeta da zajedno igraju, natječu se i surađuju unutar virtualnih svjetova. Kroz primjere popularnih igara, prikazat ću kako se teorija primjenjuje u praksi i što čini mrežne igre privlačnima širokom spektru igrača. Računalne igre predstavljaju jedan od najraširenijih oblika zabave.



Slika 1

Osnove mrežnog programiranja za računalne igre

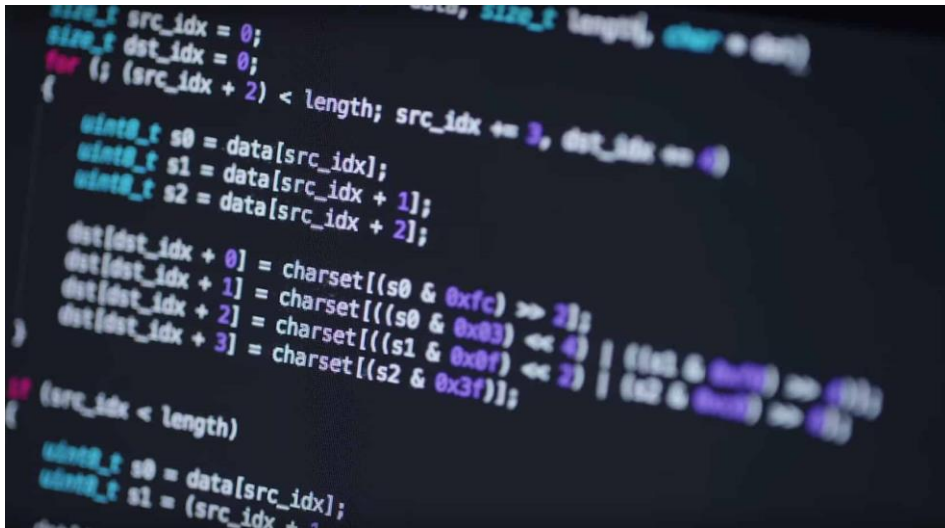
Mrežno programiranje za računalne igre uključuje upotrebu mrežnih protokola i arhitektura za omogućavanje komunikacije između računala. Temelji se na korištenju [TCP/IP](#) i [UDP](#) protokola za razmjenu podataka između klijenata (igrača) i servera. [TCP/IP](#) se često koristi za pouzdanu komunikaciju, poput prijenosa važnih podataka o stanju igre, dok se [UDP](#) koristi za brzu komunikaciju koja može tolerirati gubitak nekih paketa, što je ključno za realno vremenske aspekte igre. Programeri koriste različite mrežne biblioteke i alate, poput [Photon](#), [Unity Networking](#) i [Unreal Engine's Networking](#), kako bi olakšali ovaj proces i osigurali glatku igračku iskustva.

Arhitekture mrežnih igara

Postoje dvije glavne arhitekture korištene u mrežnom programiranju računalnih igara:

1. [klijent-server](#) model koristi centralni server koji obrađuje sve zahtjeve i sinkronizira stanje igre među svim igračima, osiguravajući konzistentnost i smanjujući varanje.

2. [Peer-to-peer \(P2P\)](#) arhitektura omogućuje izravnu komunikaciju između igrača, smanjujući ovisnost o centralnom serveru i potencijalno smanjujući kašnjenja, ali postavlja veće zahtjeve na sigurnost i sinkronizaciju.



Slika 2

Izazovi i rješenja u programiranju mrežnih igara

Samo programiranje mrežnih igara suočeno je sa izazovima kao što su: [kašnjenje \(latencija\)](#), sinkronizacija igre, sigurnost i sl. Da bi smanjili kašnjenje, programeri često koriste tehniku poput predviđanja klijenta. Sinkronizacija stanja igre postiže se konzistentnim ažuriranjima i korištenjem autoritativnog servera za sprječavanje varanja. Sigurnost se poboljšava kroz enkripciju i autentifikaciju, dok se skalabilnost osigurava kroz efikasno upravljanje resursima i upotrebom cloud tehnologija.

Primjeri popularnih mrežnih računalnih igara

Popularne mrežne igre poput "[Fortnite](#)", "[League of Legends](#)" i "[Call of Duty](#)" ilustriraju uspješnu primjenu mrežnog programiranja. "Fortnite" koristi klijent-server arhitekturu za brzu i sinkroniziranu igru na velikim mapama. "League of Legends" oslanja se na složene mrežne algoritme za smanjenje kašnjenja i osiguravanje poštenog natjecanja. "Call of Duty" primjer je kako skalabilni serveri mogu podržavati veliki broj istovremenih igrača, nudeći bogato i dinamično online iskustvo. Ove igre i mnoge druge koriste različite tehnike i tehnologije za upravljanje skalabilnošću servera, ali zajednički cilj im je osigurati da veliki broj igrača može zajedno uživati u bogatom i dinamičnom svijetu bez prekida ili problema s performansama. Razvoj naprednih algoritama za uravnoteženje opterećenja i efikasno upravljanje resursima nastaviti će omogućavati razvoj još ambicioznijih online igara u budućnosti.



slika 3

Zaključak

Programiranje mrežnih računalnih igara složen je zadatak koji zahtijeva duboko razumijevanje mrežnih protokola, arhitektura i izazova online okruženjima. Kroz primjenu naprednih tehnika i alata, programeri su sposobni stvoriti uzbudljive i interaktivne igre koje povezuju ljude diljem svijeta. Uspjeh igara kao što su "Fortnite", "League of Legends" i "Call of Duty" svjedoči o važnosti kvalitetnog mrežnog programiranja i njegovoj sposobnosti da stvori bogate, dinamične svjetove u kojima igrači mogu istovremeno uživati u igri.

Literatura

<https://repozitorij.unipu.hr/islandora/object/unipu%3A5775/datastream/PDF/view>

<https://transactions.games/>

<https://unity.hr/unity-i-multiplayer-igre-razvoj-mrezne-funkcionalnosti/>

slika 1: https://www.google.com/url?sa=i&url=https%3A%2F%2Funity.hr%2Funity-i-multiplayer-igre-razvoj-mrezne-funkcionalnosti%2F&psig=AOvVaw1HC2-CRc1dtIDbWXQwNA59&ust=1710886994839000&source=images&cd=vfe&opi=89978449&ved=0CBMQjRxqFwoTCPD8teTK_oQDFQAAAAAdAAAAABA

slika 2: https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.quora.com%2Fis-video-game-development-considered-computer-programming&psig=AOvVaw1YyaV_7W4hi2tqjnKrzDNT&ust=1710887767769000&source=images&cd=vfe&opi=89978449&ved=0CBMQjRxqFwoTCKDL_9TN_oQDFQAAAAAdAAAAABAE

slika 3: https://www.google.com/url?sa=i&url=https%3A%2F%2Fstore.epicgames.com%2Fen-US%2Fp%2Ffortnite&psig=AOvVaw0-9iVhMmi-yqQ1g6y_iYGq&ust=1710887889227000&source=images&cd=vfe&opi=89978449&ved=0CBMQjRxqFwoTCMDKrl_O_oQDFQAAAAAdAAAAABAE