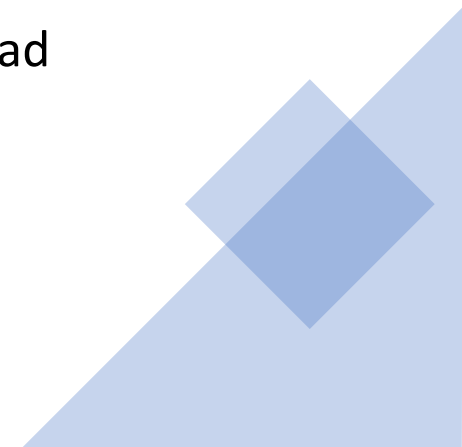


# Upravljanje resursima i raspoređivanja (scheduling)

Računarstvo u oblaku  
(Cloud Computing)



# Sadržaj

- Uvod – upravljanje resursima
  - Upravljanje resursima u oblaku
  - Politike i mehanizmi upravljanja resursima
  - Primena teorije upravljanja i optimizacije na upravljanje resursima
  - Upravljanje resursima i raspoređivanje (scheduling)
  - Raspoređivanje (scheduling) u oblaku
  - Ciljevi koje raspoređivanje treba da postigne
  - Upravljanje opterećenjem (Workload management)
- 

# Upravljanje resursima

- Upravljanje resursima je osnovna funkcija koju obavlja bilo koji sistem koji je napravio čovek. Direktno utiče na tri osnovna kriterijuma za evaluaciju nekog sistema: performanse, funkcionalnost i cena.
  - Neefikasno upravljanje resursima ima direktne negativne posledice na performanse i eksploatacionu cenu sistema, a najčešće indirektno pogađa i funkcionalnost sistema (jer izvršavanje nekih funkcija može postati preskupo ili trajati predugo).

# Upravljanje resursima u oblaku

- Računarsko okruženje u oblaku je veoma kompleksan sistem koji treba da ispunjava nepredvidljive zahteve i na koji utiču i spoljašnji događaji koji su van njegove kontrole. Upravljanje resursima u oblaku zahteva složene politike upravljanja i odlučivanja koje se zasnivaju na “višeciljnoj” optimizaciji (*multi-objective optimization*).
  - Upravljanje resursima u oblaku je veoma izazovan problem upravo zbog kompleksnosti sistema, gde je skoro nemoguće imati uvid u tačno globalno stanje sistema, kao i zbog velikog broja različitih i u suštini nepredvidljivih interakcija koje takav sistem ima sa okruženjem.

# Upravljanje resursima u oblaku

- Usluge u oblaku isporučuju se po različitim modelima (IaaS, PaaS, SaaS) i strategije za upravljanje resursima se mogu razlikovati u zavisnosti od ovih modela.
- U bilo kom modelu ponuđač usluge mora da reši problem obezbeđivanja obećane elastičnosti u situaciji kada se suočava sa velikim, fluktuirajućim opterećenjem.
  - U nekim situacijama, kada je povećanje opterećenja predvidljivo alokacija resursa se može planirati unapred.
  - Ako se desi nagli nepredviđeni skok opterećenja, *autoscaling* algoritam može da se u konačnom vremenu izбори s tim, ali samo ako
    - A) u tom momentu postoje resursi koji su slobodni ili se mogu osloboditi i dodeliti za izvršavanje usluge
    - B) Postoji dobar monitoring podsistem koji omogućava kontrolnoj petlji da u realnom vremenu donese odluku koji resursi se mogu realocirati

# Upravljanje resursima u oblaku

- Skoro od samog nastanka sistema u oblaku polemiše se da centralizovana kontrola teško da može da se izbori sa ovim brzim i nepredvidljivim promenama i da obezbedi kontinualnu isporuku servisa sa odgovarajućim stepenom kvaliteta
- Autonomne politike odlučivanja o raspoređivanju resursa su ovde od interesa zbog:
  - veličine samog sistema, velikog broja zahteva, velikog broja korisnika, nepredvidljivosti opterećenja pojedinih servisa

# Politike i mehanizmi upravljanja resursima

- Politika definiše donošenje ključnih odluka, a mehanizam predstavlja način kako se te ključne odluke sprovode u delo (impelmentiraju). U računarstvu je često bitno napraviti jasnu separaciju ove dve stvari.
- Politike upravljanja resursima u oblaku se mogu grupisati:
  - Kontrola prihvatanja (admission control)
  - Alokacija kapaciteta (Capacity allocation)
  - Balansiranje opterećenja (Load balancing)
  - Optimizacija utroška energije
  - QoS garancije

# Politike i mehanizmi upravljanja resursima

- *Admission control* – sprečava sistem da prihvati rad na zadacima (workload) koji narušavaju postavljene politike sistema na visokom nivou.
  - Npr. ne prihvatati novi posao ukoliko to sprečava da se tekući ili već pripremljeni zadaci završe.
- *Capacity allocation* – alociranje resursa za pojedinačnu aktivaciju servisa.
  - Kada se stanje svakog sistema često menja, ovo podrazumeva pretraživanje preko velikog broja čvorova da se nađe odgovarajući.
- *Load balancing* - distribuiranje opterećenja ravnomerno između raspoloživih instanci servera (ili servisa).
  - Npr. opterećenje se rasporedi da svaki server radi sa 50% kapaciteta. Ali u cloudu ovo često ima drugačiji smisao – minimizovati i troškove – onda ovakvo rešenje nije dobro, bolje je posao rasporediti tako da se minimizuje broj servera koji učestvuje u obradi a ostali prebace u stand-by
- *Energy optimisation* – minimizovanje energije utrošene na izvršavanje zadataka
  - Npr. snižavanje takta i napona na procesoru može neznatno da smanji performanse, a mnogo više utrošak energije
- *Quality of service (QoS) guarantees* – sposobnost da se ispune vremenski ili drugi zahtevi specificirani u ugovoru - Service Level Agreement (SLA).



# Primena teorije upravljanja i optimizacije na upravljanje resursima

- Teorija upravljanja se dugo koristi kao način za analizu i dizajn sistema za adaptivno upravljanje resursima (upravljanje potrošnjom energije, upravljanje raspoređivanjem zadataka, prilagođavanje QoS u web serverima, balansiranje opterećenja).
- U svim ovim slučajevima koristi se klasična metoda povratne veze pomoću kojih se regulišu ključni operativni parametri sistema, a na osnovu merenja trenutnog izlaza iz sistema.
- Optimalno upravljanje generiše sekvencu kontrolnih (upravljajućih) ulaza (signala) estimirajući promene sistema. Koriste se tehnike optimizacije koje su usmerene na minimizaciju konveksne funkcije “koštanja” (*cost function*).

# Upravljanje resursima i raspoređivanje (scheduling)

- Šta je problem raspoređivanja (scheduling) u računarskom sistemu – odlučivanje kako dodeliti resurse sistema (vreme na procesoru, memoriju, prostor za skladištenje, I/O i mrežni propusni opseg) različitim zadacima i korisnicima.
- Scheduler - program koji implementira određeni algoritam raspoređivanja resursa.

# Raspoređivanje (scheduling) u oblaku

- Raspoređivanje je kritična komponenta upravljanja resursima u oblaku.
- Odgovorno je za efikasno deljenje/multipleksiranje resursa na različitim nivoima.
  - Hardver servera može deliti više virtuelnih mašina, na svakoj od njih se može nalaziti aktivno više programa, od kojih svaki može imati pokrenute više niti.
  - Raspoređivanje na CPU (CPU scheduling) omogućava virtuelizaciju procesora, gde se svaka nit “vidi” ponaša kao virtuelni procesor.
  - Komunkacioni link može biti multipleksiran tako da ga koristi više virtuelnih kanala.

# Raspoređivanje (scheduling) u oblaku

- Scheduler (raspoređivač) odlučuje o:
  - Količini resursa koje je neophodno dodeliti nekome
  - Trajanju alokacije
- Ove odluke se donose na osnovu algoritma koji mora da bude:
  - Efikasan
  - Pravedan
  - Sprečava izgladnjivanje (starvation-free)

# Ciljevi koje raspoređivanje treba da postigne

- Ciljevi mogu da variraju u zavisnosti od vrste aplikacija
  - Batch sistemi – cilj je maksimizovati propusnost
  - Sistemi u realnom vremenu (real-time) - završiti posao pre zadatog roka (deadline)
- Neki česti algoritmi raspoređivanja
  - Round-robin
  - First-come-first-serve
  - Shortest-job-first

# Vremenska ograničenja - *deadlines*

- „Tvrdo“ ograničenje – ukoliko se neki zadatak ne završi do zadatog roka, drugi zadaci, koji od njega zavise, mogu biti pogođeni i postoje penali (kazne). Precizan je i izražava se u milisekundama, sekundama
- „Meko“ ograničenje – više kao smernica kada obaviti posao, i u opštem slučaju ne dovodi do penala ukoliko se to ne desi “na vreme”. Ograničenje može i da se „prebaci“ za određeni vremenski interval bez štetnih posledica.
- Multimedijalne aplikacije (audio, video streaming)
  - Imaju “meka” vremenska ograničenja
  - Zahtevaju statistički garantovano maksimalno očekivano kašnjenje i propusni opseg.
- Aplikacije u realnom vremenu imaju “tvrda” vremenska ograničenja

# Upravljanje resursima na virtuelizovanim serverima

- Stvari koje treba uzeti u razmatranje
  - Postoji veći broj raznovrsnih zadataka (workloads)
  - Konačan broj zadataka (workloads) se može obavljati na svakom serveru
  - Zadaci mogu imati promenljive zahteve u pogledu vremena potrebnog za njihovo izvršavanje
  - Performanse i izolacija resursa između različitih zadataka

# Terminologija raspoređivača CPU

- Dosta raspoređivača radi po principu proporcionalne zastupljenosti – Proportional Share (PS)
  - PS raspoređivanje alokira CPU u srazmeri sa “udelom” (shares), odnosno težinom koja je na neki način dodeljena zadatku
- Kriterijumi evaluacije raspoređivanja
  - Pravičnost (*fairness*) – vremenski interval tokom kojeg raspoređivač obezbeđuje pravičnu raspodelu CPU resursa
  - Greška alokacije – razlika između stvarno alociranih resursa i traženih resursa
- Fair-Share nasuprot Proportional-Share raspoređivanja
  - Raspoređivanje po principu pravičnog udela pokušava da obezbedi vremenski usrednjenu proporcionalnu raspodelu, na osnovu merenja stvarnog korišćenja resursa tokom dužeg intervala



# Primer pravičnog i proporcionalnog raspoređivanja

- Problem: dva klijenta C1 i C2 dele sistem sa istovetnim udelom u CPU resursima. C1 koristi CPU već izvesno vreme a C2 je blokiran
  - Šta se dešava kada C2 ponovo postane aktivan?
- Pravični algoritam raspoređivanja će u tom momentu povećati „udeo“ za C2 i obezbediti mu duži pristup CPU kako bi „dostigao“ C1
- Proporcionalni algoritam će tretirati C1 i C2 istovetno, jer nije potrebno penalizovati C1 za veće korišćenje resursa koji bi inače bili nekorišćeni (*idle*), jer je C2 svakako bio blokiran.

# Terminologija raspoređivača CPU (nastavak)

- Work-conserving (WC mode) – očuvanje izvršavanja zadataka
  - Udeo u CPU resursima je samo „garancija“ učešća. Sve dok ima posla koji se treba obaviti i svi klijenti su iskoristili svoj udeo, CPU će nastaviti da bude korišćen
- Non Work Conserving (NWC mode)
  - Udeo u CPU predstavlja limit. Klijenti dobijaju svoj udeo u CPU i ništa preko toga

# Terminologija raspoređivača CPU (nastavak)

- Non-preemptive raspoređivači
  - Dozvoljavaju klijentima koji trenutno vrše obradu da dovrše iskorištavnje CPU resursa. Ovakvi raspoređivači donose odluku o prebacivanju CPU resursa na drugog klijenta tek nakon što tekući klijent oslobodi CPU
- Preemptive raspoređivači
  - Tekući klijent čiji se zadatak izvršava može biti „ispražnjen“ iz CPU kako bi ostali klijenti došli na red. Ovakvi raspoređivači odluke o preraspodeli resursa donose čim se pojavi novi klijent.

# Upravljanje opterećenjem (Workload management)

- Sposobnost da se aplikacijama **precizno** dodele resursi (CPU, memorija, I/O). Aplikacije saopštavaju koje performanse su im potrebne (u skladu sa očekivanim nivoom usluge), a *workload manager* se brine da se aplikaciji dodele potrebni resursi kako bi ispunila te zahteve.
- pristupi koji se koriste:
  - Statički – potreba za resursima se procenjuje samo jednom i oni se statički dodeljuju aplikaciji čak i kada postoji varijabilno opterećenje
  - Dinamički – resursi se dinamički prilagođavaju trenutnim zahtevima aplikacije
- Workload manageri obično koriste PS u kombinaciji na NWC režimom
  - Kako bi obezbedili izolaciju performansi

# Neki od algoritama raspoređivanja CPU resursa

- Pozajmljeno virtuelno vreme (Borrowed Virtual Time)
- Jednostavni algoritam prednosti za najranije rokove (Simple Earliest Deadline First)
- Kreditno raspoređivanje (Credit Scheduler)

# Borrowed Virtual Time - BVT

- Cilj: koristi princip pravične raspodele bazirane na konceptu virtuelnog vremena, dodeljujući CPU onim nitima čija VM ima najmanje stanje brojača virtuelnog vremena
- Omogućava aplikacijama koje su osetljive na kašnjenje da se „vrate unazad“ u virtuelnom vremenu i time podignu svoj prioritet, i na taj način „posuđuje“ vreme od svojih budućih alokacija
- Osobine:
  - Preemptive, WC režim
  - Optimalno pravična raspodela
  - Mali *overhead* za implementaciju
- Ne podržava NWC pa ograničava izolaciju

# Simple Earliest Deadline First - SEDF

- Kod SEDF svaki domen (izvršivi zadatak – proces, nit) specificira tri vrednosti: ( $s_i$ ,  $p_i$ ,  $x_i$ ), gde su:
- $s_i$  := vremenski odsečak (slice)
- $p_i$  := period
- $Dom_i$  dobija si vremenskih jedinica unutar perioda  $p_i$ .
- $x_i$  := određuje da domen može dobiti ekstra vreme na CPU (WC-mode)
- Ako je WC režim omogućće, SEDF distribuira slobodno vreme ravnomerno između domena koji čekaju izvršavanje.
- Primer: Kako se može dodeliti 30% CPU nekom od domena (zadataka)?
- (3ms, 10ms, 0), ili (30ms, 100ms, 0)
- Ima li razlike?

# Simple Earliest Deadline First - SEDF

- Za svaki domen SEDF prati još dve vrednosti ( $d_i$ ,  $r_i$ ), gde je:
- $d_i$  := momenat u vremenu kada se za  $Dom_i$  završava njegov tekućem period (deadline)
- $r_i$  := preostalo alocirano CPU vreme koje  $Dom_i$  ima u tekućem periodu
- Politika raspoređivanja: Domen koji ima najbliži rok se bira da bude sledeći na redu za izvršavanje na CPU
- SEDF osobine:
  - Preemptive, WC i NWC režimi rada
  - Pravična raspodela zavisi od dužine perioda
  - Impelmentira se za CPU redove, ali mu nedostaje sposobnost globalnog balansiranja na multiprocesorskim okruženjima



# Credit Scheduler

- PS raspoređivač sa automatizovanim balansiranjem opterećenja na virtualnim CPU (vCPU) formiranim nad fizičkim CPU
- Pre nego fizički CPU pređe u *idle* stanje, raspoređivač proverava da li postoje drugi CPU-ovi kod kojih neki od vCPU čeka u stanju spremnom za izvršavanje
- Cilj – ovaj algoritam obezbeđuje da nijedan stvarni fizički CPU ne bude u *idle* stanju sve dok ima posla koji čeka na izvršavanje
- Za svaku VM raspoređivač dodeljuje težinu (*weight*) i limit (*cap*)
- Ukoliko je limit 0, tada ta VM može dobiti i dodatno potrebno vreme (WC režim rada)
- Ako je limit različit od nule, onda on reprezentuje maksimalni odsečak vremena koje VM može dobiti na procesoru (NWC)

# Credit Scheduler

- Raspoređivač radi sa odsečcima od 30ms: vCPU dobija 30 ms za izvršavanje pre nego što bude zamenjen nekim drugim vCPU
- U svakom vremenskom odsečku prioritet (kredit) za svaku CM se preračunavaju
- Raspoređivač radi monitoring na svakih 10 ms
- Osobine:
  - Non preemptive, WC i NWC
  - Globalno balansiranje opterećenja na multiprocesorskim sistemima

# Zaključak

- Pitanja?