Telekomunikacijskih inženiring

doc.dr. Iztok HUMAR
prof.dr. Janez BEŠTER

Agenda

- Uvod, telekomunikacijski inženiring
- Osnovni pojmi telekomunikacijskega inženiringa
- Osnovni pojmi o delovanju telekomunikacijskih omrežij (ponovitev)

Uvod

Uvod

- Telekomunikacijski inženiring:
  - načrtovanje
  - razvoj
  - merjenje
  - modeliranje
  - simulacije
  - vodenje
  - nadzor
  - upravljanje
  - vrednotenje zmogljivosti omrežja:
  - zagotavljanje kakovosti
  - širitev

Telekomunikacijski inženiring

Osnovni pojmi telekomunikacijskega inženiringa

Načrtovanje omrežij

- Naloga načrtovalca omrežja je *zgraditi/razširiti* TK omrežje:
  - v skladu s potrebami uporabnikov
  - ki je učinkovito
  - napraviti to ekonomično
- Metodologija načrtovanja omrežij:
  - od začetka
  - nadgradnja obstoječih sistemov

Razvoj omrežij od začetka

- Zahtevnejša metodologija. Možni pristopi:
- Pridobivanje informacij/izkušenj sorodnih omrežij iz tujine
  - težko pridobiti verodostojne podatke
  - ni nujno, da se lahko medsebojno enačimo (nacionalne in kulturne razlike)
  - lahko smo/želimo biti pionirji v svetovnem smislu
- Izkušnje na podlagi sorodnih storitev
  - poskusimo identificirati sorodne storitve, iz katerih črpamo podatke
- Pilotska implementacija
  - postavimo pilotsko implementacijo
  - vrednotimo odziv zgodnih posvojiteljev
- Predpostavke, simulacije
  - predpostavimo potencialne uporabnike, odzive
  - simulacija

Nadgradnja obstoječih sistemov

- Enostavnejša metodologija
  - mogoče črpati podatke, zbrane v nadzornih sistemih operaterjev
  - analize, simulacije, testi
  - možnost pilotskih in postopnih prehodov na nov sistem

Tendence

- Konkurenca med ponudniki infrastrukture nosilnih storitev
  - v hrbteničnih omrežjih
  - v dostopovnih omrežjih
- Razdalje nimajo več stroškov
  - optika
- Dosegljivost koderkoli
  - mobilnost
  - satelitske komunikacije

Vodenje/upravljanje omrežij

- Posamezne elemente telekomunikacijskih omrežij je potrebno voditi, nadzorovati, upravljati:
  - centralno vodenje telekomunikacijskih omrežij
  - porazdeljeno vodenje telekomunikacijskih omrežij
- Pristopi za upravljanje v telekomunikacijskih omrežjih:
  - SNMP
  - CORBA
  - WWW + XML

Merjenje

- Omogoča natančno pridobivanje podatkov o parametrih omrežij
  - zelo učinkovito (velika natančnost)
- Zahteven in drag pristop
- Lahko poteka v realnem času
- Draga merilna oprema
- Zamudno odpravljanje napak
- Velikokrat neizvedljivo (npr. načrtovanje novih sistemov)

Modeliranje

- Modeliranje je proces gradnje modela
- Model sistema:
  - je poenostavljena predstavitev sistema
    - ki analitiku omogoča razumevanje odziva na vhodne parametre
  - mora biti čimbolj podoben realnemu sistemu
    - upoštevati je potrebno ključne lastnosti in parametre sistema
  - mora biti čimbolj enostaven
- Dober model je kompromis med realnostjo in enostavnostjo
- Model je navadno matematičen opis sistema:
  - reševanje različnih vrst in sistemov enačb
  - računalnik le kot matematični pripomoček
- Rezultati:
  - matematično natančni
  - realno približni, zaradi poenostavitev
  - v primeru velikih omrežij so rezultati lahko neuporabni

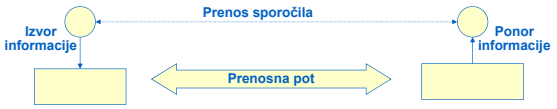
Simulacije

- Uporaba modela sistema
- Reševanje s pomočjo simulacijskih programov
- Časovni potek običajno ni realen
- Najprimernejša metoda za vrednotenje TK omrežij
- Zvezne simulacije
- Diskretne simulacije
  - opis stanja sistema z diskretnimi spremenljivkami
  - diskretni dogodek
  - izvrševanje po kronološkem redu
- Orodja za izvajanje simulacij
  - ns2
  - ComNet

Osnovni pojmi o delovanju TK omrežij

Naloga omrežja

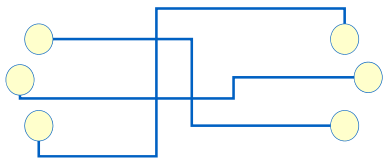
- Naloga omrežja je prenos podatkov od izvora do ponora med oddaljenima entitetama



- Pogosto je lahko hkrati dejavnih več izvorov in ponorov informacije
  - Prenos sporočil med posameznimi pari I in P lahko časovno in prostorsko sovpada (isto omrežje; skupna prenosna pot)
- Sporočila na prenosni(h) poti(eh) zato podatkovne tokove združujemo/prepletamo/kombiniramo/povezujemo
  - Multipleksiranje
  - Komutacija

Neposredne povezave

- Nekomutirane tokokrogovne povezave



Povezovanje uporabnikov

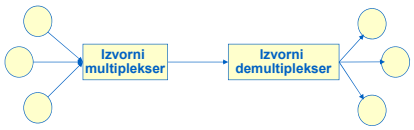
- Sprva povezovanje vsakega uporabnika z vsakim
  - Število zvez je enaka vsoti števila stranic in diagonal v N-kotniku
  - $N*(N-1)/2 \approx N^2$
- Uvedba komutacijskih stikal
  - Povezujejo uporabnike, ki s si (geografsko) blizu
  - Danes so komunikacijska stikala integrirana vezja
- Uporabniki komunicirajo tudi med različnimi skupinami
  - Zelo oddaljene skupine (Telefon, Internet)
  - Hierarhično povezovanje komutacijskih stikal med seboj
  - Rast omrežja:
    - Mestna (MAN) omrežja
    - Regionalna omrežja
    - Državna omrežja
  - Večina prometa se zaključuje v vozlišču.
    - Telefonsko omrežje 10:1 do 5:1

Multipleksiranje in komutacija

- **Multipleksiranje v viru**
  - če multipleksiramo samo v viru in
  - če je povezava med izvorom in ponorom enoumno določena (povezava točka-točka)
  - komutacija NI potrebna.
- **Komutacija (usmerjanje)**
  - = dinamično/sprotno multipleksiranje v omrežju
    - temelji na vzpostavljanju povezav oz.
    - sprotnem odločanju o tem, kam usmeriti določeno PDU
  - možna je:
    - statična/permanentna postavitev pravil komutacije
    - dinamična/sprotna postavitev pravil komutacije
      - ločeno za vsak PDU, sejo, zvezo
      - potrebujemo signalizacijo

Multipleksiranje in komutacija

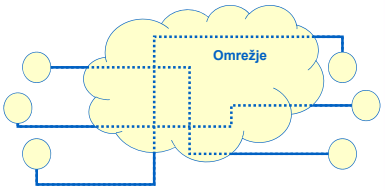
- **Multipleksiranje**



- **Komutacija**

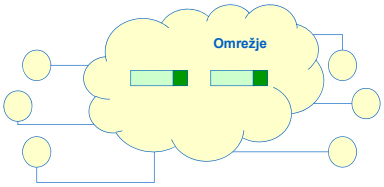
Tokokrogovne povezave in komutacija

- **Tokokrogovna komutacija**
  - fleksibilno postavljanje tokokrogovnih povezav
    - statična top.
    - dinamična top.



Paketna komutacija

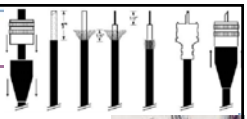
- **Paketna komutacija**
  - vsak paket (PDU) nosi vse podatke za usmerjanje skozi omrežje
  - paket je lahko spremenljive dolžine
  - izvor lahko oddaja toliko paketov, kot to dopušča celotno omrežje (ne pa le kapaciteta tokorogovne povezave)



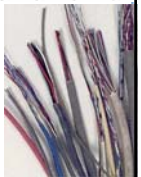
Začilnosti različnih tipov komutacij

	Tokokrogovna komutacija	Paketna komutacija
Prednosti	▪stalno zagotovljeni viri v omrežju ▪ni medsebojnega vpliva povezav ▪predvidljive zakasnitve, majhno potresavanje zakasnitve ▪kvaliteta	▪možnost statističnega multipleksiranja ▪možnost podpore izbruhom prometa ▪omejitve oddaje sporočil postavlja omrežje (ne tokokr. povezava)
Slabosti	▪lahko slabša izkoriščenost virov v omrežju (odvisno od tipa prometa!) ▪nujno je vzpostavljanje povezav	▪zakasnitve pri prenosu niso nujno predvidljive ▪v nepovezavnih omrežjih večja verjetnost napak, manjša zanesljivost ▪medsebojni vpliv posameznih tokov sporočil ▪potebna je obdelava vsakega posameznega paketa

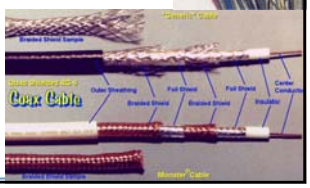
Prenosni sistemi



- **Bakreno omrežje**
  - Parica
  - UTP
  - Koaksialni kabel



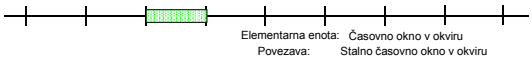
- **Brezžično omrežje, radijsko omrežje**
  - Celično omrežje (=dostopovni del)
  - Mikrovalovne povezave (=hrbtenica)
- **Optično omrežje**



- **Prenos podatkov po prenosnem sistemu:**
  - Električni signali
  - Elektromagnetno valovanje
  - Svetloba

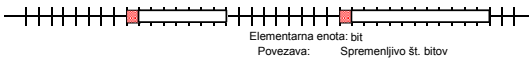
Primerjava prenosnih tehnik

Synchronous Transfer Mode



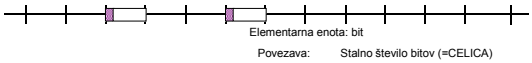
Elementarna enota: Časovno okno v okviru
Povezava: Stalno časovno okno v okviru

Packet Transfer Mode



Elementarna enota: bit
Povezava: Spremenljivo št. bitov

Asynchronous Transfer Mode - ATM



Elementarna enota: bit
Povezava: Stalno število bitov (=CELICA)

Oblike prenašanih podatkov

- Podatke lahko prenašamo v obliki zaključenih sporočil (messages)
  - Elektronska pošta
  - SMS sporočilo
- Podatke lahko prenašamo v obliki toka podatkov (stream)
  - Telefonski pogovor
  - Videokonferenčna zveza

Princip povezavnosti

- Ali komunicirajoči osebki pred izmenjavo podatkov vzpostavijo zvezo/sejo ... ali ne
- Govorimo o (ne)povezavnih protokolih/omrežjih/komunikacijah

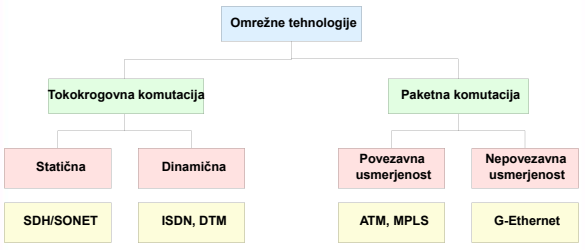
	Povezavno	Nepovezavno
Prednosti	▪možna večja zanesljivost prenosa, odprava napak ▪možnost nadzora pretoka količine sporočil, preprečevanje zasičenja	▪enostavnost
Slabosti	▪potebujemo signalizacijo ▪protokoli/omrežja so kompleksnejša	▪nezmožnost zanesljivega prenosa, zagotavljanja vrstnega reda prenesenih sporočil

Relacija povezavnost - komutacija

- Tokokrogovna omrežja so vedno povezavno naravnana
  - Povezavnost zagotavlja/vsiljuje že sam fizični tokokrog
- Paketna omrežja/povezave so lahko povezavna ali nepovezavna
  - (Ne)povezavnost je odvisna od uporabljenih komunikacijskih protokolov

		Komutacija	
		Tokokrogovna	Paketna
Povezavnost	Povezavno	ISDN, PSTN, SDH	ATM, MPLS
	Nepovezavno	/	Ethernet, IP

Umestitev tehnologij



Hvala za pozornost.
Vprašanja?

Laboratorij za telekomunikacije
Fakulteta za elektrotehniko

Telekomunikacijski inženiring

dr. Iztok Humar

Vsebina

- Značilnosti TK prometa, preprosti modeli, uporaba
  - Uvod
  - Značilnosti telekomunikacijskega prometa
  - Modeliranje vodovno komutiranih zvez
    - Erlang B
    - Erlang C
  - Modeliranje v paketnih omrežjih
    - M/M/1
  - Značilnosti internetnega prometa

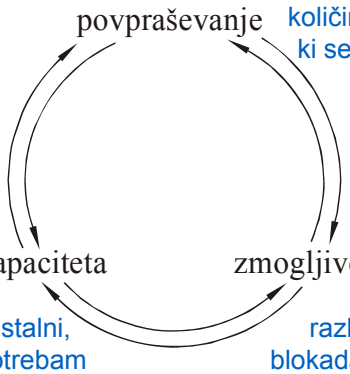
Telekomunikacijski inženiring

Možni načrtovalski pristopi

- "Od začetka":
  - težaven pristop
  - zahteva načrtovalca z izkušnjami, pristopa se ni mogoče "priučiti"
  - uporaba tujih izkušenj (drugi operaterji, podobni sistemi)
  - postavitev pilotskih sistemov – ustrezno skaliranje
  - uporaba tehnik modeliranja/simulacij
- Nadgradnja/dopolnjevanje obstoječih sistemov
  - enostavnejši pristop
  - dobro poznano izhodišče
  - običajno poznani vplivi parametrov/odzivi na spremembe v sistemu
  - mogoči poizkusi (omejenega obsega)
  - iterativni postopki
  - modeliranje in simulacije lahko služi kot dodatna podpora odločitvam

povpraševanje-kapaciteta-zmogljivost

- **Tradicionalna vloga TK inženiringa:**
  - **zagotavljati tolikšne kapacitete, kot so potrebne za prenašanje želenih podatkov v skladu z zahtevano kakovostjo storitev**
- **Medsebojna zveza treh značilnosti TK omrežij**



količina in značilnosti prometnih tokov, ki sestavljajo prometno obremenitev: porazdelitve prihodov, trajanje klicev ...

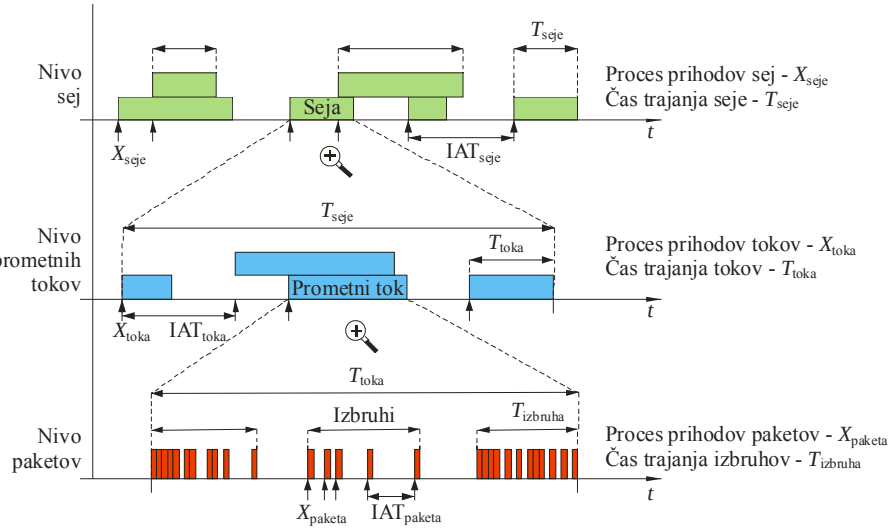
omrežni viri: stalni, prilagodljivi potrebam

različne metrike QoS/QoE: blokada, čakanje, izgube paketov

LTFE.org
telekomunikacije in informacijske tehnologije
Laboratorij za telekomunikacije
Fakulteta za elektrotehniko

Značilnosti TK prometa (porazdelitvene funkcije)

Katere parametre želimo opisati?



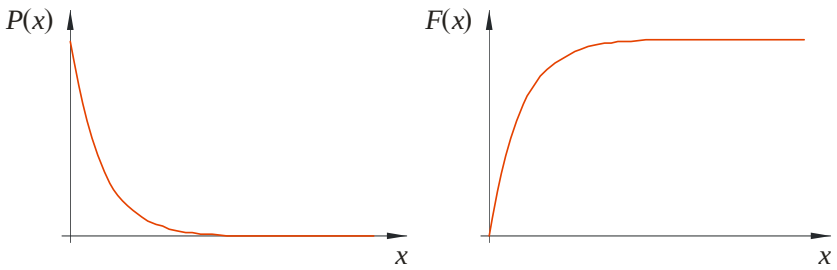
Eksponentna porazdelitev

Porazdelitev:

$$P(x) = \lambda e^{-\lambda x}$$

Kumulativna porazdelitvena funkcija:

$$F(x) = P(X \le x) = 1 - e^{-\lambda x}$$

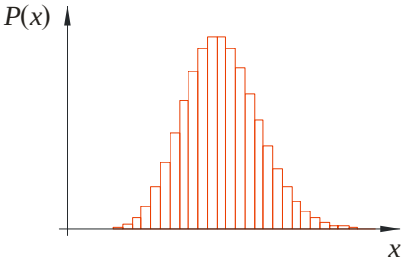


- **Značilnost: funkcija s kratkim repom** $F^c(x) \sim e^x, \quad x \to \infty$
- **Uporaba: eksponentna porazdelitev časov med prihodi zahtevkov se je uporabljala pri modelih klasične telefonije (porazdelitev časov med prihodi klicev) in zgodnjih teorijah čakalnih vrst (prihodi paketov) – v modelih Kleinrocka.**

Poissonova porazdelitev

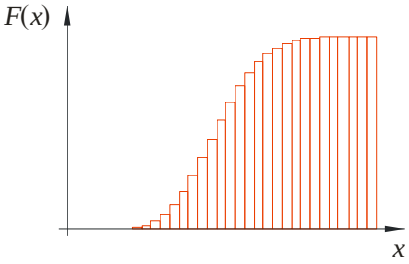
Porazdelitev:

$$P(x) = \frac{e^{-\lambda}\lambda^x}{x!}, \quad x = 0,1,2,\ldots$$



Kumulativna porazdelitvena funkcija:

$$F(x) = \sum_{i=0}^{x} \frac{e^{-\lambda}\lambda^i}{i!}$$

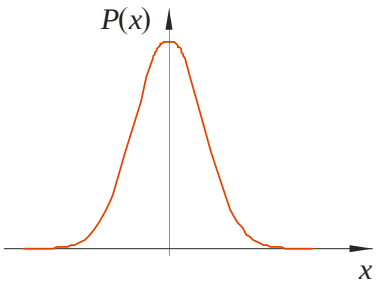


- (1) časi so eksponentno porazdeljeni in (2) časi so med seboj neodvisni.
- Uporaba: Poissonova porazdelitev je najpogosteje uporabljena v modelih s področja klasične telefonije. Erlangovi modeli verjetnosti blokade in čakanja slonijo na Poissonovi porazdelitvi števila prihodov klicev v časovnih intervalih določene dolžine.

Normalna porazdelitev

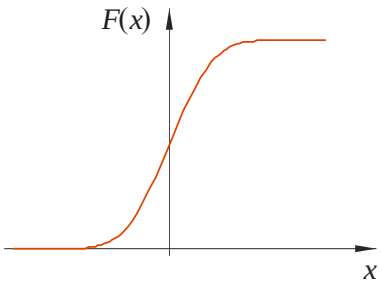
Porazdelitev:

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$



Kumulativna porazdelitvena funkcija:

$$F(x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} e^{\frac{-(x-\mu)^2}{2\sigma^2}} dx = \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{x-\mu}{\sigma\sqrt{2}}\right)\right]$$



- Matematiki tovrstno porazdelitev imenujejo normalna porazdelitev, fiziki jo nazivajo Gaussova porazdelitev, v družboslovju pa jo imenujejo zvonasta krivulja.

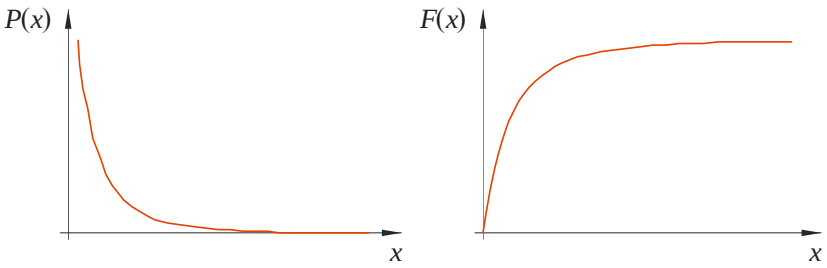
Pareto porazdelitev

Porazdelitev:

$$P(x) = \alpha b^{\alpha} x^{-\alpha-1}$$

Kumulativna porazdelitvena funkcija:

$$F(x) = 1 - (b/x)^{\alpha}, \qquad \alpha, b \geq 0, \qquad x \geq b.$$



- **Uporaba: modeliranje deležev procesorskega časa posameznih računalniških procesov (1,05 ≤ α ≤ 1,25); v telekomunikacijah za modeliranje časov držanja klicev, velikosti oken pri VBR videu, opis časov med prihodi sej pri prometu protokola *telnet*, za opis velikosti FTPDATA izbruhov in za opis porazdelitve velikosti datotek.**

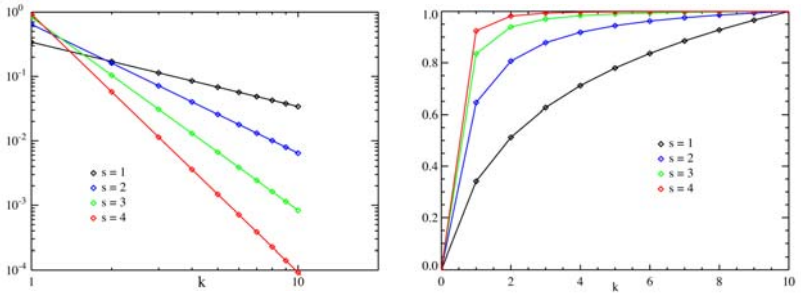
Zipfova in njej podobna porazdelitev

Porazdelitev:

$$P(k) = \Omega / k$$

$$P(k) = \Omega / k^{\alpha}$$

Kumulativna porazdelitvena funkcija:

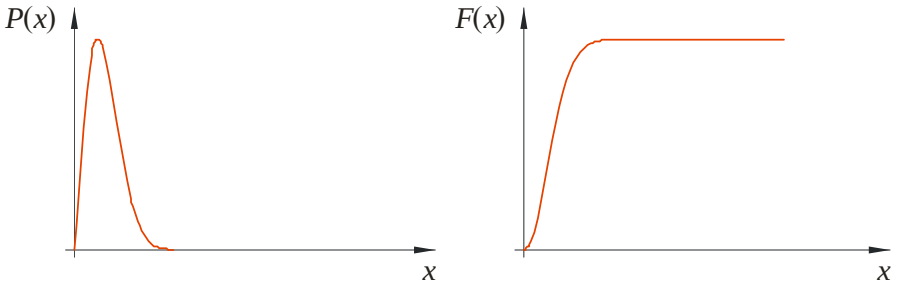


- **Zipfova porazdelitev se je prvič identificirala v knjižničarstvu (pogostnost besed). Uporaba v telekomunikacijah; modeliranje popularnosti virov – verjetnosti (pogostnosti) dostopa do npr. spletnih strani.**

Weibullova porazdelitev

Porazdelitev: **Kumulativna porazdelitvena funkcija:**

$$P(x) = (a/b)(x/b)^{(a-1)} e^{-(x/b)^a}, \qquad F(x) = 1 - e^{-(x/b)^a}, \qquad a, b \geq 0,$$



- **Uporaba: analiza čakalnih vrst v primeru uporabe samopodobnega prometa – porazdelitve v čakalnih vrstah se podrejajo Weibullovim porazdelitvam. Weibullova porazdelitev se uporablja tudi za modeliranje časov neaktivnosti uporabnikov med zaporednimi zahtevki HTTP.**

Praktična uporaba

- **Izkaže se, da za opis sej, ki so vezane na obnašanje uporabnikov (ljudi), ponavadi zelo dobro ustrezajo:**
  - **eksponentne porazdelitve časov med prihodi sej (oz. posledično)**
  - **Poissonove porazdelitve števila prihodov v časovnih intervalih**
- **Za opis entitet, vezanih na prenos datotek, pa se ponavadi uporabljajo:**
  - **porazdelitve z dolgim repom, npr. Pareto porazdelitev**
- **Bolj nadrobne značilnosti telekomunikacijskega prometa (prenos paketov) so precej bolj kompleksne. Zaenkrat ne obstajajo relativno preprosti modeli za njihov opis, ki bi omogočili široko uporabo.**
- **Nasvet: načrtovanje izvajamo na makroskopski ravni, določene nadrobne značilnost znotraj sej aproksimiramo z ustreznimi približki, pogosto kar CBR.**

Modeliranje vodovno komutiranih zvez

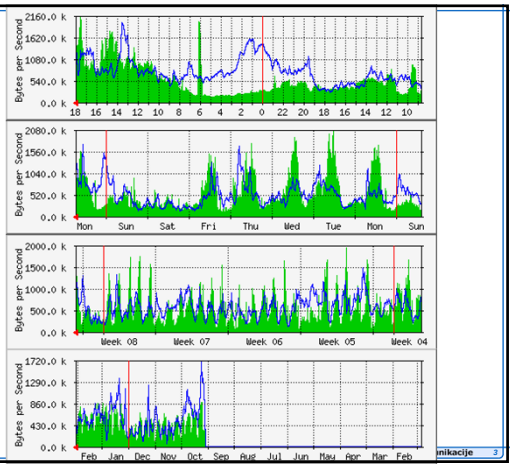
doc.dr. Iztok HUMAR
prof.dr. Janez BEŠTER

TKI

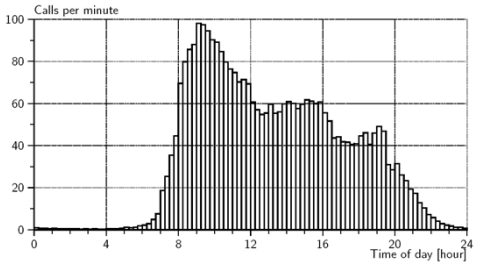
Vsebina

- Najbolj obremenjena ura (Busy Hour, BHCA)
- Erlang
- Stopnja strežbe (Grade of Service)
- Verjetnost blokade (Blocking Probability)

- Matematični model:
  - hiter in enostaven izračun
  - odstopanje od realnih porazdelitev – lahko slaba natančnost



Porazdelitev klicev preko dneva

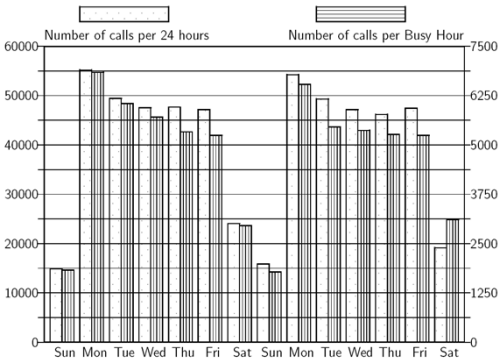


Najbolj obremenjena prometna ura

- Promet uporabnikov, ki obremenjujejo sistem močno variira glede na uro v dnevu in dan v tednu.
- Večina sistemov je močno obremenjenih le nekaj ur v celem dnevu: Najbolj obremenjene prometne ure.
- Glavni cilj načrtovanja sistemov je zagotoviti, da so sistemi dovolj zmogljivi za normalno (stabilno) delovanje v teh urah.
  - Sisteme načrtujemo z določeno stopnjo rezerve (npr. 25 % – 35 %); podano v ITU-T Recommendation Q.543.
  - Takšno načrtovanje zagotavlja, da bo sistem imel dovolj zmogljivosti za normalno delovanje tudi izven ur z maksimalno obremenitvijo.

- Nadaljnje načrtovanje se nanaša na sisteme v času največje obremenitve.

Pomen najbolj obremenjene ure

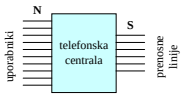


Erlang

- Erlang je enota za merjenje prometa.
- Promet v Erlangih definiramo kot:

$$a = \frac{\text{čas uporabe strežnika}}{\text{celoten čas}}$$

- Imamo 31 vodov. Na vsakem vodu komunicirajo po 30 min.
  - Čas uporabe je 30 min × 31 = 930 minut
  - Skupen čas 60 min
  - Promet je 930 min / 60 min = 15,5 erl
- Izračun ponujanega prometa:
  - *N* – število uporabnikov,
  - *a* – promet, ki ga generira povprečen uporabnik (npr. 3 min pogovora na uro → 3/60 = 0,05 Erl),
  - *A* – ponujan promet: *A* = *N* × *a* (npr. *N* = 100, *a* = 0,05 Erl → *A* = 5 Erl).



Sistemi z blokado (telefonske linije, VoIP)

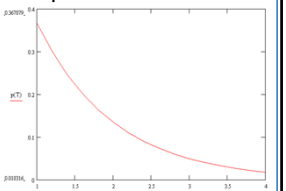
- Stopnja strežbe (Grade of Service)

$$B = GoS = \frac{\text{število izgubljenih klicev}}{\text{število ponujenih klicev}}$$

- Predstavlja verjetnost blokade – verjetnost, da bo uporabnik zavrnjen ob dostopu do strežnika.
- Izračuni verjetnosti blokade se nanašajo na najbolj prometno uro.
  - Verjetnost blokade merimo iz povprečja intenzitet prometa med 30 najbolj obremenjenimi dnevi v letu
- Verjetnost blokade lahko znižamo:
  - Če dodamo število strežnikov v sistem
  - Prerazporedimo promet iz najbolj prometne ure v ostale termine, ko je prometa v omrežju manj obremenjene termine

Erlang B model

- Omogoča izračun verjetnosti blokade – verjetnost, da so ob prihodu nove zahteve vsi strežniki zasedeni
- V tem primeru se nova zahteva blokira (zavrže) in se je ne obravnava več
- Predpostavlja:
  - eksponentne porazdelitve dohodnih časov novih zahtev,
  - eksponentne porazdelitve strežnih časov posameznih zahtev.



Erlang B model

- Parametri:
  - *A* – ponujan promet [Erlang] (zahteve),
  - *S* – število strežnikov (kapaciteta),
  - *B* – verjetnost blokade (kakovost).

$$B = \frac{\frac{A^S}{S!}}{\sum_{i=0}^{S} \frac{A^i}{i!}}$$

- Primer izračuna:
  - imamo 1000 uporabnikov (*N* = 1000),
  - uporabnik povprečno generira 0,05 Erlanga prometa (*a* = 0,05 Erl),
  - ponujan promet je torej 50 Erlangov (*A* = 50 Erl),
  - Verjetnost blokade naj ne bo večja od 1 % (*B* = 0,01),
  - Izračun pokaže, da potrebujemo 64 linij (*S* = 64).

Sistemi s čakanjem (klicni center)

- Omogoča izračun verjetnosti čakanja – verjetnost, da so ob prihodu nove zahteve vsi strežniki zasedeni
- V tem primeru se nova zahteva postavi v čakalno vrsto in čaka na prost strežnik
- Enako kot Erlang B predpostavlja eksponentne porazdelitve dohodnih časov novih zahtev in strežnih časov posameznih zahtev

Erlang C model

- Parametri:
  - *S* – število strežnikov,
  - *A* – ponujan promet [Erlang],
  - T_1 – sprejemljivi čas čakanja na prost strežnik [s],
  - T_2 – povprečni čas zadrževanja v strežniku [s],
  - *P*(>0) – verjetnost čakanja na prost strežnik,
  - $P(>T_1)$ – verjetnost, da čakamo na strežnik več kot T1 sekund,
  - D_1 – povprečni čas čakanja na prost strežnik [s],
  - D_2 – povprečni čas čakanja na prost strežnik dejansko čakajočih zahtev [s].

$$P(>0) = \frac{\frac{A^S S}{S!(S-A)}}{\frac{A^S S}{S!(S-A)} + \sum_{i=0}^{S-1} \frac{A^i}{i!}}$$

$$P(T_1) = P(>0)\, e^{-(S-A)\frac{T_1}{T_2}}$$

$$D_1 = P(>0)\frac{T_2}{(S-A)}$$

$$D_2 = \frac{T_2}{(S-A)}$$

Erlang C model - primer izračuna

- **Vhodni podatki:**
  - $S = 24$
  - $A = 20$ Erl
  - $T_1 = 4$ s
  - $T_2 = 4$ s
- **Rezultati:**
  - $P(>0) = 0{,}298$
  - $D_1 = 0{,}298$ s
  - $D_2 = 1$ s

$$P(>0) = \frac{\frac{A^S S}{S!(S-A)}}{\frac{A^S S}{S!(S-A)} + \sum_{i=0}^{S-1} \frac{A^i}{i!}}$$

$$P(T_1) = P(>0)\, e^{-(S-A)\frac{T_1}{T_2}}$$

$$D_1 = P(>0)\frac{T_2}{(S-A)}$$

$$D_2 = \frac{T_2}{(S-A)}$$

HVALA ZA POZORNOST,
VPRAŠANJA !?

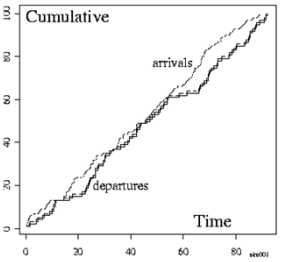
Modeliranje prometa v paketnih omrežjih

as.mag. Iztok HUMAR
prof.dr. Janez BEŠTER

Agenda

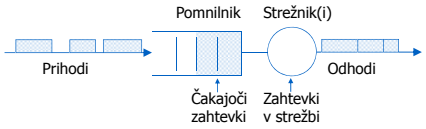
- Model splošnega strežnega sistema
- Parametri splošnega strežnega sistema
- Označevanje splošnih strežnih sistemov
- Strežni sistemi M/M/1
  - Čas med prihodi
  - Število prihodov v intervalu
  - Strežba
- Sistemska stanja
  - Verjetnosti
  - Prihodi med stanji
- Parametri zmogljivosti sistema
- Littleov Teorem

Strežni sistem



- Kumulativna razporeditev prihodov in strežbe
  - razliko med krivuljama predstavlja število elementov v vrsti
- Od sistema pričakujemo, da streže zahtevam v končnem času:
  - krivulji ne smeta divergirati druga od druge

Model strežnega sistema s čakalno vrsto

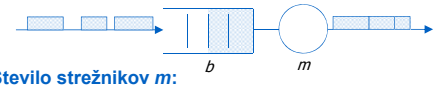


- Strežni sistem vsebuje enega ali več strežnikov
- Zahtevki prihajajo po določeno storitev
- Zahtevki, ki pridejo v sistem in ne prejmejo storitve takoj, so razporejeni v čakalno vrsto (pomnilnik)

Razlogi za zakasnitev v omrežju

- Čas procesiranja
  - Privzamemo lahko, da procesorska moč ne predstavlja omejitve
- Čas čakanja v vrsti
  - Čas čakanja v vrsti pred oddajo
- Čas prenosa po povezavi
  - Čas, ki je porabljen za prenos preko povezave (prenos električnega, optičnega signala)
  - Neodvisen od prometa, ki obremenjuje omrežje
- Največji po iznosu: čas čakanja v vrsti

Parametri strežnega sistema



- Število strežnikov *m*:
  - Eden
  - Več
  - Neskončno
- Velikost pomnilnika *b*
- Razvrščanje zahtevkov v vrsto (scheduling):
  - FCFS = FIFO,
  - LCFS = LIFO,
  - Processor Sharing (PS) ...
- Proces prihodov
- Proces strežbe

Kendall-Lee označevanje strežnih sistemov

- Omogoča enostaven opis strežnega sistema s čakalno vrsto

A/B/c:(XXXX/m/n)

- *A* določa proces prihodov
- *B* določa proces strežbe
  - M: porazdelitev Markovskega (eksponentna porazdelitev časa med prihodi)
  - D: deterministična (degenerate/deterministic) porazdelitev
  - E_k: Erlangova porazdelitev s parametrom k
  - G: splošna porazdelitev
- *c* določa število strežnikov/kanalov

Kendall-Lee označevanje strežnih sistemov

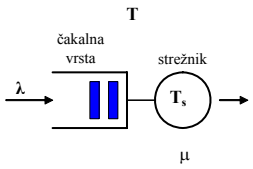
A/B/c:(XXXX/m/n)

- XXXX določa način strežbe
  - FCFS=FIFO, LCFS=LIFO, SIRO = Service In Random Order, prioritete
- *m* določa največje možno število uporabnikov v sistemu – skupaj v pomnilniku in prejemajočih storitev. Vsi ostali prihodi so zavrnjeni
- *n* določa največje število populacije, iz katere izvirajo uporabniki. To posredno omejuje hitrost prihodov.
- Pogosto se uporablja samo prva trojica oznake. V tem primeru se za drugi del privzame FIFO/∞/∞

Primeri označevanja

- M/M/1: Naključni prihodi in strežba, časi med prihodi in strežbo porazdeljeni po eksponentni porazdelitvi, en strežnik, neskončen pomnilnik in neskončna populacija, FIFO način razvrščanja
- M/M/c: Enaka porazdelitev kot pri prejšnjem, le da je v tem primeru c strežnikov
- M/G/3: Eksponentna porazdelitev časa prihodov, splošna porazdelitev časa strežbe, trije strežniki, neskončen pomnilnik in neskončna populacija, FIFO način razvrščanja
- */D/∞ : Sistem s konstantno zakasnitvijo
- M/M/c:FIFO/c: Eksponentna porazdelitev časa prihodov, splošna porazdelitev časa strežbe, c strežnikov, pomnilnika ni, neskončna populacija, FIFO način razvrščanja

Strežni sistem M/M/1



- En sam strežnik
- Neskončen pomnilnik

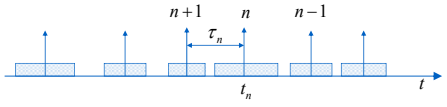
- Proces prihodov: Poissonov s številom zahtevkov λ
- Proces strežbe: eksponenten s parametrom $\mu = \frac{1}{T_S}$
- Stabilno stanje: λ < μ

- Časi strežbe in intervali prihodov so med seboj neodvisni

Naključni prihodi

- *λ* predstavlja povprečno število zahtevkov (arrival rate), ki vstopijo v sistem v časovni enoti.
- Verjetnost prihoda novega paketa v kratkem intervalu *h* je *λh* = konstanta.
- Verjetnost novega prihoda ni odvisna od časa zadnjega prihoda.
- Tovrstni proces prihodov imenujemo *Poissonov proces*.
- Pogoji za uporabo:
  - Število zahtev je veliko
  - Vpliv posamezne zahteve na celoten sistem je majhen
  - Vse zahteve so med seboj neodvisne t.j.: njihova odločitev za uporabo sistema je neodvisna od ostalih zahtev

Čas med prihodi

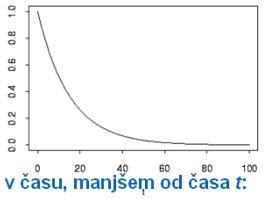


- τ_n čas med zahtevkoma n in n+1.
- τ_n je naključni čas.
- $\{\tau_n,\ n \geq 1\}$ predstavlja stohastični proces.
- Časi med zahtevki so enakomerno porazdeljeni in imajo skupno povprečno vrednost.

Eksponentna porazd. časa med prihodi

- Čas med prihodi τ_n predstavlja naključno spremenljivko z eksponentno porazdelitvijo s parametrom λ.
- Naključna spremenljivka τ_n ima eksponentno porazdelitev s parametrom λ, če je verjetnostna gostota funkcije podana kot:

$$f_{\tau_n}(t) = \begin{cases} \lambda e^{-\lambda t} & \text{if } t \geq 0 \\ 0 & \text{if } t < 0 \end{cases}$$



- Verjetnost, da pride do prihoda v času, manjšem od časa *t*:

$$F_{\tau_n}(t) = P(\tau_n \leq t) = \begin{cases} 1 - e^{-\lambda t} & \text{if } t \geq 0 \\ 0 & \text{if } t < 0 \end{cases}$$

Povprečni čas med prihodi

- Povprečna vrednost in varianca:

$$E[\tau_n] = \frac{1}{\lambda}, \quad \mathrm{Var}[\tau_n] = \frac{1}{\lambda^2}$$

➢ Dokaz:

$$E[\tau_n] = \int_0^\infty t f_{\tau_n}(t)\,dt = \int_0^\infty t\lambda e^{-\lambda t}\,dt =$$

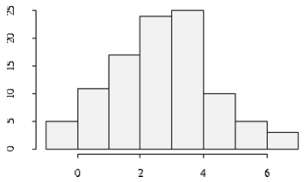
$$= -te^{-\lambda t}\Big|_0^\infty + \int_0^\infty e^{-\lambda t}\,dt = \frac{1}{\lambda}$$

$$E[\tau_n^2] = \int_0^\infty t^2 \lambda e^{-\lambda t}\,dt = -t^2 e^{-\lambda t}\Big|_0^\infty + 2\int_0^\infty te^{-\lambda t}\,dt = \frac{2}{\lambda}E[\tau_n] = \frac{2}{\lambda^2}$$

$$\mathrm{Var}(\tau_n) = E[\tau_n^2] - (E[\tau_n])^2 = \frac{2}{\lambda^2} - \frac{1}{\lambda^2} = \frac{1}{\lambda^2}$$

Število prihodov v intervalu

- V intervalu dolžine *D* je število prihodov naključna spremenljivka s Poissonovo porazdelitvijo s parametrom m=λ*D*.



- Slika prikazuje histogram porazdelitve za realen Poissonov proces.
- Pričakovano število prihodov v intervalu dolžine *D* je m=λ*D*.

Poissonova porazdelitev števila prihodov

- Proces ustreza Poissonovi porazdelitvi s parameterom *m*, če je verjetnostna funkcija podana kot:

$$P(X = x) = e^{-m}\frac{m^x}{x!}$$

- Široka uporabnost Poissonovega procesa pri modeliranju naključnih dogodkov, ki se zgodijo v danem časovnem intervalu.
  - Uporabniki, ki pridejo v trgovino tekom dneva
  - Klici na napačno številko tekom tedna
  - Študentje, ki pridejo v kabinet v času govorilnih ur
  - Paketi, ki prispejo v omrežno stikalo

Poissonova porazdelitev števila prihodov

- Srednja vrednost in varianca:

$$E[X] = m, \quad \mathrm{Var}(X) = m$$

➢ Dokaz:

$$E[X] = \sum_{x=0}^{\infty} xP(X=x) = e^{-m}\sum_{x=0}^{\infty} x\frac{m^x}{x!} = e^{-m}\sum_{x=0}^{\infty}\frac{m^x}{(x-1)!}$$

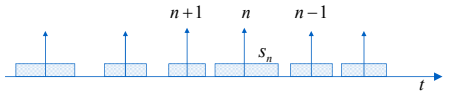
$$= e^{-m}m\sum_{j=0}^{\infty}\frac{m^j}{j!} = e^{-m}me^m = m$$

$$E[X^2] = \sum_{x=0}^{\infty} x^2 P(X=x) = e^{-m}\sum_{x=0}^{\infty} x^2\frac{m^x}{x!} = e^{-m}\sum_{x=0}^{\infty} x\frac{m^x}{(x-1)!}$$

$$= e^{-m}m\sum_{j=0}^{\infty}(j+1)\frac{m^j}{j!} = m\sum_{j=0}^{\infty} je^{-\lambda}\frac{m^j}{j!} + me^{-m}\sum_{j=0}^{\infty}\frac{m^j}{j!} = m^2 + m$$

$$\mathrm{Var}(X) = E[X^2] - (E[X])^2 = m^2 + m - m^2 = m$$

Čas strežbe



- s_n : Čas strežbe zahtevkov n strežnika
- $\{s_n,\ n \geq 1\}$ je stohastični proces
- Časi strežbe so enakomerno porazdeljeni in imajo skupno povprečno vrednost
- Ali je za pakete v stikalu čas strežbe res naključen?

Eksp. porazdelitev trajanja strežbe

- *μ* predstavlja število zahtev, ki jih strežnik lahko postreže v časovni enoti (service rate).
- Naključna spremenljivka s_n ima eksponentno porazdelitev s parametrom μ, če je verjetnostna gostota funkcije podatna kot:

$$f_{s_n}(t) = \begin{cases} \mu e^{-\mu t} & \text{if } t \geq 0 \\ 0 & \text{if } t < 0 \end{cases} \qquad E[s_n] = \frac{1}{\mu}, \qquad Var[s_n] = \frac{1}{\mu^2}$$

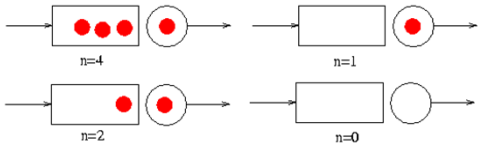
- Verjetnost, da bo čas strežbe manjši od časa *t* je:

$$F_{s_n}(t) = P(s_n \leq t) = \begin{cases} 1 - e^{-\mu t} & \text{if } t \geq 0 \\ 0 & \text{if } t < 0 \end{cases}$$

- μ predstavlja hitrost strežbe. Podaja število postreženih zahtev na časovno enoto neprestano obremenjenega strežnika.
- Verjetnost, da zahteva zapusti zaseden strežnik v intervalu *h* je *μh=konstanta*. Zahteve zapuščajo strežnik po Poissonovem procesu.

Sistemska stanja in verjetnosti

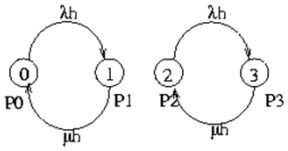
- Sistem z vrsto se nahaja v določenem stanju:
  - Stanje sistema M/M/1 določa število uporabnikov v sistemu – *n*
  - Sistem nima pomnilnika za pretekle dogodke



- Za vsako stanje obstaja verjetnost, da se sistem nahaja v njem: P_n.

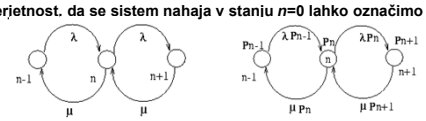
Diagram prehoda (Transition diagram)

- Pri Markovskih sistemih (sisteme s Poissonovimi prihodi in eksponentno porazdelitvijo časa strežbe) je verjetnost prehoda odvisna zgolj od trenutnega stanja.
  - Ni odvisna od tega, koliko časa se sistem nahaja v preteklem stanju.
- Za sisteme M/M/1 je v intervalu dolžine *h* konstanta:
  - Verjetnost prihoda zahtevka: *λh*
  - Verjetnost odhoda zahtevka - strežbe: *μh*



Prehod med stanji

- Delujoč Markovski sistem se premika iz stanja v stanje z določeno verjetnostjo.
- Po dolgem času se sistem ustali v stabilno stanje.
  - Verjetnost, da se sistem nahaja v stanju *n*, je konstantna (časovno neodvisna)
  - Verjetnost. da se sistem nahaja v stanju *n*=0 lahko označimo s P_0



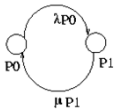
- Da zagotovimo stabilno stanje, morajo biti prehodi med vsakim parom stanj P_n in P_{n-1} med seboj uravnoteženi. Zapišemo lahko:

$$\lambda P_{n-1} = \mu P_n \rightarrow P_n = \frac{\lambda}{\mu} P_{n-1} = \rho P_{n-1}$$

Rekurzivna formula za prehod med stanji

- Zapišimo za n=1: $P_1 = \rho P_0$
- Zapišimo za splošen n>1: $P_2 = \rho P_1 = \rho^2 P_0$

$$P_n = \rho^n P_0$$

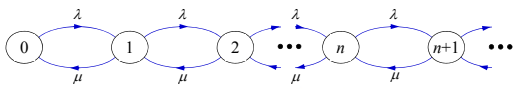


- Vsota verjetnosti je 1:

$$1 = \sum_{n=0}^{\infty} P_n = \sum_{n=0}^{\infty} \rho_n P_0 = P_0 \sum_{n=0}^{\infty} \rho_n = \frac{P_0}{1-\rho}$$

- Od tod dobimo splošno enačbo za P_n:

$$P_n = \rho^n (1-\rho)$$



Parametri zmogljivosti sistema

- Zanimajo nas parametri:
  - Povprečen čas čakanja
  - Povprečna zakasnitev pri prehodu zahteve preko sistema
  - Povprečno število čakajočih zahtev
- ρ: Izkoriščenost sistema (utilization, traffic intensity)

$$\rho = \frac{\lambda}{\mu}$$

  - Ko gre λ -> μ, gre ρ -> 1
- Verjetnost, da je v sistemu (vrsti in strežbi) *n* zahtev:

$$P_n = \rho^n (1-\rho)$$

  - geometrijsko zaporedje

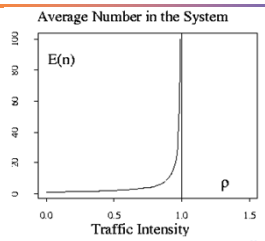
Povprečno število zahtev v sistemu

- L=E(n) predstavlja povprečje geometrijske porazdelitve:

$$L = E(n) = \sum_{n=0}^{\infty} nP_n = \sum_{n=0}^{\infty} n\rho^n(1-\rho) =$$
$$= (1-\rho)\sum_{n=0}^{\infty} n\rho^n = (1-\rho)\rho\sum_{n=0}^{\infty} n\rho^{n-1} =$$
$$= \rho(1-\rho)\frac{1}{(1-\rho)^2} = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu-\lambda}$$

- Narišimo graf odvisnosti *L(ρ)*

Povprečno število zahtev v sistemu



- Povprečno število zahtev v sistemu je večje od 0, tudi če je izkoriščenost sistema (ρ) relativno nizka.
- Povprečno število zahtev v sistemu raste preko vseh mej, če gre izkoriščenost sistema (ρ) proti 1, kljub temu, da še vedno velja λ < μ.
- Pogoj za stabilnost sistema M/M/1: ρ<1

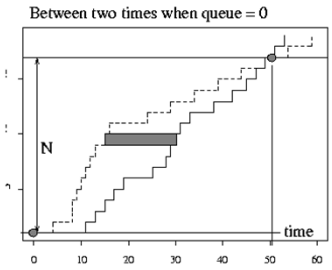
Littleov Teorem

- Littleov Teorem povezuje povprečen čas, ki ga potrebuje neko opravilo v kateremkoli sistemu z številom opravil, čakajočih na sistemu.
- Več o teoremu v:
  - Vignaux, G. A. (1997). *Analyzing Queues using cumulative graphs*. Technical report, School of Mathematical and Computing Sciences, Victoria University of Wellington.
  - Winston, W. L. (1994). *Operations Research, Applications and Algorithms*. Duxbury Press, 3rd edition.
- Izpeljava povprečnih vrednosti iz grafov kumulative prihajajočih in odhajajočih opravil.

Littleov Teorem: povprečni čas v sistemu

- Izračunajmo povprečni čas čakanja zahteve v sistemu z uporabo ploščine med presečišči krivulj.
- Povprečni čas zahteve v sistemu je W:

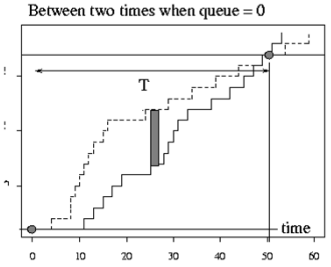
$$W = \frac{Ploščina}{N}$$



Littleov Teorem: povprečno št. v sistemu

- Izračunajmo povprečno število zahtev v sistemu z uporabo ploščine med presečišči krivulj.
- Povprečno število zahtev v sistemu je L:

$$L = \frac{Ploščina}{T}$$



Littleov Teorem

- Ploščina med presečišči krivulj je v obeh primerih enaka.

$$NW = TL \rightarrow L = \frac{N}{T}W$$

- ker je $\lambda = N/T$, dobimo: $\boxed{L = \lambda W}$ kjer je:
  - L= povprečno število zahtev na sistemu
  - λ = povprečno število prihodov zahtev
  - W = povprečni čas, ki ga zahteve porabijo na sistemu
- Videti je, da teorem velja za periodo od ene ničle do druge ničle.
  - Interval lahko raztegnemo od ničle do neskončnosti, in s tem zajamemo ničli; Littleov Teorem torej velja tudi za daljše periode za stabilne sisteme.

Littleov Teorem - pregled

- Littleov teorem velja ne glede na: $L = \lambda W$
  - Tip sistema in meje sistema
  - Število strežnikov
  - Karakteristike (naključnost, regularnost) prihodov v sistem
  - Karakteristike strežbe sistema
  - Pristop razvrščanja v vrste ozioma načina strežbe
- Littleov teorem ne velja, če se enote sistema izgubljajo

Povprečen čas zadrževanja zahteve v sist.

- W=E(w)
- Littleov teorem (Little's Theorem): $L = \lambda W$
- Z upoštevanjem teorema lahko zapišemo povprečen čas zadrževanju zahteve v sistemu:

$$W = \frac{L}{\lambda} = \frac{1}{\lambda}\frac{\lambda}{\mu - \lambda} = \frac{\rho}{\lambda(1-\rho)}$$

  - Ta čas je vsota časa čakanja v vrsti in časa strežbe

Povprečno število (čak.) zahtev v vrsti

- L_q = E(n) – E(število zahtev v strežbi)
- Povprečno število v strežbi ni 1, lahko ga izračunamo:

$$E(\text{v str}) = 0P_0 + 1P(>0)$$
$$E(\text{v str}) = 0P_0 + 1(1-P_0) = 1 - P_0 = \rho$$

- Iz česar sledi povprečno število zahtev v vrsti:

$$L_q = E(n) - \rho = \frac{\rho}{1-\rho} - \rho = \frac{\rho^2}{1-\rho}$$

Poprečen čas čakanja v vrsti

- Povprečen čas, ki ga zahteva porabi, preden opravi storitev.
- Od povprečnega časa zadrževanja odštejemo čas strežbe:

$$W_q = W - \frac{1}{\mu} = \frac{\rho^2}{\lambda(1-\rho)}$$

- Do istega rezultata lahko pridemo z uporabo Littleovega teorema:

$$W_q = \frac{L_q}{\lambda}$$

Čakanje na sistemu

- Verjetnost, da je v sistemu manj oz. ravno N paketov (P[Q≤N])

$$P[Q \le N] = \sum_{i=0}^{N} (1-\rho)\rho^i$$

- Verjetnost, da čas čakanja v čakalni vrsti ne preseže m_{Tq} sekund

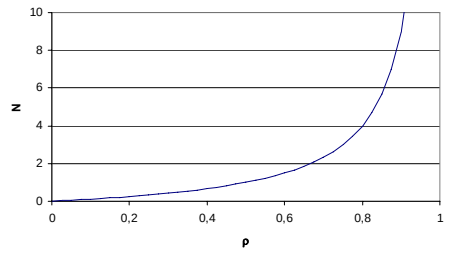
$$m_T(r) = T \ln\left(\frac{100}{100-r}\right)$$

HVALA ZA POZORNOST,
VPRAŠANJA !?

Strežni sistem M/M/1

Pogoj za stabilnost:

- **Povprečno število zahtevkov mora biti manjše od povprečnega števila postreženih**



Primeri

- **Avtomobili na avtocesti:**
  - **Kot primer si predstavljajmo avtomobile, ki vstopajo na avtocesto. Preverimo, ali ustrezajo pogojem za uporabo:**
    1. **Celotno število avtomobilov, ki uporablja avtocesto, je relativno veliko.**
    2. **Posamezni avtomobil uprorablja relativno majhen odstotek avtoceste.**
    3. **Odločitev posameznega avtomobila o vstopu na avtocesto je neodvisna od odločitev ostalih avtomobilov.**
  - **Zgornje predpostavke kažejo na to, da proces prihodov avtomobilov na avtocesto lahko aproksimiramo s Poissonovim procesom.**
  - **Če eden izmed navedenih pogojev ne bi bil izpolnjen, potem ne moremo privzeti Poissonovih prihodov. Če bi – denimo – na avtocesti imeli dirko avtomobilov, potem odločitev posameznika o vstopu na avtocesta ni neodvisna, temveč imajo vsi vozniki skupen cilj.**

- M/M/1 Queueing System
- We have already covered queueing theory basics in a previous article. In this article we will focus on M/M/1 queueing system. As we have seen earlier, M/M/1 refers to negative exponential arrivals and service times with a single server. This is the most widely used queueing system in analysis as pretty much everything is known about it. M/M/1 is a good approximation for a large number of queueing systems.
- The following topics will be discussed in detail:
- Poisson Arrivals Poisson Service Times Single Server M/M/1 Results
- Poisson Arrivals
- M/M/1 queueing systems assume a Poisson arrival process. This assumption is a very good approximation for arrival process in real systems that meet the following rules:
- The number of customers in the system is very large.
- Impact of a single customer on the performance of the system is very small, i.e. a single customer consumes a very small percentage of the system resources.
- All customers are independent, i.e. their decision to use the system are independent of other users.
- Cars on a Highway
- As you can see these assumptions are fairly general, so they apply to a large variety of systems. Lets consider the example of cars entering a highway. Lets see if the above rules are met.
- Total number of cars driving on the highway is very large.
- A single car uses a very small percentage of the highway resources.
- Decision to enter the highway is independently made by each car driver.
- The above observations mean that assuming a Poisson arrival process will be a good approximation of the car arrivals on the highway. If any one of the three conditions is not met, we cannot assume Poisson arrivals. For example, if a car rally is being conducted on a highway, we cannot assume that each car driver is independent of each other. In this case all cars had a common reason to enter the highway (start of the race).
- Telephony Arrivals
- Lets take another example. Consider arrival of telephone calls to a telephone exchange. Putting our rules to test we find:
- Total number of customers that are served by a telephone exchange is very large.
- A single telephone call takes a very small fraction of the systems resources.
- Decision to make a telephone call is independently made by each customer.
- Again, if all the rules are not met, we cannot assume telephone arrivals are Poisson. If the telephone exchange is a PABX catering to a few subscribers, the total number of customers is small, thus we cannot assume that rule 1 and 2 apply. If rule 1 and 2 do apply but telephone calls are being initiated due to some disaster, calls cannot be considered independent of each other. This violates rule 3.

- Poisson Service Times
- In an M/M/1 queueing system we assume that service times for customers are also negative exponentially distributed (i.e. generated by a Poisson process). Unfortunately, this assumption is not as general as the arrival time distribution. But it could still be a reasonable assumption when no other data is available about service times. Lets see a few examples:
- Telephone Call Durations
- Telephone call durations define the service time for utilization of various resources in a telephone exchange. Lets see if telephone call durations can be assumed to be negative exponentially distributed.
- Total number of customers that are served by a telephone exchange is very large.
- A single telephone call takes a very small fraction of the systems resources.
- Decision on how long to talk is independently made by each customer.
- From these rules it appears that negative exponential call hold times are a good fit. Intuitively, the probability of a customers making a very long call is very small. There is a high probability that a telephone call will be short. This matches with the observation that most telephony traffic consists of short duration calls. (The only problem with using the negative exponential distribution is that, it predicts a high probability of extremely short calls).
- This result can be generalized in all cases where user sessions are involved.
- Transmission Delays
- Lets see if we can assume negative exponential service times for messages being transmitted on a link. Since the service time on a link is directly proportional to the length of the message, the real question is that can we assume that message lengths in a protocol are negative exponentially distributed?
- As a first order approximation you can assume so. But message lengths aren't really independent of each other. Most communication protocols exchange messages in a certain sequence, the length distribution is determined by the length of the messages in the sequence. Thus we cannot assume that message lengths are independent. For example, internet traffic message lengths are not distributed in a negative exponential pattern. In fact, length distribution on the internet is bi-modal (i.e. has two distinct peaks). The first peak is around the length of a TCP ack message. The second peak is around the average length of a data packet.

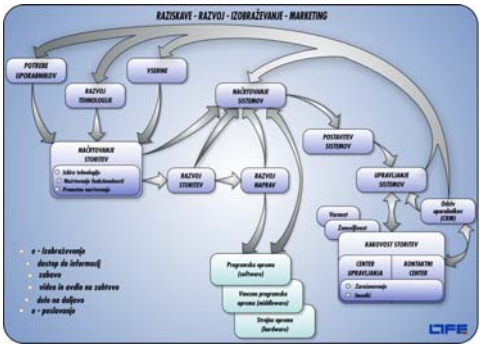
Modeliranje in simulacije telekomunikacijskih sistemov in omrežij

doc. dr. Iztok HUMAR
viš. pred. dr. Matevž PUSTIŠEK
prof. dr. Janez BEŠTER

Agenda

- Pristopi k ugotavljanju zmogljivosti
- Model, modeliranje
- Simulacija, simuliranje
- Orodja:
  - COMNET III
  - NS2
  - DES

Telekomunikacijski inženiring



Pristopi k ugotavljanju zmogljivosti

- Zmogljivosti
  - kapacitete
  - razpoložljivost
  - učinkovitost
  - raba oz. izkoriščenost virov
  - kakovost storitev
- Analiza realnih omrežij in sistemov
  - realni sistemi včasih (še) ne obstajajo
  - analize ni mogoče izvesti brez kvarnega posega v delovanje sistema
- Izdelava in analiza modelov omrežij in sistemov
  - modeli virov (ang. Workload generator, Source modelling, ...)
  - modeli omrežij in sistemov (ang. Network modeling)

Analiza realnih sistemov oz. modelov

- Analiza realnih sistemov
  - najučinkovitejše, velika natančnost
  - najzahtevnejše, najdražje
  - potek v realnem času
  - draga merilna oprema
  - zamudno odpravljanje napak
  - velikokrat neizvedljivo (npr. novi sistemi)
- Analitično modeliranje
  - izgradnja matematičnega modela
  - reševanje različnih vrst in sistemov enačb
  - računalnik le kot matematični pripomoček
  - rezultati
    - matematično natančni
    - realno približni, zaradi poenostavitev
    - v primeru velikih omrežij lahko rezultati neuporabni
- Simulacije
  - izdelava modela sistema
  - reševanje s pomočjo simulacijskih programov
  - časovni potek običajno ni realen
  - najprimernejša metoda za vrednotenje TK omrežij

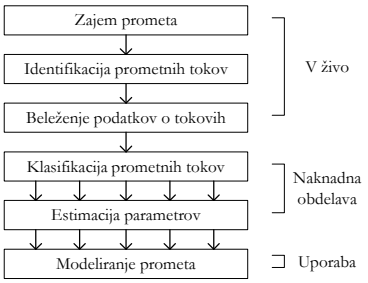
Model

- Model sistema je poenostavljena predstavitev sistema, ki analitiku omogoča razumevanje odziva sistema na spremembe
- Model:
  - mora biti čimbolj podoben realnemu sistemu
    - upoštevati je potrebno ključne lastnosti in parametre sistema
  - mora biti čimbolj enostaven
- Dober model je kompromis med realnostjo in enostavnostjo
- Model je potrebno preizkusiti
  - simulacije
  - primerjanje rezultatov

Modeliranje

- Modeliranje je proces izgradnje (matematičnega) modela
- Za simulacije se navadno uporabljajo matematični modeli
- Matematični modeli so:
  - deterministični (vhodni in izhodni podatki so določene vrednosti)
  - stohastični (vsaj ena izmed vhodnih/izhodnih spremenljivk je podana z verjetnostjo (običajno)
  - časovno nespremenljivi – statični
  - časovno spremenljivi – dinamični (običajno)

Primer: modeliranje TK prometa



Fokus modeliranja

- Različne vidike analiziranega sistema težko hkrati obravnavamo enako podrobno
- Časovna odvisnost
  - µs, ms, s, h, d, y
- Prostorska odvisnost
- Nivojska odvisnost
  - nivo seje oz. povezave
  - nivo PDU
  - fizični nivo
- Namen modeliranja mora biti znan vnaprej. En sam model, ki bi (enako) uspešno pokrival vse vidike, praviloma ne obstaja oz. je prekompleksen.

Verifikacija in validacija modelov

- Modeli morajo biti verodostojni (ang. credible)
  - verifikacija in validacija modelov
- Validacija: ugotavljanje primernosti
  - model in njegovo obnašanja sta primerna predstavitev realnega sistema in njegovega obnašanja (glede na namen modeliranja)
- Verifikacija: ugotavljanje pravilnosti
  - model je pravilno izdelan
- Primer: zemljevid
  - zemljevid je "model terena".
  - verifikacija potrdi, da je bil pravilno narisan.
  - validacija potrdi, da zemljevid ustreza obravnavanemu terenu
    - npr. da ga nismo obrnili na glavo,
    - da ne uporabljamo zemljevida napačnega kraja.

Vrste simulacij

- Zvezna simulacija
  - opis stanja sistema z algebrajskimi ali diferencialnimi enačbami
  - določajo spremembe na sistemu časovno zvezno, glede na dogodke
  - uporablja se jih predvsem za študijo povratnih vplivov (vpliv izhoda na vhod)
  - primer: letalski simulatorji, vremenske simulacije
- Diskretna simulacija
  - enostavnejše, manj natančne kot zvezne simulacije
  - opis stanja sistema z diskretnimi spremenljivkami
  - izvrševanje po kronološkem redu
  - sistem se spreminja kot odziv na diskretne dogodke
    - v sistemu s čakalno vrsto M/M/1 se sistem spremeni s prihodom novega paketa
  - primer: simulacije v telekomunikacijah

Glede na čas

- Sinhrone simulacije
  - iteracija - numerične metode
  - za izhodišče vzamemo matematične modele
  - čas povečujemo po vnaprej določenem koraku δt
  - natančnost lahko spreminjamo z velikostjo časovnega koraka
  - primerne za simulacijo prevajanja toplote, pretakanja tekočin, dogajanj v atmosferi ...
- Asinhrona simulacija (Discrete Event)
  - tipično vsebuje naslednje faze:
    - generiranje dogodka:
      - asinhrono, npr. oddaja paketa na linijo
      - simulacijski čas se poveča, računalnik ni obremenjen
    - simulacija odziva omrežja:
      - sinhrona, npr. obdelava glave paketa
      - simulacijski čas stoji, računalnik je obremenjen
  - časovni korak je lahko poljuben, ni določen vnaprej
  - natančnost lahko poljubno spreminjamo, odvisna je od natančnosti opisa modela

Simuliranje

- Simuliranje: uporaba modela nekega sistema
  - model lahko spreminjamo
  - sistema ne moremo (ni praktično) spreminjati
- Preučujemo lastnosti, obnašanje dejanskega sistema glede na parametre
- Simulacija se uporablja za vrednotenje zmogljivosti različnih konfiguracij sistema

Kdaj je problem primeren za simulacijo?

- Kadar se pojavi potreba po modeliranju in analizi naključnosti v sistemu, moremo uporabiti simulacijo.
- Primeri, ko uporabimo simulacije in modeliranje, so naslednji:
  - kadar je nemogoče ali izjemno drago opazovati nek proces v realnem svetu:
    - vpliv povečanja prometa na obremenjenost komunikacijskih poti
    - statistika raka v prihajajočem letu
    - zmogljivost prihajajočih satelitov
    - učinek oglaševanja na Internetu na prodajo v podjetju
  - problemi, ki jih je mogoče opisati z matematičnim modelom, pa vendar so analitično nerešljivi:
    - problem trgovskega potnika, diferenčne enačbe višjih stopenj, urnik

    ali pa težko rešljivi:
    - modeli večjih sistemov s čakalnimi vrstami, borzni trg
  - kadar je nemogoče ali izjemno drago preveriti matematične modele, ki opisuje določen sistem (npr. zaradi pomanjkanja podatkov)

Namen simulacij

- Simulacije izvajamo:
  - pred izgradnjo sistema
  - pred spreminjanjem/nadgradnjo sistema
- S simulacijami:
  - zmanjšamo možnost napak
  - preprečimo nepredvidene zamašitve v omrežju
  - preprečimo pre/podobremenjenost omrežnih virov
  - optimiziramo zmogljivosti sistema
- S simulacijami si odgovorimo na vprašanja, kot npr.:
  - kako najbolj optimalno izgraditi novo omrežje?
  - kakšne omrežne zmogljivosti potrebujemo?
  - kako se bo omrežje odzvalo, če promet povečamo za 50%?
  - kako bo nov usmerjevalni algoritem vplival na zmogljivost omrežja?
  - kako optimizirati zmogljivost omrežja?
  - vpliv okvare na povezavi na delovanje omrežja.

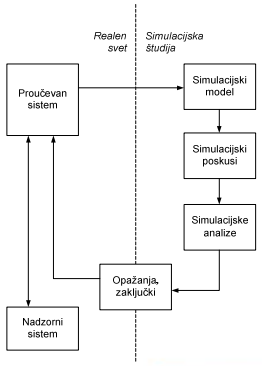
Simulacijska orodja se uporabljajo pri:

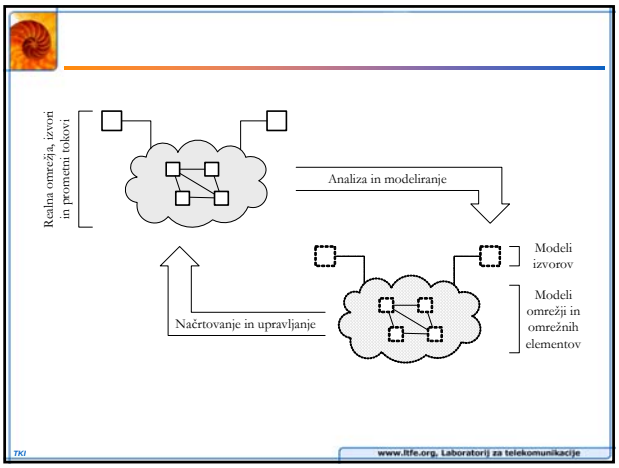
- Telekomunikacijskih sistemih in v računalniških omrežjih
- Proizvodnji
- Prometu (letalski, cestni promet)
- Strežbi (banke, trgovine)
- Zdravstvu
- Ekologiji
- Sociologiji
- Epidemiologiji
- Ekonomiji in poslovnih analizah

Koraki simulacijske študije

1. Identificiramo problem
2. Izoblikujemo problem
3. Zberemo in obdelamo podatke iz realnega sistema
4. Izoblikujemo in razvijemo model
5. Raziščemo veljavnost modela
6. Dokumentiramo model za kasnejšo uporabo
7. Izberemo primeren načrt za preizkus
8. Vzpostavimo primerne pogoje za izvedbo preizkusa
9. Izvedemo simulacijo
10. Interepretiramo in predstavimo rezultate
11. Predlagamo nadaljnje pristope/korake

Shema simulacijske študije





Zaključki na podlagi simulacij

- določitev problematičnih/poddimenzioniranih elementov v omrežju
- spoznavanje situacij v primeru napak, spremembi protokola
- predlog za optimizacijo oz. nadgraditev
  - zvišanje izkoriščenosti omrežja
  - zmanjšanje zakasnitev pri prenosu
  - povečanje zanesljivosti omrežja

Simulacije: prednosti

- Modeliranje, simulacije in analiza se zelo pogosto uporabljajo pri operacijskih raziskavah
- Omogočajo:
  - boljše razumevanje sistema
  - testiranje hipotez o sistemu
  - hitrejše opazovanje počasnih fenomenov in počasnejše opazovanje hitrih fenomenov
  - preizkušanje novih, neznanih situacij
  - identifikacija najbolj pomembnih parametrov v sistemu
  - identifikacija ozkih grl
  - omogočajo razvoj dobro načrtovanega in robustnega sistema
  - skrajšajo čas načrtovanja

Simulacije: slabosti

- Simulacije so časovno potratne in kompliciranе naloge
- Pri načrtovanju modela in analizi rezultatov mora so potrebne strokovne izkušnje
- Pri simulacijah lahko naletimo na naslednje težave:
  - nejasno definirani cilji
  - uporaba simulacij, kadar obstaja analitična rešitev
  - napačno izdelan model
  - simulacijski model je preveč kompleksen ali preveč preprost
  - napačne predpostavke
  - nedokumentirane predpostavke
  - uporaba napačnih verjetnostnih porazdelitev za vhodne podatke
  - nadomestitev stohastične porazdelitve s povprečno vrednostjo
  - uporaba napačnih meritev zmogljivosti
  - napake v simulacijskem programu
  - uporaba napačnih statističnih metod/enačb

Drugi načini analize zmogljivosti

- Emulacije
  - souporaba modelov in realnih sistemov
- Tomografija
  - aktivno merjenje

Simulacijskega orodja

- Model je lahko zgrajen v kateremkoli programskem jeziku (splošno)
- Simulacijski paketi:
  - potrebno manj programerskega dela
  - okvir za modeliranje in simuliranje
  - avtomatsko zbirajo statistične podatke
  - grafični simboli za vnos modela
  - animacije
- Simulacijski jeziki
  - bolj prilagodljivi
  - zahtevajo več programerskega znanja
- Aplikacijsko orientirani simulatorji
  - enostavnejši za učenje
  - vsebujejo gradnike modelov, ki so bližje aplikacijam
- Orodja za naknadno obdelavo rezultatov

Tip simulacijskega paketa	Primer
Simulacijski jeziki	Ns2, Arena, AweSim!, Extend, GPSS, Micro Saint, SIMSCRIPT, SLX
Aplikacijsko orientirani simulator	COMNET III, NETWORK II.5, OPNET Modeler, OPNET Planner, SES/Strategizer

Primeri simulacijskih orodij

- CACI COMNET III
- The Network Simulator: NS2
  - http://isi.edu/nsnam/ns/
- DESSP
  - prototip diskretnega paketnega simulatorja
- Cisco Packet Tracer
  - omrežni simulator, podpira vključevanje modelov resničnih omrežnih elementov
- GNS3 – Graphical Network Simulator
  - podpira vključevanje modelov resničnih omrežnih elementov
  - http://www.gns3.net/
- OPNET Modeler
  - načrtovanje in analiza omrežij, naprav, protokolov in aplikacij
  - http://www.opnet.com/solutions/network_rd/modeler.html

Simulator CACI COMNET III

Uvod v COMNET III

- Uporaba in značilnosti
  - deluje v Windows okolju
  - na voljo obsežna dokumentacija
  - preprosta uporaba s intuitivnim uporabniškim vmesnikom
  - uporabnost predvsem v simulaciji manjših in srednje velikih mrež s poudarkom na testiranju obremenjenosti povezav in morebitne nadaljnje širitve uporabnikov
  - tekstoven in grafičen prikaz rezultatov, možnost nadaljnje obdelave rezultatov v drugih programih

Uporabnost COMNET III

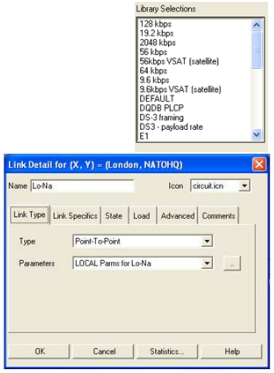
- Najpogostejši primeri simuliranja
  - študije največje obremenjenosti omrežja
  - upoštevanje širjenja omrežja v fazi načrtovanja
  - testiranje redundance omrežja v primeru odpovedi raznih elementov (časovno proženje dogodkov)
  - predvidevanje novih uporabnikov/aplikacij v obstoječem omrežju in odpravljanje težav preden se pojavijo
  - simulacija nadgradnje obstoječega omrežja ter študija upravičenosti investicije

Elementi simulatorja

- Povezave
  - CSMA/CD, Ethernet, TokenRing, ISDN, FDDI, Aloha, SONET, satelitska povezava
- Omrežja
  - podomrežja, prehodna omrežja, WAN oblaki
- Vozlišča
  - procesno vozlišče, skupina računalnikov, omrežni elementi
- Izvori prometa
  - izvor sporočil in seje sporočil, aplikacijski izvor in odgovor, klicni izvor, zunanji izvori prometa (Baseliner)
- Prikaz rezultatov
  - grafičen prikaz, tekstoven prikaz, izvoz v Excel za nadaljnjo obdelavo

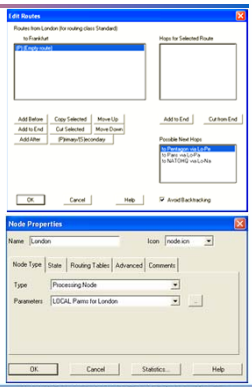
Posamezni elementi - povezave

- Možne nastavitve
  - tip povezav (Aloha, ethernet, CSMA, satelitska PPP, modem, ...)
  - lastnosti povezav (propagacijska zakasnitev, verjetnost in vrsta napak, dinamičnost dogodkov na povezavi)
  - nastavljanje specifičnih lastnosti posameznega tipa povezave



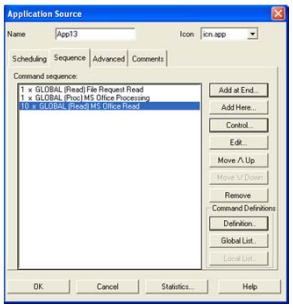
Posamezni elementi - vozlišča

- Možne nastavitve
  - vrsta vozlišča (Processing node, Computer Group, HOL Blocking Switch, Network Device)
  - lastnosti vozlišč (število procesorjev, procesni cikli, zakasnitve na vratih, čas procesiranja posameznega paketa)
  - dinamičnost agenta (čas napake in ponovne vzpostavitve, zaporedje dogodkov)
  - ročna izdelava usmerjevalne tabele



Posamezni elementi – izvori prometa

- Vrste prometnih izvorov
  - izvor sporočil in seje sporočil:
    - porazdelitvena funkcija prihajanja sporočil, velikost, izvor in cilj, prenosni protokol (UDP, TCP, ...), zakasnitev pri tvorbi sporočil
  - odgovori na sporočila:
    - velikost, prenosni protokol (UDP, TCP, ...), zakasnitev pri tvorbi sporočil
  - aplikacijski izvor:
    - predstavlja kompleksnejši izvor prometa s že vnaprej pripravljenimi modeli najpogosteje uporabljenih aplikacij (File Read / Write, MS Office Processing / Wait, ...)
    - število ponovitev aplikacijskih akcij
    - kompleksnejši aplikacijski konstrukti s pogojnimi stavki (IF-THEN-ELSE, WHILE, REPEAT-UNTIL)
  - klicni izvori / uvažanje zunanjega prometa v simulator (Baseliner)



Posamezni elementi - podomrežja

- Vrste podomrežij
  - prehodna omrežja,
  - podomrežja,
  - network clouds

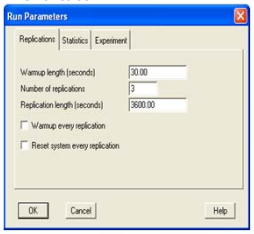
  - prehodno omrežje predstavlja abstrakten model, ki se lahko obnaša kot navadna povezava ali kot podomrežje.
  - v njih lahko nastavljamo lastne usmerjevalne algoritme, neodvisno od ostalega omrežja.

  - podomrežja lahko uporabimo zaradi večje preglednosti nad simulacijskim scenarijem
  - podomrežjem določimo drugačne lastnosti od preostalega omrežja
  - sem spadajo predvsem usmerjevalni algoritmi z vsemi pripadajočimi nastavitvami.

  - WAN oblak predstavlja abstrakcijo pri modeliranju WAN storitev v smislu dostopnih točk in navideznih povezav.
  - ključna nastavitev je kontrola zamašitve (congestion control) in definicija parametrov, kamor spadajo Frame Relay, ATM, X.25, IP, popoln časoven nadzor nad zgostitvami.

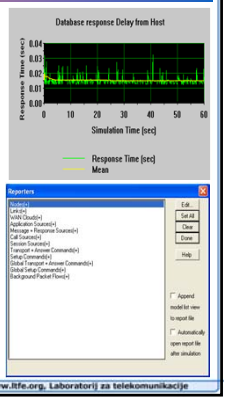
Globalni parametri simulacije

- Nastavitev usmerjevalnega algoritma
  - nastavitev velja za celotno omrežje, razen če za podomrežja določimo drugačne parametre
- Vsi parametri agentov, povezav, aplikacij in protokolov
- Dolžina simulacije, oblika izhodnih rezultatov
  - na katerih elementih bi radi zajemali izhodne podatke
  - v kakšni obliki bi zajete podatke radi prikazali



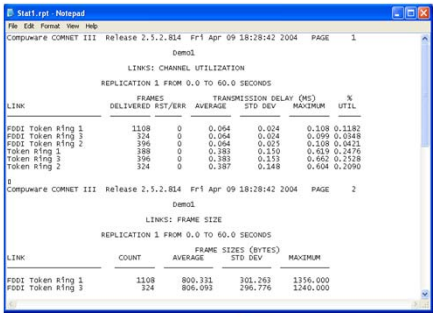
Rezultati simulacije (I)

- Oblika izpisa rezultata
  - tekstovna datoteka
  - Excelov format
  - neposreden izris grafa med simulacijo
- Katere podatke lahko opazujemo?
  - na vozliščih, povezavah, agentih, ...
  - obremenjenost povezave:
    - zaradi aplikacije
    - glede na protokole
  - kolizije
  - število uspešnih / zavrnjenih klicev
  - obremenjenost procesorskih in pomnilniških zmogljivosti vozlišča, ...



Rezultati simulacije (II)

- Primer tekstovnega izpisa rezultatov simulacije

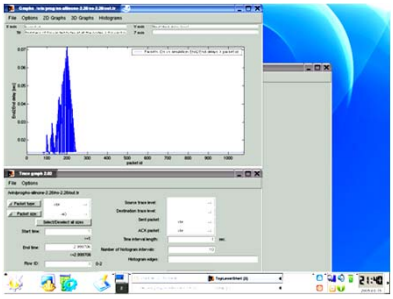


Pristop k simulaciji

- Gradnja topologije
  - postavitev izvorov prometa, agentov, vozlišč, povezav, ponorov, ...
- Nastavitev globalnih parametrov
  - čas trajanja, število ponovitev, prenosni protokoli, usmerjevalni algoritmi, ...
- Verifikacija modela
  - ali je model pravilen?
  - so vsi elementi povezani med sabo?
- Nastavitve izpisa rezultatov
  - omejimo se samo na izpis rezultatov, ki nas zanimajo
- Interpretacija rezultatov
  - grafičen prikaz rezultatov
  - primerjava z drugimi (enostavnejšimi in lažje razumljivimi modeli) omogoča lažjo interpretacijo

Pregled rezultatov s ostalimi programi

- Excel
- Matlab
- Tracegraph
- Xgraph



Simulator paketnih omrežij NS2

Vsebina

- NS2, simulator TK omrežij
- Struktura simulatorja
- Osnovni postopki za opis scenarija
- Struktura posameznih objektov
- Razno

NS2, simulator TK omrežij

- Uporablja metodo simulacije diskretnih dogodkov
- Napisan je bil predvsem zaradi simulacije IP omrežij, uporaben pa je tudi za študije drugih vključenih omrežij
- Podpira različne omrežne protokole, usmerjanja, povezave in promet
  - TCP, UDP
  - multicast, unicast
  - žične povezave, brezžične povezave, satelitske povezave
  - cbr, web, telnet, ftp
- Razvit leta 1989 kot različica simulatorja “Real network simulator”

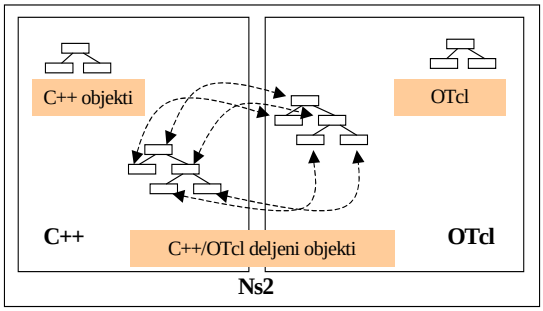
Komponente simulatorja NS2

- NS – osnovna struktura simulatorja
- Orodja za prikaz simulacije
  - Nam - Network AniMator
    - za prikaz ožičenih simulacij
    - GUI za vnos enostavnih scenarijev
  - Ad-Hockey
    - za prikaz brezžičnih simulacij
- Pred-procesiranje
  - generatorji prometa in topologije
- Post- procesiranje
  - enostavne analize sledilnih datotek (Awk, Perl, Tcl)

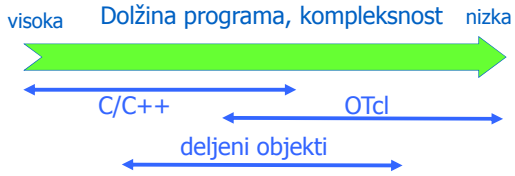
Programska struktura simulatorja NS2

- Uporablja dva programska jezika: C++ in OTcl
  - C++
    - natančen opis protokolov
    - hitro izvajanje izračunov
  - OTcl
    - načrtovanje topologije
    - primeren za hitre spremembe scenarijev
- +
  - kompromis med sestavljanjem scenarija in hitrost izvajanja
- -
  - razumevanje dveh jezikov in odpravljanje napak

Dualnost C++ in OTcl

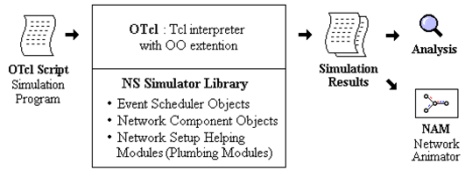


Primerjava kompleksnosti



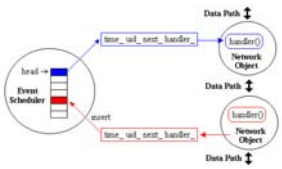
Potek simuliranja v NS2

- Topologijo omrežja opišemo v OTcl skripti
  - določitev vozlišč, povezav, izvorov prometa
- Postavimo ukaze za nadzor dogajanja
  - sledilna datoteka (trace file)
  - nadzor pretoka (število prispelih, zavrženih, odhodnih paketov)
- Podati moramo časovni razpored dogodkov
- Zagon simulacije in uporaba rezultatov



Odpravnik dogodkov

- Postavitev odpravnika dogodkov
  - set ns [new Simulator]
- Razvrščanje dogodkov
  - $ns at <čas> <dogodek>
  - <dogodek>: ustrezen ns/tcl ukaz
  - primer:
  - $ns at 0.1 "$cbr start"
  - $ns at 4.0 "$ftp stop"
  - primer prekinitve linije:
  - $ns rtmodel-at 1.0 down $vozlišče1 $vozlišče2
  - $ns rtmodel-at 2.0 up $vozlišče1 $vozlišče2
- Zagon odpravnika dogodkov
  - $ns run



Postavitev omrežja

- Vozlišča
  - set n0 [$ns node]
  - set n1 [$ns node]
- Povezave in čakalne vrste
  - $ns duplex-link $n0 $n1 <pasovna širina> <zakasnitev> <čakalna vrsta>
  - <čakalna vrsta>: DropTail, RED, CBQ, FQ, SFQ, DRR
- primer:
  - $ns duplex-link $n0 $n2 2Mb 10ms DropTail
  - $ns duplex-link $n1 $n2 1Mb 5ms RED

Opis usmerjanja prometa

- Enouporabniško usmerjanje
  - $ns rtproto <vrsta usmerjanja>
  - < vrsta usmerjanja >: Static, Session, DV, cost, multi-path
- Večuporabniško usmerjanje
  - set ns [new Simulator –multicast on] ali
  - set ns [new Simulator]
    $ns multicast
  - $ns mrtproto < vrsta usmerjanja >
  - < vrsta usmerjanja >: CtrMcast, DM, ST, BST

Postavitev prometa

- Promet določimo na dveh nivojih:
  - transportnem
  - aplikacijskem
- Transportni nivo
  - TCP, UDP
- Aplikacijski nivo
  - cbr, ftp, telnet
- UDP
  - izvor in ponor
    - set udp [new Agent/UDP]
    - set sink [new Agent/Null]
  - povezava izvora in ponora na vozlišče
    - $ns attach-agent $n0 $udp
    - $ns attach-agent $n1 $sink
    - $ns connect $udp $sink
- TCP
  - izvor in ponor
    - set tcp [new Agent/TCP]
    - set sink [new Agent/TCPSink]
  - povezava izvora in ponora na vozlišče
    - $ns attach-agent $n0 $tcp
    - $ns attach-agent $n1 $sink
    - $ns connect $tcp $sink

Postavitev prometa

- Nad TCP agentom
  - FTP
    - set ftp [new Application/FTP]
    - $ftp attach-agent $udp
    - $ns at <time> "$ftp start"
  - Telnet
    - set telnet [new Application/Telnet]
    - $telnet attach-agent $udp
- Nad UDP agentom
  - CBR
    - set cbr [new Application/Traffic/CBR]
  - Exponential or Pareto on-off
    - set expo [new Application/Traffic/Exponential]
    - set expo [new Application/Traffic/Pareto
- Primer postavitve in nastavljanja CBR prometa

```
set udp0 [new Agent/UDP]
$ns attach-agent $node0 $udp0
set null0 [new Agent/Null]
$ns attach-agent $node1 $null0
set cbr [new
Application/Traffic/CBR]
$cbr set packetSize_ 512
$cbr set interval_ 4.0
$cbr set random_ 1
$cbr set maxpkts_ 10000
$cbr attach-agent $udp0
$ns connect $udp0 $null0
$ns at 127.45 "$cbr start"
```


Vstavljanje napak

- Postavitev modula za napake
  - set loss_module [new ErrorModel]
  - $loss_module set rate_ 0.01
  - $loss_module unit pkt
  - $loss_module ranvar [new RandomVariable/Uniform]
  - $loss_module drop-target [new Agent/Null]
- Vstavljanje modula za napake
  - $ns lossmodel $loss_module $n0 $n1

Slednje prometu

- Slednje paketov s sledilno datoteko tr.out
- $ns trace-all [open tr.out w]

event	time	from node	to node	pkt type	pkt size	flags	fid	src addr	dst addr	seq num	pkt id

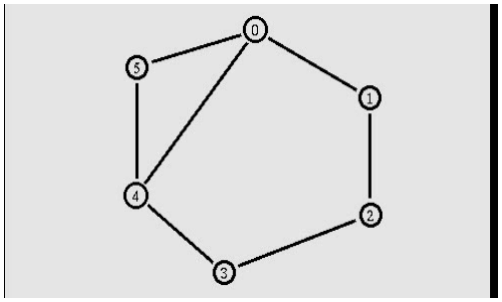
```
r : receive  (at to_node)                src_addr : node.port (3.0)
+ : enqueue  (at queue)                  dst_addr : node.port (0.0)
- : dequeue  (at queue)
d : drop     (at queue)
```

```
r 1.3556 3 2 ack 40 ------- 1 3.0 0.0 15 201
+ 1.3556 2 0 ack 40 ------- 1 3.0 0.0 15 201
- 1.3556 2 0 ack 40 ------- 1 3.0 0.0 15 201
r 1.35576 0 2 tcp 1000 ------- 1 0.0 3.0 29 199
+ 1.35576 2 3 tcp 1000 ------- 1 0.0 3.0 29 199
d 1.35576 2 3 tcp 1000 ------- 1 0.0 3.0 29 199
+ 1.356 1 2 cbr 1000 ------- 2 1.0 3.1 157 207
- 1.356 1 2 cbr 1000 ------- 2 1.0 3.1 157 207
```


Uporaba rezultatov

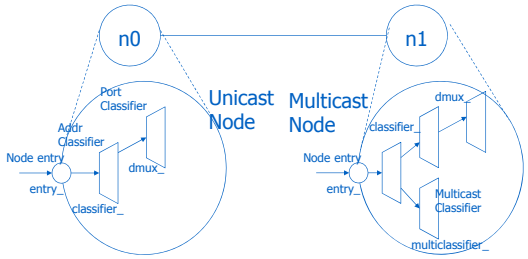
- Sledenje paketov z Nam-om
- $ns namtrace-all [open tr.nam w]

Primer NAM animacije (1/2)

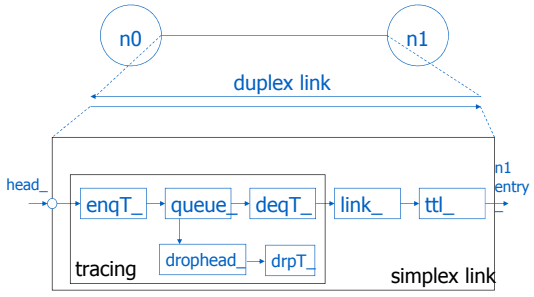


Primer NAM animacije (2/2)

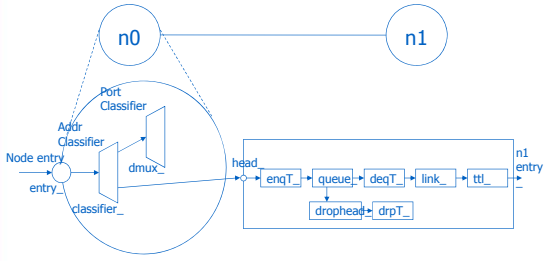
Omrežna toplogija - vozlišče



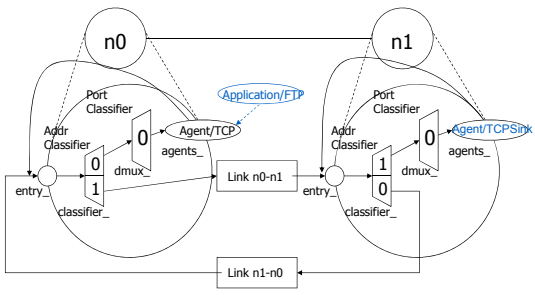
Omrežna toplogija - povezava



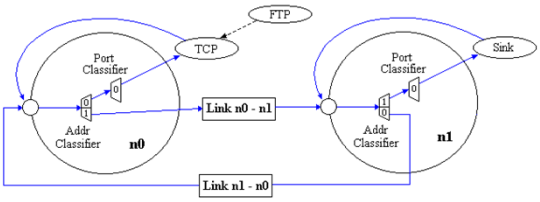
Povezovanje omrežnih elementov



Povezovanje objektov



Pretok paketov



Instalacija in dokumentacija NS2

- http://www.isi.edu/nsnam/ns/
  - prenos datoteke “ns-allinone”
  - Vsebuje celoten sklop: Tcl, OTcl, TclCL, Ns, Nam, itd.
- mailling lista: ns-users@isi.edu
- dokumetacija (glej zgornji naslov)
  - Marc Gries tutorial
  - Ns dokumentacija

Primerjava: COMNET III in NS2

- Lastnosti COMNET
  - enostaven
  - celotno upravljanje z uporabo grafičnega uporabniškega vmesnika
  - že zgrajeni modeli realnih (komercialno dostopnih) elementov (3Com stikala, ...)
  - vsebuje skoraj vse potrebne elemente, protokole, usmerjevalne algoritme
  - možnost prilagajanja aplikacijskih izvorov s kompleksnejšimi konstrukti
- Lastnosti NS2
  - brezplačen
  - popoln vpogled v izvorno kodo
  - možnost spreminjanja / dodajanja poljubnih elementov
  - neomejene možnosti pri simulacijah
  - potrebno veliko znanja in zelo podrobno poznavanje celotne izvorne kode simulatorja

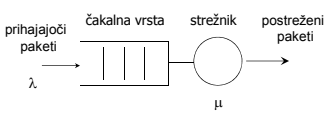
Simulator diskretnih dogodkov DES

Simulator diskretnih dogodkov (DES)

- Konfiguracija oz. sestavljanje modelov
- Izvajanje simulacij (ang. Engine)
- Zajem rezultatov
  - sledilne datoteke
- Post-obdelava rezultatov
  - statistična analiza, ...
- Prikaz rezultatov
  - tekstovno, grafično, animacije

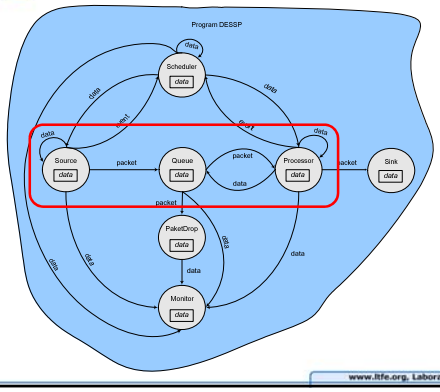
Zasnova modela

- Simulacija sistema M/D/1

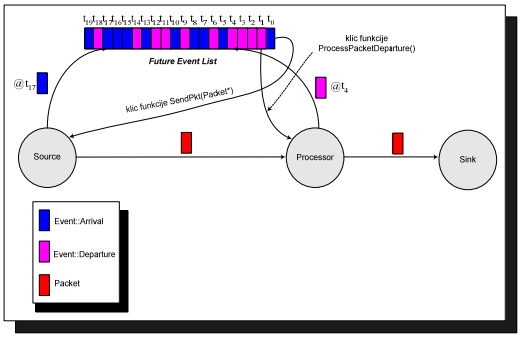


Sistem M/D/1

Objektni model simulatorja DESSP



Simulator DESSP – delovanje



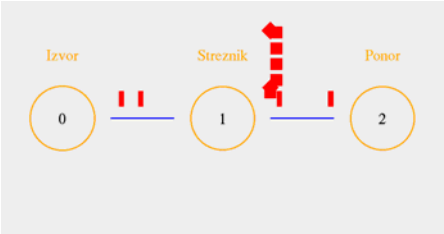
Zajem rezultatov

- Sledilne (ang. trace) datoteke

```
+  1.84375  0  2  cbr   210   -------  0  0.0  3.1   225   610
-  1.84375  0  2  cbr   210   -------  0  0.0  3.1   225   610
r  1.84471  2  1  cbr   210   -------  1  3.0  1.0   195   600
r  1.84566  2  0  ack   40    -------  2  3.2  0.1   82    602
+  1.84566  0  2  tcp   1000  -------  2  0.1  3.2   102   611
```


Prikaz rezultatov - animacija

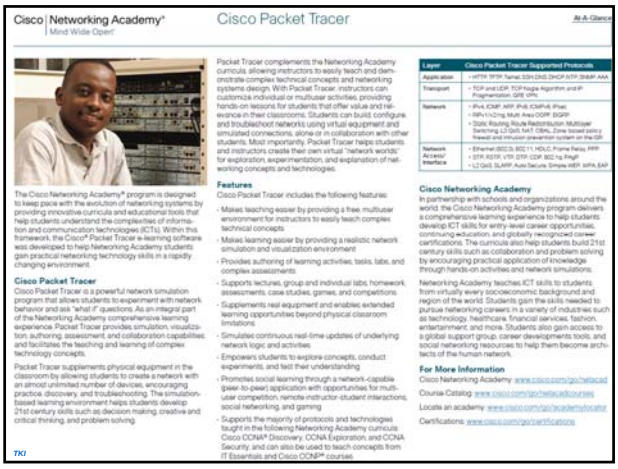
- Omrežni animator - NAM



Cisco Packet Tracer

Packet Tracer

- http://www.cisco.com/web/learning/netacad/course_catalog/PacketTracer.html
- http://www.packettracertrivia.com/
- http://www.cisco.com/web/learning/netacad/packet_tracer/packet_tracer_03-4_web.html



Cisco | Networking Academy®
Mind Wide Open®

Cisco Packet Tracer

At-A-Glance

Packet Tracer complements the Networking Academy curricula, allowing instructors to easily teach and demonstrate complex technical concepts and networking systems design. With Packet Tracer, instructors can customize individual or multiuser activities, providing hands-on lessons for students that offer value and relevance in their classrooms. Students can build, configure, and troubleshoot networks using virtual equipment and simulated connections, alone or in collaboration with other students. Most importantly, Packet Tracer helps students and instructors create their own virtual "network worlds" for exploration, experimentation, and explanation of networking concepts and technologies.

Layer	Cisco Packet Tracer Supported Protocols
Application	• HTTP, TFTP, Telnet, SSH, DNS, DHCP, NTP, SNMP, AAA
Transport	• TCP and UDP, TCP Nagle Algorithm and IP Fragmentation, GRE, VPN
Network	• IPv4, ICMP, ARP, IPv6, ICMPv6, IPsec • RIPv1/v2/mg, Multi Area OSPF, EIGRP • Static Routing, Route Redistribution, Multilayer Switching, L3 QoS, NAT, OBAL, Zone based policy firewall and intrusion prevention system on the ISR
Network Access/ Interface	• Ethernet (802.3, 802.11, HDLC, Frame Relay, PPP • STP, RSTP, VTP, DTP, CDP, 802.1q, PAgP • L2 QoS, SLARP, Auto Secure, Simple WEP, WPA, EAP

Cisco Packet Tracer

Cisco Packet Tracer is a powerful network simulation program that allows students to experiment with network behavior and ask "what if" questions. As an integral part of the Networking Academy comprehensive learning experience, Packet Tracer provides simulation, visualization, authoring, assessment, and collaboration capabilities and facilitates the teaching and learning of complex technology concepts.

Packet Tracer supplements physical equipment in the classroom by allowing students to create a network with an almost unlimited number of devices, encouraging practice, discovery, and troubleshooting. The simulation-based learning environment helps students develop 21st century skills such as decision making, creative and critical thinking, and problem solving.

Features

Cisco Packet Tracer includes the following features:

- Makes teaching easier by providing a free, multiuser environment for instructors to easily teach complex technical concepts
- Makes learning easier by providing a realistic network simulation and visualization environment
- Provides authoring of learning activities, tasks, labs, and complex assessments
- Supports lectures, group and individual labs, homework assessments, case studies, games, and competitions
- Supplements real equipment and enables extended learning opportunities beyond physical classroom limitations
- Simulates continuous real-time updates of underlying network logic and activities
- Empowers students to explore concepts, conduct experiments, and test their understanding
- Promotes social learning through a network-capable (peer-to-peer) application with opportunities for multi-user competition, remote instructor-student interactions, social networking, and gaming
- Supports the majority of protocols and technologies taught in the following Networking Academy curricula: Cisco CCNA® Discovery, CCNA Exploration, and CCNA Security, and can also be used to teach concepts from IT Essentials and Cisco CCNP® courses

Cisco Networking Academy

In partnership with schools and organizations around the world, the Cisco Networking Academy program delivers a comprehensive learning experience to help students develop ICT skills for entry-level career opportunities, continuing education, and globally recognized career certifications. The curricula also help students build 21st century skills such as collaboration and problem solving by encouraging practical application of knowledge through hands-on activities and network simulations.

Networking Academy teaches ICT skills to students from virtually every socioeconomic background and region of the world. Students gain the skills needed to pursue networking careers in a variety of industries such as technology, healthcare, financial services, fashion, entertainment, and more. Students also gain access to a global support group, career development tools, and social networking resources to help them become architects of the human network.

For More Information

Cisco Networking Academy: www.cisco.com/go/netacad

Course Catalog: www.cisco.com/go/netacadcourses

Locate an academy: www.cisco.com/go/academylocator

Certifications: www.cisco.com/go/certifications

TKI

Teme za seminarsko nalogo

- **Comnet III**
  - **Simulacija ADSL omrežja**
- **NS2**
  - **Mehanizmi QoS, multicast video odjemalci, zajem realnega prometa, VoIP, brezžična omrežja, ...**
- **Orodja za prikaz rezultatov simulacije in njihova interpretacija**

Laboratorij za telekomunikacije
Fakulteta za elektrotehniko

Hvala za pozornost.
Vprašanja?

QoS v sistemih IP

as.mag. Iztok HUMAR
mag. Mitja GOLJA
prof.dr. Janez BEŠTER

Potreba po mehanizmih za QoS

- Internet ponuja en sam razred storitev: best-effort razred
  - ni zagotovila, da bodo datagrami:
    - dostavljeni pravočasno
    - sploh dostavljeni
- Danes v internetu najpogosteje uporabljane aplikacije (http, ftp, smtp, ...) so elastične aplikacije
  - dopuščajo zakasnitve in izgube datagramov
  - so se zmožne prilagajati trenutnim zamašitvam v omrežju
- Nekatere prihajajoče aplikacije (VoIP, videokonference, ...) so neelastične aplikacije
  - zahtevajo prenos podatkov v realnem času
  - izguba datagramov ali njihova zakasnitev pomeni prekinitev delovanja

Agenda

- Problematika kvalitete v omrežjih IP
  - potreba po mehanizmih za zagotavljanje QoS
  - neelastične aplikacije
- Temeljni mehanizmi za izvajanje QoS
- IntServ
- DiffServ
- Evaluacija mehanizmov za QoS v sintetičnih omrežjih

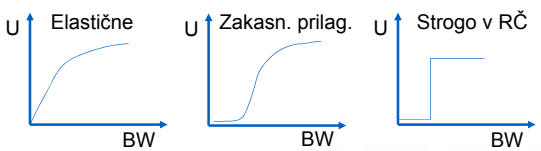
Neelastične aplikacije

- Aplikacije pretočnega medija
  - zgornja in spodnja meja sprejemljive zmogljivosti omrežja
    - bitna hitrost, pod katero video oziroma audio nista več uporabna
    - velike zakasnitve (200 – 300 ms) v primeru telefonije ali videokonferenc kvarno vplivajo na komunikacijo med ljudmi
  - Dolgotrajne seje teh aplikacij
- Aplikacije, ki delujejo strogo v realnem času
  - imajo stroge meje – kar se tiče zmogljivosti omrežja
  - primer: krmilne aplikacije

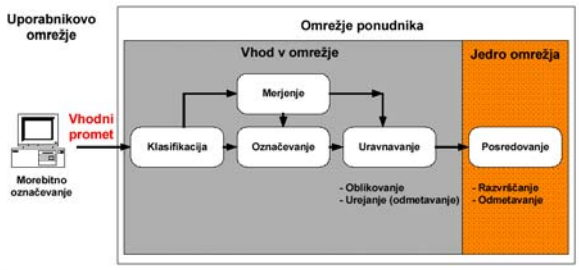
Problematika kvalitete v omrežjih IP

Stopnja koristnosti aplikacije

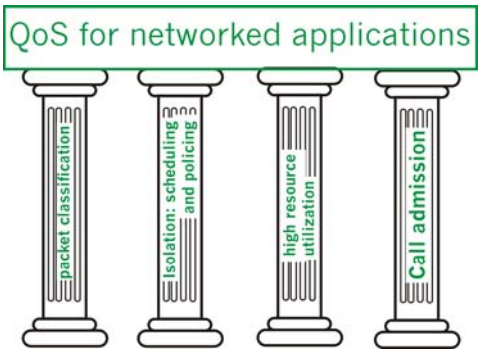
- Kaj je ključnega pomena pri načrtovanju omrežja?
  - povečevanje pasovne širine omrežja BW?
  - zmanjševanje zakasnitev v omrežju?
  - povečevanje zadovoljstva uporabnika -> koristnost, usability
- Krivulja odvisnosti U od BW:
  - monotono rastoča funkcija
  - oblika odvisna od tipa aplikacije
- Opazimo, da dodelitev enake BW v vseh primerih ne maksimira U



Temeljni mehanizmi za izvajanje QoS

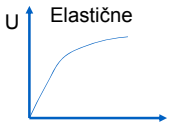


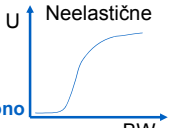
Pregled QoS mehanizmov:



Nadzor dostopa

- Z nadzorom dostopa se izogibamo preobremenitvam
  - odločitev – ali se bo v primeru dodajanja novega zahtevka poslabšala kakovost storitvam, ki imajo že dodeljene omrežne vire
- Elastične aplikacije imajo konkavno odvisnost U od BW
- Porast U se zmanjšuje s porastom BW
- Ob povečevanju tokov se njihova BW zmanjšuje
- Ni potrebno krmiljenje dostopa





- Neelastične aplikacije imajo konveksno odvisnost U od BW
- U (število tokov) se ne povečuje monotono
- Potrebno krmiljenje dostopa za maksimiranje U

Temeljni mehanizmi za izvajanje QoS

- Nadzor dostopa (Admission Control)
  - sprejema oziroma zavrača nove zahteve v skladu z razpoložljivimi zmogljivostmi v omrežju
- Oblikovanje (Shaping), vodenje (Policing), označevanje (Marking) datagramov
  - skrbi, da je bitna hitrost v skladu z dogovorjenimi vrednostmi
  - omogoča razlikovanje med datagrami različnih prioritetnih razredov
- Ločevanje (Isolation) razredov med seboj
  - preprečuje, da bi preobremenitev enega izmed razredov vplivala na zmogljivost preostalih storitvenih razredov
- Delitev (Sharing) zmogljivosti virov
  - s storitvenimi razredi je potrebno izkoristiti vire kolikor je mogoče učinkovito

Nadzor dostopa

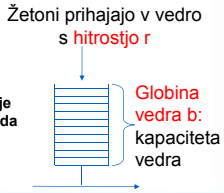
- Zavračanje
  - mehanizmi za krmiljenje dostopa zavračajo prihajajoče (nove) zahtevke
  - včasih bi bilo bolj smiselno, če bi lahko terminirali obstoječe tokove
- Alternativa: “overprovisioning” omrežja
  - težava: veliko spreminjanje poteka uporabe

Shaping, Policing, Marking

- Oblikovanje (shaping), vodenje (policing), označevanje (marking) datagramov izvajamo s postopki:
  - razvrščanja v razrede
  - razporejanja datagramov
- Pristop vedra z žetoni:
  - zakasnjevanje datagramov
  - zavračanje datagramov
  - prepusščanje datagramov

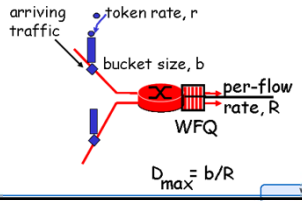
Razporejanje datagramov

- Razporejanje = izbor paketa, ki bo šel naslednji po povezavi
- Razporejanje datagramov po metodi "vedro z žetoni" (token bucket)
  - promet je določen s:
    - številom prihodov datagramov v časovni enoti r
    - globino vedra b
- Najslabša možna zakasnitev v vrsti = b/r
- Delovanje metode "vedro z žetoni":
  - če se vedro zapolni, se žetoni izgubijo
  - ko se pošlje datagram velikosti P, se porabi P žetonov
  - če je v vedru P žetonov, se datagram pošlje pri polni hitrosti, sicer je potrebno čakati, da se žetoni akumulirajo

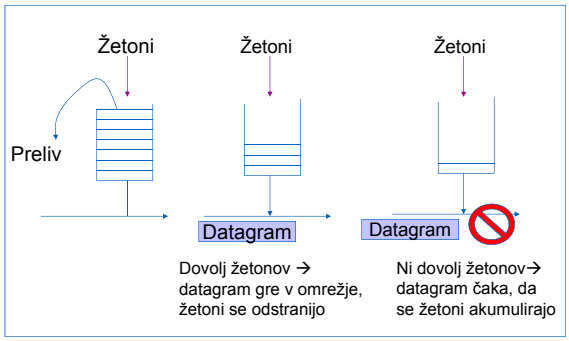


Razporejanje in razvrščanje

- Vsak prihajajoč datagram mora biti:
  - razvrščen (Classified):
    - povezan z rezervacijo neke aplikacije
    - navadno na podlagi izvornega in ponornega naslova, številke protokola, izvornih in ponornih vrat
  - razporejen (Scheduled):
    - postavljen v čakalno vrsto na način, da prejme zahtevano storitev
    - implementacija ni določena v strežnem modelu



Delovanje metode vedra z žetoni

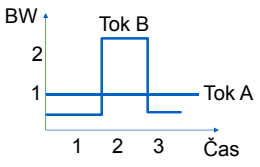


Razvrščanje v razrede

- Implicitno določanje storitvenih razredov
  - omrežje proučuje datagrame in samo razvršča toke v storitvene razrede
  - odjemalcev in aplikacij ni potrebno spreminjati
  - končen nabor aplikacij je podprt v vsakem trenutku
  - neprilagodljivost aplikacijam, ki delujejo v različnih načinih
- Eksplicitno določanje storitvenih razredov
  - aplikacija lahko neposredno zahteva storitveni razred
  - zakaj bi aplikacija zahtevala nižji razred od najboljšega:
    - cena
    - neformalne socialne zaveze
    - Congestion Control (kot pri best-effort)
  - aplikacija mora vedeti, katere storitvene razrede ima na razpolago
    - težko menjanje razredov med delovanje aplikacije
  - omrežje mora podpirati tovrstni pristop

Lastnosti metode vedra z žetoni

- V daljšem obdobju je bitna hitrost omejena na r
- V krajšem intervalu se lahko pojavijo izbruhi velikosti b
- Količina prometa v intervalu T je omejena z enačbo
  - Traffic = b + r*T
- O podatkih se aplikacija izpogaja z nadzorom dostopa



Tok A: r = 1 MBps, B=1MB

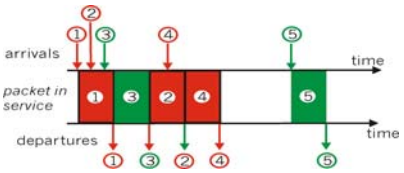
Tok B: r = 1 MBps, B=5MB

Namen uporabe metode vedra z žetoni

- Oblikovanje (shaping), vodenje (policing), označevanje (marking) datagramov
  - zakasni datagrame pred vstopom v omrežje in s tem zagotavlja, da bitna hitrost ne preraste dogovorjene vrednosti (shaping)
  - zavrže datagrame, če prispejo brez žetonov (policing)
    - omrežni element zavrže datagrame brez žetonov v primeru zamašitve v omrežju
  - prepusti datagrame v omrežje in jih označuje jih z žetoni (marking)

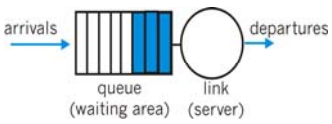
Round Robin razvrščanje

- Več razredov
- Ciklično se pregleduje vse vrste razredov, streže po en datagram iz vsakega razreda (če je ta na razpolago)



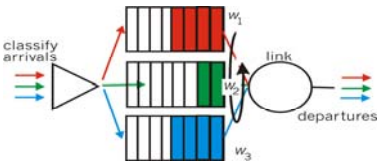
FIFO

- First In First Out: “kdor prej pride, prej melje”
- Najpreprostejši način razvrščanja
- Datagrami se pošiljajo v istem vrstnem redu, kot so prispeli v čakalno vrsto
  - politika odmetavanja datagramov: če datagramov prispe v polno vrsto, je potrebno en datagram odvreči
    - Tail drop (rezanje repa): odvrže se prihajajoč datagram
    - Priority (prioritetno): odvrže datagram na osnovi prioritet
    - Rrandom (naključen izbor): odvrže naključen datagram



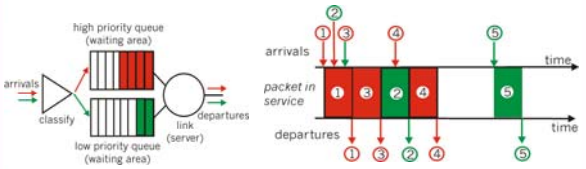
Weighted Fair Queuing (WFQ)

- Posplošen Round Robin
- Vsak razred dobi uteženo količino storitve v vsakem ciklu
  - npr. število paketov



Prioritetno razvrščanje

- Priority scheduling: najprej prenese najbolj prioritetne datagrame
  - več razredov z različnimi prioritetami
  - razred je idvisen od označb ali informacij v glavi datagrama (izvor/ponor, številka vrat, ...)



Ločevanje in delitev virov

- Ločevanje (Isolation) storitvenih razredov med seboj
  - potrebno je ločiti izvore, ki se ne obnašajo v skladu z dogovorom od tistih, ki se držijo dogovora
  - s tem se prepreči, da bi preobremenitev enega izmed razredov vplivala na zmogljivost preostalih razredov
- Delitev (Sharing)
  - mešanje prometa iz različnih aplikacij z željo po čimbolj učinkovitem izkoriščenju omrežnih virov
- Mehanizmi za izvedbo ločevanja in delitve:
  - Weighted Fair Queuing: WFQ
    - odlično ločevanje, vendar brez delitve
  - FIFO
    - odlična delitev, vendar brez ločevanja

Implementacija QoS v IP

- IETF je standardiziral dva mehanizma za izvajanje QoS
  - IntServ (RFC 1633)
  - DiffServ (RFC 2474, RFC 2475)

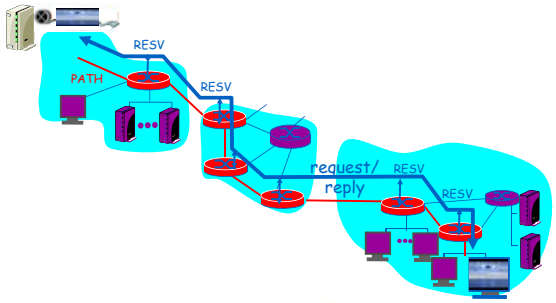
Resource Reservation Protocol: RSVP

- IntServ izvaja rezervacijo poti z uporabo Protokola za rezervacijo virov: RSVP
- RSVP je hop-by-hop signalizacijski protokol, s katero aplikacija v elementih vzdolž celotne poti rezervira potrebno pasovno širino
- Signalizacija se prenaša v obliki IP sporočil
- Izvedba RSVP rezervacije:
  - sprejemnik informacij pošlje PATH sporočilo, v katerem definira lastnosti prometa
  - vsak usmerjevalnik vzdolž poti posreduje to sporočilo naslednjemu usmerjevalniku
  - usmerjevalniki lahko:
    - sprejme zahtevo in odgovori z RESV sporočilom
    - zavrne zahtevo in odgovori s sporočilom o napaki in prekine proces rezervacije

Integrated Services in Resource Reservation Protocol

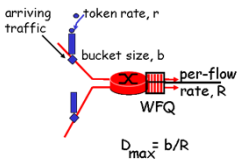
RSVP: Primer vzpostavitve zveze

- IntServ z RSVP omogoča povezavno orientirano povezavo preko nepovezavno orientiranega omrežja



Integrated Services: IntServ

- Model integriranih storitev definira tri razrede:
  - zagotovljene storitve - Guaranteed Service: GS
    - za aplikacije, ki imajo stroge zahteve glede zakasnitev in pasovne širine
  - storitve s krmiljeno obremenitvijo - Controlled Load Service: CLS
    - za aplikacije, ki zahtevajo boljšo zanesljivost in večjo pasovno širono, kot jo ponuja best-effort način
    - kvaliteta se približuje nivoju, ki bi ga podatkovni tok prejel od neobremenjenega omrežnega elementa. Z nadzorom kapacitete se omrežje obremeni do take stopnje, da aplikacija dobi zahtevane parametre tudi takrat, ko je omrežni element preobremenjen
  - najboljša zmožnost – Best-Effort



Kaj RSVP ni:

- RSVP standard ne določa pristopa, s katerim elementi omrežja dejansko izvedejo prioretizacijo datagramov oziroma rezervacijo pasovne širine
  - RSVP aplikacijam omogoča zgolj rezervacijo pasovne širine
  - pasovno širino morajo omogočiti usmerjevalniki z uporabo mehanizmov za razvrščanje
- RSVP ni usmerjevalni protokol. RSVP ne določa poti, po kateri se izvede rezervacija
  - spodaj ležeči usmerjevalni protokoli določajo usmerjevalno pot za podatkovni tok
  - ko je določena usmerjevalna pot, RSVP lahko rezervira pasovno širino vzdolž usmerjevalne poti

Slabosti IntServ pristopa

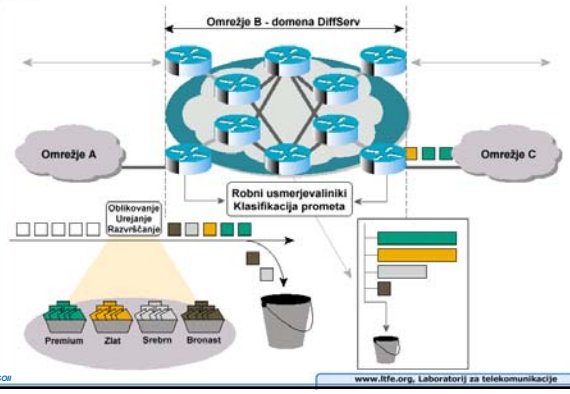
- Vsi usmerjevalniki morajo podpirati RSVP, nadzor dostopa, multifield označevanje in razvrščanje datagramov
- Količina informacij o odobrenih podatkovnih tokih na usmerjevalniku narašča premosorazmerno s številom tokov. To zahteva velike količine pomnilnika in procesne zmogljivosti na usmerjevalnikih
- Za GS je potrebna implementacija IntServ vzdolž celotnega omrežja od izvora do ponora, medtem ko se pri storitvah CLS mehanizem IntServ lahko uvaja predvsem v omrežjih z ozkimi grli in tunelira datagrame RSVP
- IntServ ni primeren za hrbtenična omrežja

Arhitektura DiffServ

- Dogovori o storitvah znotraj domene
  - Service Level Agreements (SLA) s ponudnikom internetnega dostopa ISP
- Promet sprejmejo in oblikujejo usmerjevalniki na robu omrežja (Edge routers)
  - izvajajo oblikovanje (shaping) in vodenje (policing)
  - označujejo datagrame z majhnim številom bitov; vsak bit predstavlja storitveni razred ali podrazred
- Jedrni usmerjevalniki (Core routers)
  - obdelujejo datagrame glede na njihove označbe in v skladu s pravili za per-hop obnašanje
- Arhitektura DiffServ je bolj razširljiva kot arhitektura IntServ
  - ni signalizacije, povezane s tokovi

Differentiated Services: Diffserv

Arhitektura DiffServ: Slika

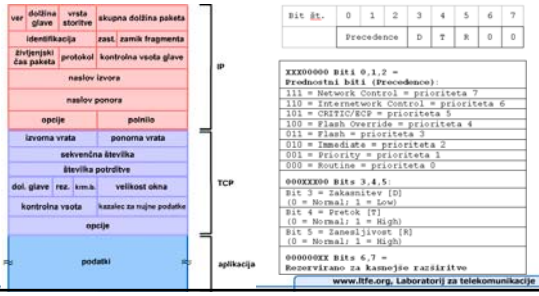


Differentiated Services: Diffserv

- Zaradi prej navedenih težav pri implementaciji mehanizma IntServ in protokola RSVP, je bil razvit protokol Differentiated Services – DiffServ
- DiffServ
  - ima popolnoma odpravljeno shranjevanje in obdelavo sej
  - ni zmogljivostno in pomnilniško potraten; zato pogosto uporabljen
- DiffServ omogoča več razredov storitev:
  - Premium service: namenjena aplikacijam, ki zahtevajo nizko zakasnitev in nizko potresavanje zakasnitve (jitter)
  - Assured service: namenjena aplikacijam, ki zahtevajo zanesljivost, boljšo od best-effort
  - Olympic service: ponujajo tri različne tipe storitev: zlata, srebrna in bronasta

DiffServ - označevanje

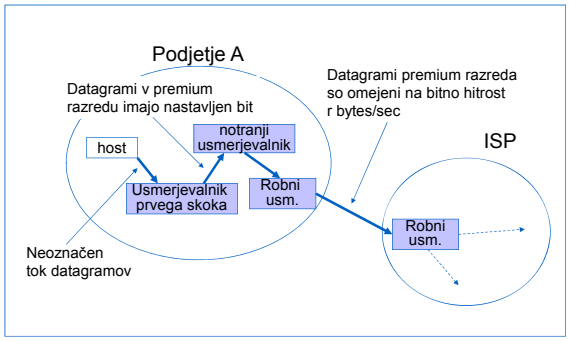
- Protokol IP (že v v4) določa v glavi datagrama polje ToS
  - ToS – Type of Service
- To polje je v DiffServ redefinirano
  - DSCP – DiffServ Code Point: vrsta storitve (RFC 791)



DiffServ - označevanje

- Označevanje prometa pomeni razvrščanje v storitvene razrede
- Razvrščanje ni natančno določeno – prepuščeno je administratorju omrežja
  - da uporabnik pridobi storitev določene prioritete, more s svojim ISP podpisati SLA.
    - statična
    - dinamična (uporaba RSVP)
  - deleži izgub niso natančno ampak samo relativno definirani
- Datagrame lahko označujejo:
  - uporabniki sami
  - vstopna vozlišča
- Po vstopu v omrežje so datagrami označeni in razvrščeni
- Prometni tok je po potrebi preoblikovan, v skladu s pravili, določenimi v SLA
- Kadar datagram preide iz ene domene v drugo, se označba njegovega DSCP polja lahko spremeni v skladu z SLA dogovori

EF tokovi



Per-hop Behaviors (PHBs)

- Določa obnašanje posameznega usmerjevalnika in ne celotne poti od izvora do ponora
- Različni načini obnašanja usmerjevalnika za različne skupine storitev (Behaviors) – potrebujemo več bitov v glavi datagramov
- Zahtevkov na usmerjevalniku je lahko bistveno več, kot različnih načinov obnašanj usmerjevalnika
- DiffServ uporablja shemo relativnih prioritet

- Definirana sta dva načina PHB:
  - Expedited forwarding oziroma Premium service (tip P)
  - Assured forwarding (tip A)

Assured Forwarding PHB

- Možna uporaba: zagotavljanje virov prometu znotraj profila, dovoljena prekoračitev s profilom določenega dogovora
- Nadzor temelji na pričakovani kapaciteti uporabniškega profila
- Uporabnik in omrežje se dogovorita glede profila podatkov
  - datagrami, ki se držijo profila, so označeni
    - malo verjetno je, da bi se datagrami zavrgli, če se uporabnik drži profila
  - datagrami, ki se ne držijo profila, se označijo z dvakrat večjo verjetnostjo izgube

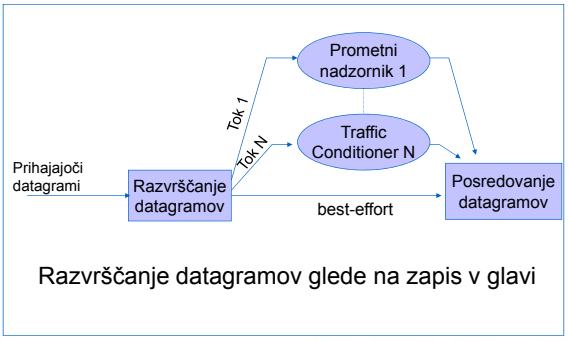
- Če pride do zamašitve na vozlišču, ki podpira DiffServ, potem vozlišče ščiti datagrame z nižjo verjetnostjo izgube, datagrami, ki se ne držijo profila so prej zavrženi
  - Mehanizem implementiran z uporabo mehanizma RIO: RED with In/Out bit
  - WFQ
  - GRR,
  - ...

Expedited Forwarding PHB

- Možna uporaba: navidezni vod
- Nadzor dostopa se izvaja na podlagi parametra maksimalnega pretoka
- Neuporabljena pasovna širina se porabi za best-effort promet
- Uporabniki pošiljajo podatke v okviru dogovorjenega profila
  - signalizacija in nadzor dostopa se bosta v prihodnosti verjetno še nekoliko razvila (Bandwidth Brokerji)
- Hitrost EF datagramov se omejuje zgolj na robovih omrežja
  - v ta namen se uporablja že predstavljen mehanizem vedra z žetoni
- Enostavno posredovanje datagramov:
  - datagrame razvrstimo v eno izmed dveh vrst in izvajamo prioretizacijo
  - EF datagrami se posredujejo s:
    - čimnižjo zakasnitvijo
    - čimmanj izgubami

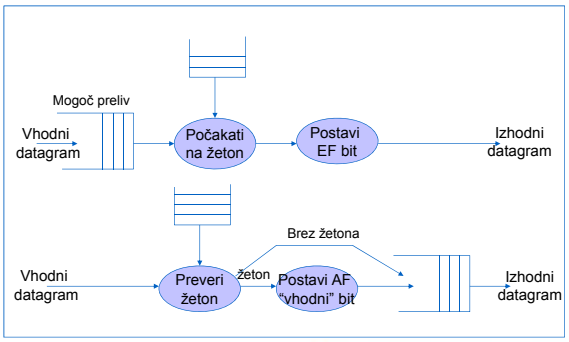
    (do kolikor to dopušča kapaciteta usmerjevalnika)

Vhodna funkcionalnost robnih usm.



Razvrščanje datagramov glede na zapis v glavi

Prometni nadzor

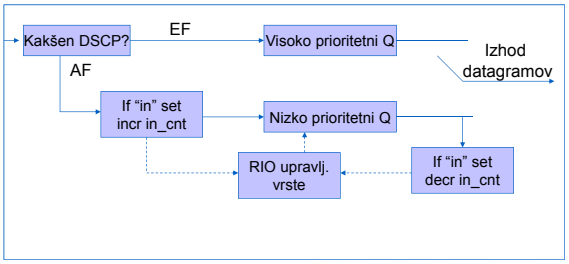


Primerjava med DiffServ in IntServ

- DSCP polje pri DiffServu omogoča le omejen nabor storitvenih razredov
- Količina informacij je premosorazmerna številu storitvenih razredov
- DiffServ je bolj razširljiv kot IntServ
- Kompliciran postopek označevanja, razvrščanja in preoblikovanja se izvaja le na robnih usmerjevalnikih
- Jedrni usmerjevalniki se ukvarjajo zgolj s po razredih agregiranim prometom

Izhodno posredovanje

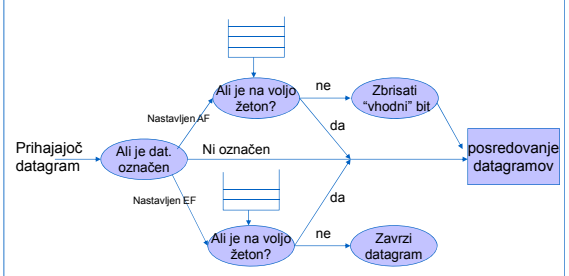
- 2 vrsti: EF datagrami v vrsti z večjo prioriteto
  - Nižja prioriteta implementirana po RIO (RED "In or Out") shemi



Primerjava

	Best-Efforts	Diffserv	Intserv
Storitev	• Povezljivost • Ni izolacije • Nobenih zagotovil	• Izolacija agregiranih tokov • Zagotavljanje celotni agregaciji	• Tokovna izolacija • Zagotavljanje tokom
Doseg storitve	• End-to-end	• Domenski	• End-to-end
Kompleksnost	• Ni vzpostavitve	• Dolgotrajna vzp.	• Vzpostavitev za tok
Razširljivost	• Velika razširljivost • (vozlišča skrbijo samo za stanje usmerjevaljnikov)	• Razširljivost (samo robni usmerjevalniki skrbijo za agregiran pretok, jedrni usmerjevalniki pa skrbijo za celoten razred)	• Ni razširljiv (vsak usmerjevalnik mora skrbeti za stanje nabora tokov)

Policing robnih usmerjevalnikov



Hvala za pozornost.
Vprašanja?

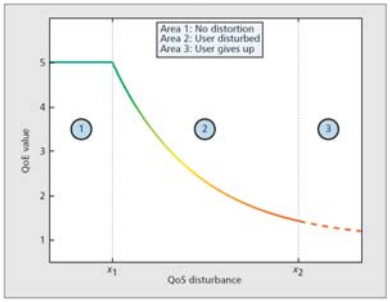
Prejeta kakovost – uporabniška izkušnja QoE

doc.dr. Iztok HUMAR
prof.dr. Janez BEŠTER

Vsebina

- Uporabniška izkušnja
- Vrednotenje uporabniške izkušnje za različne aplikacije
  - VoIP
  - Video
  - Splet
  - HASS

Uporabniška izkušnja – generalno



- Odvisnost uporabniške izkušnje (zadovoljstva uporabnika) od motnje v telekomunikacijskem sistemu

Motnja – neodvisna spremenljivka

- ATM
  - izgubljanje celic
- IP
  - izgubljanje in podvajanje paketov: loss [%]
  - zakasnitve pri prenosu paketov: delay [s]
  - potresavanje zakasnitev: jitter [s]
  - premajhna pasovna širina: bandwidth [bit/s]
- Mobilna
  - podvajanje paketov
  - namerno dodajanje šuma za identificiranje prisotnosti zveza
  - premajhna pasovna širina

Vpliv motenj na uporabniško izkušnjo

- Močno odvisen od vrste aplikacije:
  - Vpliv motenj na kakovost npr. govorne komunikacije
    - napačno predvajanje posameznih segmentov govora
    - izgubljanje posameznih segmentov (tišina)
    - prekinitev zveze
- Odvisen od:
  - mehanizmov za odpravljanje napak & redundance
  - uporabljenih kodekov
- Iz podatkov o motnjah težko sklepamo o njihovem vplivu na uporabniško izkušnjo
  - potreba po mehanizmih za subjektivno vrednotenje kakovosti pri znanih stopnjah motenj
  - vrednotenje na podlagi modelov (izdelava modelov)

Subjektivno ocenjevanje UI

- Koncept subjektivnega ocenjevanja uporabniške izkušnje:
  - primarno razvit za področje prenosa govora
  - se aplicira tudi na ostalih področjih
  - temelji na merjenju uporabniške izkušnje
- Osnovna ideja:
  - na podlagi spremljanja osnovnih lastnosti realnega signala, prenašanega preko testiranega sistema/omrežja, zberemo ocene kakovosti izhodnega signala, ki ga prejeme uporabnik
- ITU-T v letu 1996 izdelal priporočilo:
  - Recommendation P.800 (1996): "Methods for subjective determination of transmission quality".
- Prednost:
  - pridobimo relevantno oceno zaznane kakovosti za vzorčno populacijo
- Slabost:
  - subjektivno ocenjevanje: zelo časovno potratno
  - ni mogoče dobiti enoumne ocene za kakovost: večji vzorec

Subjektivno ocenjevanje: MOS

- Ocenjevanje kakovosti poteka z uporabo točkovanja :
  - Lestvica vrednsoti: od 1 do 5
  - neposredno ocenjevanje po metodi določanja absolutnih ocen - ACR (Absolute Category Rating)
  - relativno glede na referenčni signal po metodi določanja degradacije - DCR (Degradation Category Rating)

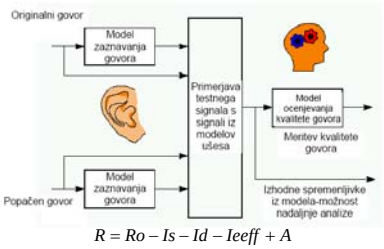
Točke	Kakovost govora	Quality of Speech
5	Odlična	Excellent
4	Dobra	Good
3	Zadovoljiva	Fair
2	Skromna	Poor
1	Slaba	Bad

Točke	Sprememba signala	Degradation is
5	Neslišna	Inaudible
4	Slišna, nemoteča	Audible, not annoying
3	Rahlo moteča	Slightly annoying
2	Moteča	Annoying
1	Zelo moteča	Very annoying

- Rezultat je podan kot ocena MOS (Mean Opinion Score)

Objektivno ocenjevanje UI

- Identifikacija tistih parametrov, ki pomembno vplivajo na uporabniško izkušnjo za posamezno aplikacijo
- Vrednotenje njihovega vpliva skozi matematične modele



$$R = Ro - Is - Id - Ieeff + A$$

$$MOS = 1 + R \cdot 0{,}035 + R \cdot (R - 60) \cdot (100 - R) \cdot 0{,}0000007$$

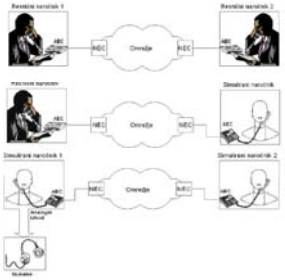
Vrednotenje UI pri govorni komunikaciji

- VoIP predstavljala eno izmed donosnih aplikacij v omrežju IP, zato so se tu začele raziskave in standardi na področju vrednotenja UI.
- Subjektivno ocenjevanje na podlagi mnenja (dolgotrajno)
  - *Opinion Equivalent-Q*: P-819, uporaba Modulated noise reference unit
- Objektivno ocenjevanje – modeli, izvedeni na podlagi MOS
  - *Mnenjski modeli*: poznavanje parametrov omrežja, terminalske opreme
  - *Objektivni modeli govornega nivoja*: PSQM, PEAQ, PESQ
  - *Objektivni modeli paketnega nivoja*: P.VTQ

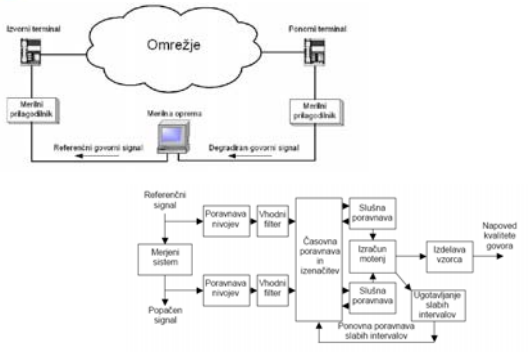
ITU-T kodek	shema	bitna hitrost	vzorčenje	kodirna zak.	MOS
G.711	PCM	64 kbps	8	<1 msec	4,4
G.726	ADPCM	32 kbps	4	1 msec	4,2
G.729	CS-CELP	8 kbps	80	15 msec	4,2
G.723.1	MPMLQ	6,3 kbps	192	37,5 msec	4,0
G.723.1	ACELP	5,3 kbps	160	37,5 msec	3,5

Metode ocenjevanja UI pri VoIP

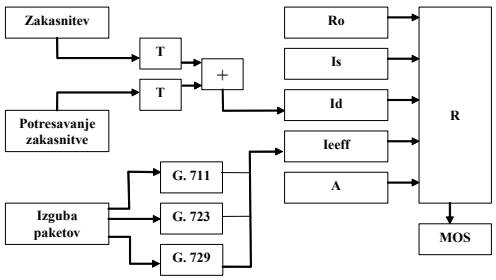
- Slušno kakovost (ena smer):
  - algoritma P.862 (PESQ) in P.862.1
- Pogovorno kakovost (interaktivno):



Objektivno ocenjevanje VoIP: PESQ

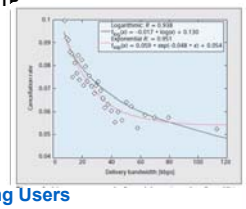


Objektivno ocenjevanje VoIP: E-model



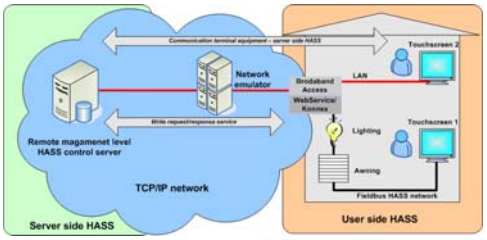
Uporabniška izkušnja s spletom

- ITU-T Recommendation G.1030, "Estimating End-to-End Performance in IP Networks for Data Applications"
- Uporabniki ocenjujejo odziv in hitrost nalaganja strani:
  - izmerjeno v omrežju
  - izračunano iz transakcijskih časov HTTP
- Podajajo na skali MOS
- Za tri čase trajanja sej:
  - 6 s, 15 s, 60 s.
- Tri aktivnosti:
  - Zahtevek za spletno stran
  - Vnos vsebine v polje (Form)
  - Sprejem rezultata
- Cancellation Rate of Web Browsing Users
  - Pri različnih pasovnih širinah



Vpliv zakasnitve na UI pri HASS

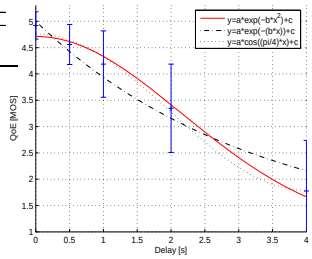
- Testirane aplikacije:
  - Diskretno (prižiganje/ugašanje) svetil
  - Zvezno krmiljenje svetil (dimming)
  - Nastavljanje žaluzije na določeno višino
  - Sprememba kota žaluzije



Rezultati UI v HASS

			HASS service		
par.	*discr. light. ctrl*	*cont. light. ctrl*	*awn. ctrl–up/down*	*awn. ctrl–stop at point*	*awn. ctrl–adjust angle*
R^2	0.98	0.98	0.97	0.97	0.98
RMSE	0.18	0.17	0.23	0.31	0.23
MCI	0.32	0.29	0.31	0.31	0.31

Zone type	*Estimated MOS value*
Zone of satistfaction	4 ≤ MOS ≤ 5
Zone of tolerance	3 ≤ MOS < 4
Zone of frustration	1 ≤ MOS < 3

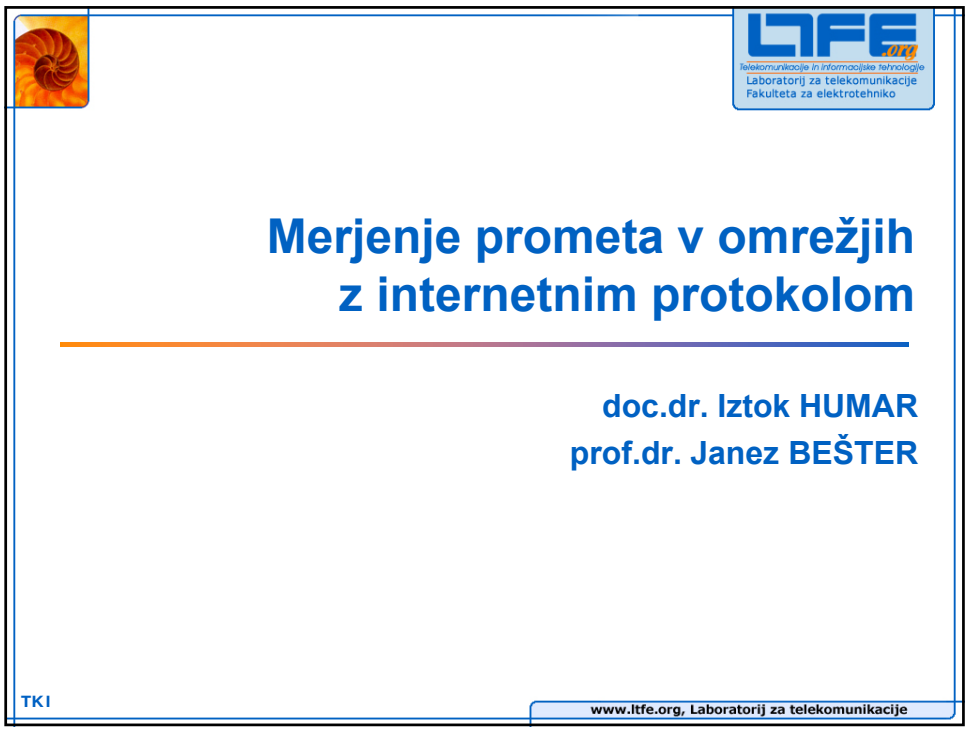


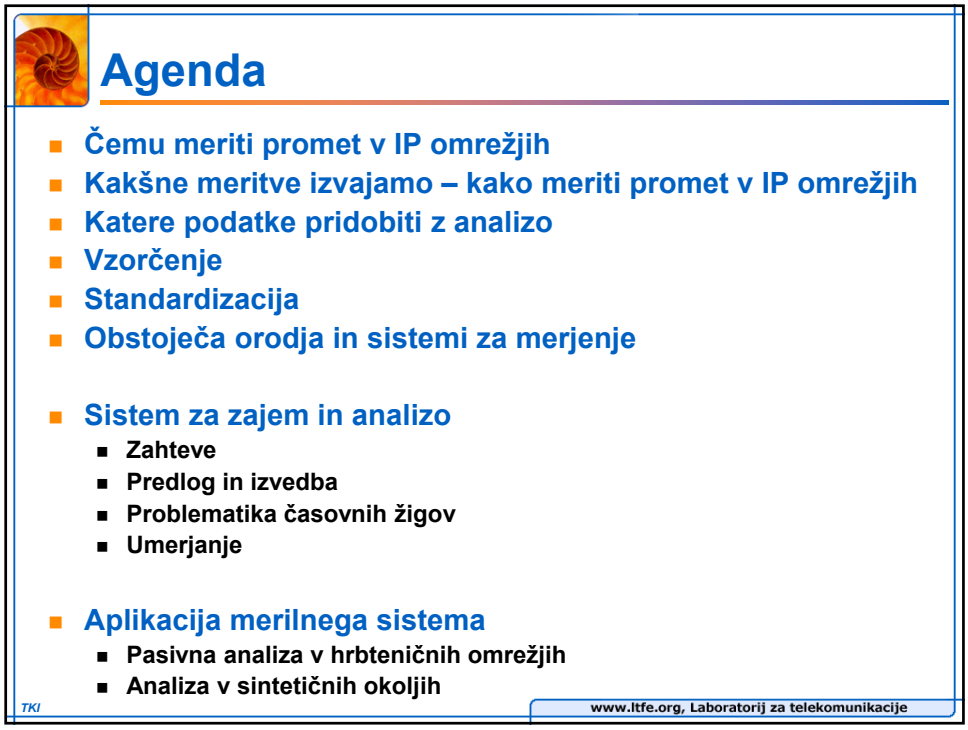
Laboratorijska vaja

- Ocenjevanje prejete kakovosti za različne aplikacije/protokole v omrežjih z oteženimi razmerami
  - Preizkus http preko povezave z oteženimi razmerami
  - Simetrična povezava – emulacija povezave preko geostacionarnega satelita
  - Asimetrične povezave ADSL brez izgub
  - Asimetrične povezave Wimax z izgubami
- Uporaba emulatorja omrežja - NetEM

Emulacija omrežja

HVALA ZA POZORNOST,
VPRAŠANJA !?


Laboratorij za telekomunikacije
Fakulteta za elektrotehniko
Merjenje prometa v omrežjih
z internetnim protokolom
doc.dr. Iztok HUMAR
prof.dr. Janez BEŠTER
TKI
www.ltfe.org, Laboratorij za telekomunikacije


Agenda
Čemu meriti promet v IP omrežjih
Kakšne meritve izvajamo – kako meriti promet v IP omrežjih
Katere podatke pridobiti z analizo
Vzorčenje
Standardizacija
Obstoječa orodja in sistemi za merjenje
Sistem za zajem in analizo
Zahteve
Predlog in izvedba
Problematika časovnih žigov
Umerjanje
Aplikacija merilnega sistema
Pasivna analiza v hrbteničnih omrežjih
Analiza v sintetičnih okoljih
TKI
www.ltfe.org, Laboratorij za telekomunikacije

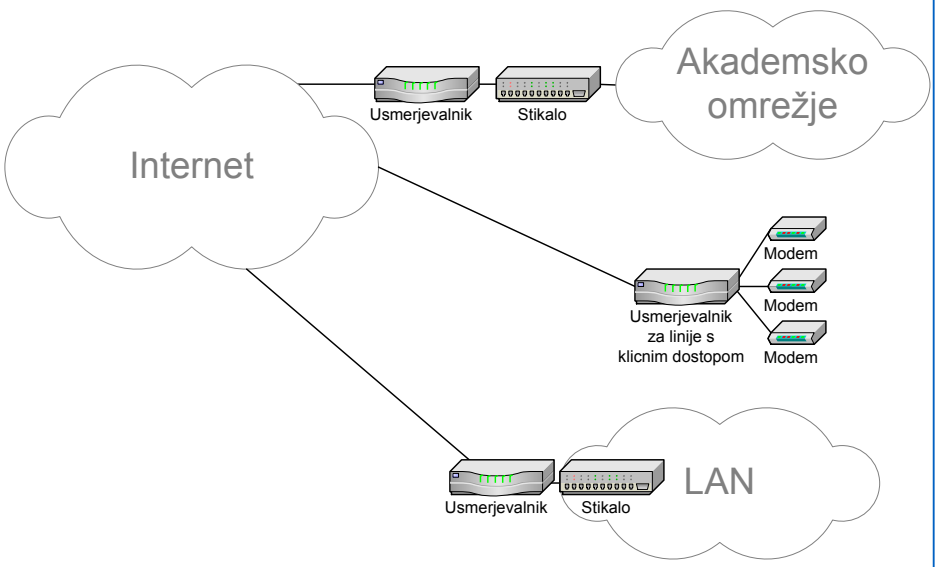
Čemu meriti promet?

- **Nadzor nad omrežnim prometom**
  - Spremljanje delovanja omrežja
  - Upravljanje z omrežjem, vzdrževanje omrežja
  - Ugotavljanje izkoriščenosti omrežja
  - Preprečevanje zamašitev, odpravljanje napak na omrežju
- **Analiza prometa omogoča odkrivanje napadov**
  - DoS
- **Pridobivanje temeljnih informacij za razvoj omrežja**
  - Načrtovanje izrabe omrežnih virov
  - Načrtovanje kakovosti storitev
- **Modeliranje**
  - Prometa
  - Omrežnih elementov
- **Potreba po merjenju prometa v LTFE:**
  - Projekt: Aktivne meritve videa pri uporabi mehanizmov za QoS
  - Raziskovalno delo: Zajeti sled realnega prometa za analizo prometa in analizo zmogljivosti delovanja stikal

Kako meriti promet?

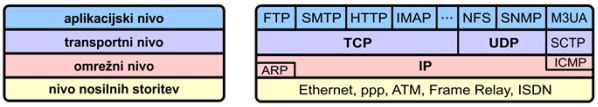
- **Merjenje prometa vključuje:**
  - Zajem prometa
  - Analizo prometa
- **Glede na vir prometa**
  - Pasivni sistemi
    - Zgolj zajemajo in analizirajo promet na povezavah, v omrežju
    - Navadno v realnih omrežjih
  - Aktivni sistemi
    - V omrežje pošiljajo “sintetičen promet” in analizirajo obnašanje
    - Omrežje je lahko sintetično ali realno
- **Glede na analizo prometa**
  - Sprotno (*Online*) analiza
    - Težave z dinamično obdelavo podatkov (počasnejše povezave)
  - Naknadna (*Offline*) analiza zajetega prometa
    - Podrobnejše analize
    - Naknadne analize

Topologija omrežja z internetnim protokolom



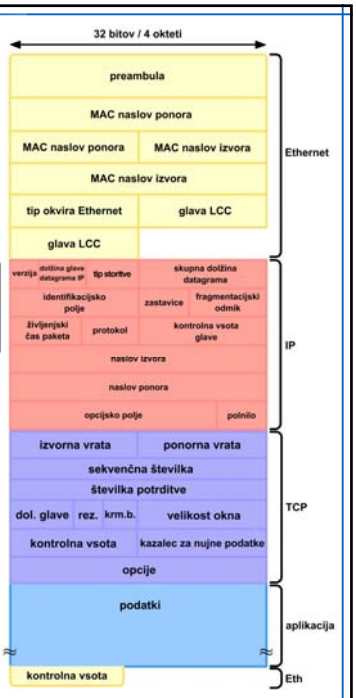
Zajemani podatki

- **Podatki iz glave:**
  - **mrežnega nivoja IP**
  - **transportnega nivoja**



- **Časovni žig:**

```
7.12.2012, 8:12:23.253845
```



Merjeni in analizirani podatki

- **Analiza časovnih žigov, štetje datagramov**
  - Določanje števila prenesenih datagramov v časovnem intervalu
  - Usmerjevalniki: število paketov na sekundo (10000 - 10^7)
- **Analiza časa med prihodi paketov (interarrival time)**
  - Obremenitev stikal, usmerjevalnikov
- **Merjenje obhodnega časa in enosm. zakasnitev v omrežju**
- **Merjenje avtokorelacijske funkcije**
  - Ugotavljanje *dolgoročne odvisnosti* (LRD): $r(k) \approx Ck^{-\beta}$, ko gre $k \to \infty, 0 < \beta < 1$
  - Ugotavljanje *kratkoročne odvisnosti* (SRD): $r(k) \sim \rho^k, 0 < \rho < 1 \quad \sum_k r(k) < \infty$
- **Analiza samopodobnosti (*Hurst* parameter)**
  - X ima samopodobne lastnosti s parametrom H, če velja, da ima za vsak pozitiven m enako porazdelitev kot X, le da je ta m-krat skalirana

$$X_n = m^{-H} \sum_{i=(n-1)m+1}^{nm} X_i = m^{-H} X^{(m)}; \quad 0{,}5 \le H \le 1$$

Merjeni in analizirani podatki

- **Analiza velikosti datagramov**
  - Določitev povprečne velikosti paketov
  - Določitev porazdelitve števila paketov glede na velikost paketa
  - Določitev bitne hitrosti na merjeni povezavi v posamezno smer
- **Analiza naslovov omrežnega nivoja**
  - Podatki o kdo komunicira s kom (vloge sistemov)
  - Koliko prometa izvira/konča na istem sistemu
  - Simetričnost prometa
- **Analiza življenskega časa paketa**
  - Preko koliko usmerjevalnikov je prepotoval datagram
  - Različne začetne vrednosti

OS	TTL
Windows 95	32
Digital OSF	60
Linux, Sun OS	64
Windows 98, NT, 2000	128

Merjeni in analizirani podatki

- **Analiza protokola**
  - Grobo sklepanje na tip aplikacij
    - TCP – potrjevanje
    - UDP – slepo nadgrajuje IP
  - Obnašanje prometa je povezano s protokolom
    - TCP – elastičen promet
    - UDP – pretočni promet
- **Analiza naslovov transportnega nivoja**
  - Podatki o aplikacijah
  - Potrebna analiza glav transportnega nivoja
    - večja količina zajetih podatkov
    - zamudnejša analiza
- **Analiza tokov**

  Tok (*flow*) = Zaporedje koreliranih paketov (denimov v TCP povezavi)
  - Analiza izvorov/ciljev
  - Analiza določine tokov (bitih), porazdelitev
  - Analiza trajanja tokov, porazdelitev

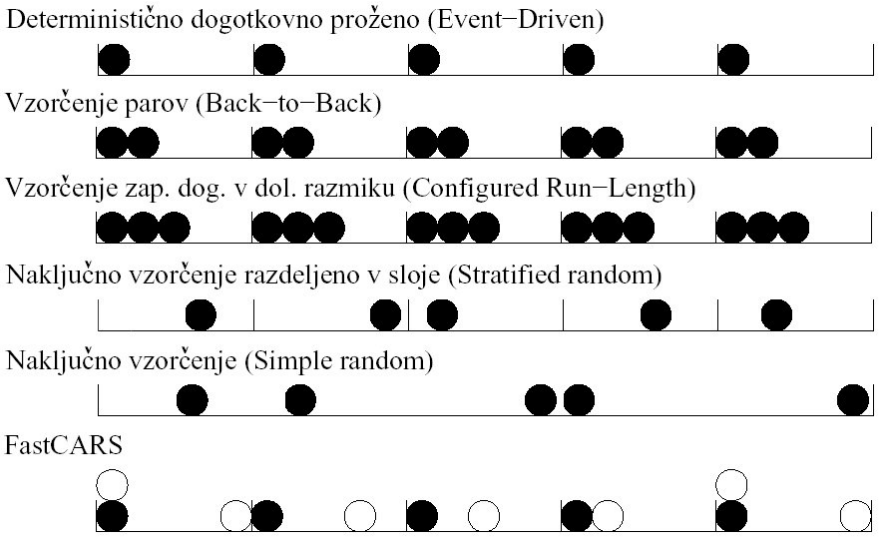
Merjenje z vzorčenjem

- **Omogoča izvajanje analize parametrov z uporabo bistveno krajših zajetih sledi.**
- **Zahteve:**
  - analiza vzorcev naj omogoča čimbolj točno merjenje statističnih parametrov celotne sledi
  - tehniko vzorčenja mora biti mogoče implementirati na čim bolj enostaven način
  - vzorci naj omogočajo določitev korelacije med zaporednimi datagrami in korelacijo med datagrami, ki so dlje narazen
- **Dogodki, ki prožijo zajem vzorca:**
  - časovnik (*timer-driven sampling*)
  - števec (*packet-driven sampling*)

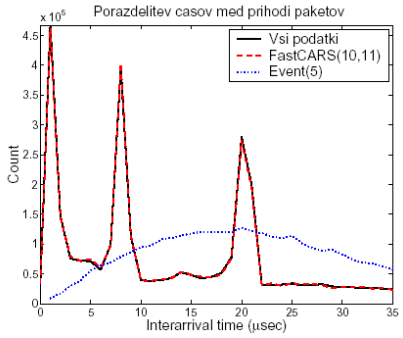
Tehnike vzorčenja

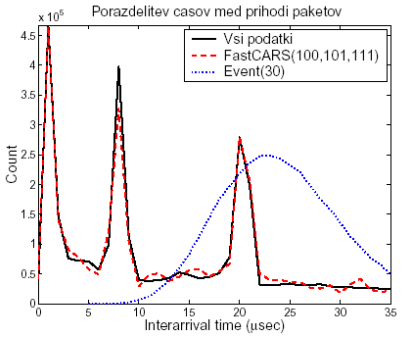
- **Deterministično dogodkovno proženo vzorčenje**
  - **event-driven sampling:** Event(p)
  - **Vzorce zajemamo v intervalih** $0, p, 2p$
- **Vzorčenje več zaporednih dogodkov v določenem razmiku**
  - **configured run-level sampling:** Conf(p, q)
  - **Vzorce zajemamo v intervalih** $kp, (kp + 1), ..., (kp + q - 1)$
- **Vzorčenje zaporednih parov**
  - **back-to-back sampling:**
    - back-to-back(p) = Conf(p, 2), **vzorce zajemamo v intervalih:** $0, 1, p, (p + 1), 2p$
    - back-to-back(p, s), **vzorce zajemamo v intervalih:** $0, s, p, (p + s), 2p, (2p + s)$
- **Naključno vzorčenje**
  - **razdeljeno v sloje (*stratified random sampling*)**
  - **enostavno naključno vzorčenje (*simple random sampling*)**
- **Hitra metoda vzorčenja, ki upošteva korelacijo: FastCARS**
  - **Fast, correlation-aware sampling method:** FastCARS(p_1, p_2, ..., p_n)
  - **Vzorce zajemamo v intervalih** $p_1, p_2, ... , p_n$; p_i **tuja števila ali praštevila**

Tehnike vzorčenja



Primerjava različnih tehnik vzorčenja





- Z vzorčenjem po metodi FastCARS izmerimo bistveno bolj točno oceno porazdelitve časov med prihodi datagramov, pri čemu ne gre spregledati dejstva, da je število vzorcev pri uporabi metode FastCARS(10,11) (713435) celo nekoliko manjše od števila vzorcev pri uporabi metode Event(5) (747407).

Standardi

- Internet Engineering Task Force
  - Internet Protocol Performance Metrics – IPPM
    - Internet Draft dokumenti (6)
    - RFC dokumenti (9)
- Cooperative Association for Internet Data Analysis
  - + Univerza v Michiganu + ponudnik internet dostopa Merit Network
  - Internet Performance Measurement and Analysis – IPMA
    - Merjenje internetne statistike za potrebe zagotavljanja stabilnosti pri usmerjanju, nadzora topologije in merjene ter vizualizacijo merjenih zmogljivosti.
    - Izdelali so priporočilo za izvajanje meritev in analizo statističnih podatkov, ter izdelali eksperimentalno študijo o stabilnosti interneta v primeru izpada hitrih hrbteničnih povezav za Ameriko.
- IP Network Management and Performance Deptartment, AT&T
  - AT&T Network Measurement Tools
    - Zajem: *PacketScope* + *Cisco NetFlow* oziroma *Traffic Aggregation Probe*
    - Analiza
    - Podatki za nadzor in upravljanje AT&T hrbtenice

Orodja

- **Pasivna orodja**
  - SNMP Agenti
  - tcpdump
  - NetFlow
- **Aktivna orodja**
  - ping
  - traceroute
- **Sistemi**
  - Dag merilna kartica
  - IPMON
  - Tstat

LTFE TestCenter

- **Test and verification lab**

  Critical mass of knowledge, experience and top-level equipment, R&D support
  Academic research – B.Sc. And Ph.D. students
  Industry partners and organizations

- **Test and verification services, R&D**

  Celovito testiranje in verifikacija

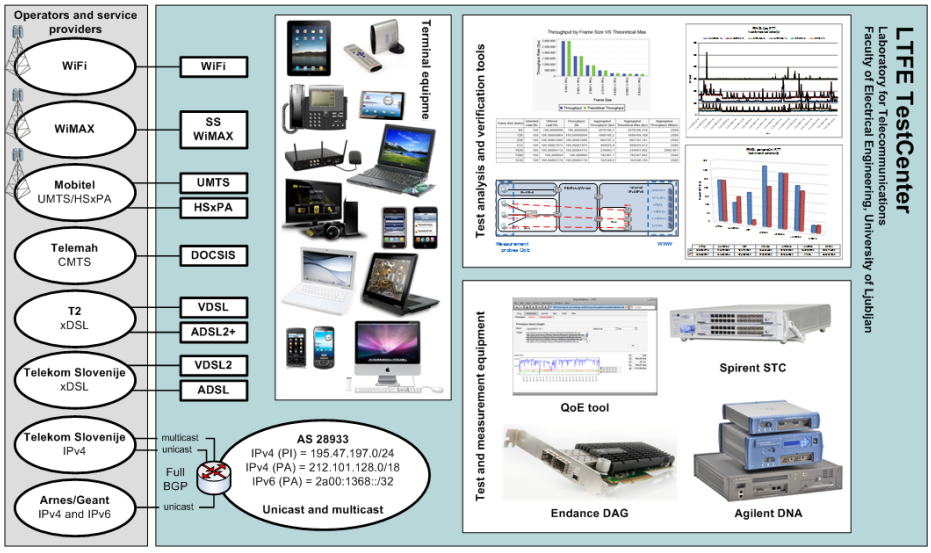
  1. Vzpostavitev in konfiguracija testnih sistemov in pilotnih implementacij
  2. Izvedba meritev, testiranj in verifikacije za telekomunikacijske sisteme, storitve in protokole
  3. Analiza rezultatov, diagnostika in verifikacija

TIPI TESTIRANJ: performančni, funkcionalnih interoperabilnostnih primerjalni in skladnostni testi
IZVEDBA: z vrhunsko opremo, ki je na voljo v TestCentru
Izvedba celotnega R&D cikla
Raziskave in analize
Načrtovanje, razvoj in implementacija sistemov, storitev in protokolov
Meritve, testiranje in verifikacija
Vzdrževanje

LTFE TestCenter – solutions&services

- **R&D**
  Entire R&D cycle – analysis, research, design, development, programming, testing and verification, maintenance&support
- **Pilot implementations**
  Networking solutions and services
  IP, IPv6, MPLS, MetroEthernet, VPN…
  Mobile&wireless architectures
  UMTS, WiFi, WiMAX, UMA…
  NGN/IMS and WEB 2.0 development (solutions, services)
  System tools
  QoS/QoE measurement tool, VPN security solution, Broadband network planning tool
- **Product development&implementations**
  Protocol development and integration into vendor equipment
  SS7, SIGTRAN, AAA (Diamater, Radius, …)
  Monitoring, SORM, DPI
  Media gateway signalling part, Call servers
  System integration and engineering
- **Professional equipment**
  Spirent STC, Agilent DNA, Endace DAG,Efficient IPv4/IPv6 networking infrastructure
  Broad range of BB SP connections

LTFE TestCenter infrastructure summ.

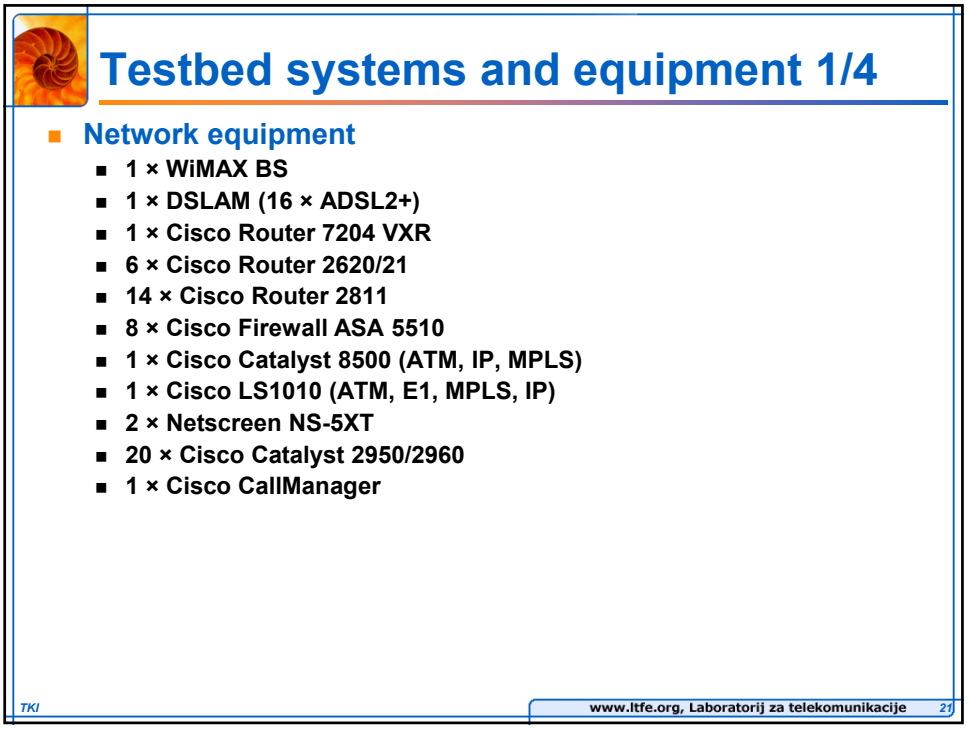


TestCenter infrastructure – details 1/2

- Dual stack IPv4/IPv6 network
  - Multicast enabeled
  - Structured wireing: cat6, fiber SM/MM
  - IPSec VPN, SSL VPN
  - Equipment
    - Cisco 7604, 11 × Cisco Catalyst 3560, Cisco ASA 5510, Juniper ISG 1000
- Connectivity
  - Full BGP connectivity, AS = 28933
  - 4 × native ISP connections
    - ARNES/GIANT, Telecom Slovenia
  - 10+ BB access connections (various slovenian providers)
    - Technolgies ADSL, ADSL2+, VDSL, VDSL2, FTTH, WiMAX 802.16D, WiMAX 802.16e, HSxPA, DOCSIS)
  - ALL Slovenian VoIP providers
    - Telekom, T2, AMIS, UPC Telemach, Detel, Planet9, Tuš …

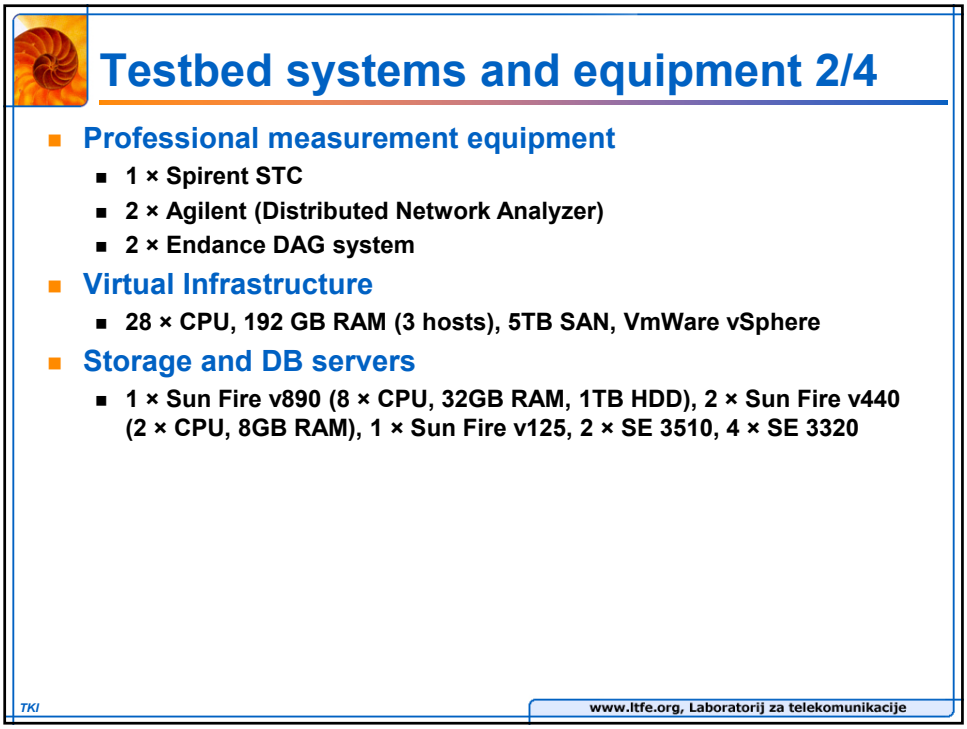
TestCenter infrastructure – details 2/2

- WiMAX test-bed (Telsima)
  - QoS implemetation for WiMAX BS/SS
  - beta testers for WiMAX 802.16D and 802.16e
- NATO – “TACOMS Phase 1” test-bed
  - Integratin: DNS, LDAP, CO BGP, H323 call control
  - Cisco CM, ISODE (M-Vault), OPEN LDAP,
- NGN/IMS test-bed
  - IMS – Fraunhofer FOKUS
  - 2 × Asterix VoIP system
  - 2 × SER VoIP system
  - Development Sigtran, Diameter, ParlayX GW, presence services
- IPTV test-bed
  - First commercial IPTV solution in Europe (coperation with Telecom Slo.)



Testbed systems and equipment 1/4

- Network equipment
  - 1 × WiMAX BS
  - 1 × DSLAM (16 × ADSL2+)
  - 1 × Cisco Router 7204 VXR
  - 6 × Cisco Router 2620/21
  - 14 × Cisco Router 2811
  - 8 × Cisco Firewall ASA 5510
  - 1 × Cisco Catalyst 8500 (ATM, IP, MPLS)
  - 1 × Cisco LS1010 (ATM, E1, MPLS, IP)
  - 2 × Netscreen NS-5XT
  - 20 × Cisco Catalyst 2950/2960
  - 1 × Cisco CallManager



Testbed systems and equipment 2/4

- Professional measurement equipment
  - 1 × Spirent STC
  - 2 × Agilent (Distributed Network Analyzer)
  - 2 × Endance DAG system
- Virtual Infrastructure
  - 28 × CPU, 192 GB RAM (3 hosts), 5TB SAN, VmWare vSphere
- Storage and DB servers
  - 1 × Sun Fire v890 (8 × CPU, 32GB RAM, 1TB HDD), 2 × Sun Fire v440 (2 × CPU, 8GB RAM), 1 × Sun Fire v125, 2 × SE 3510, 4 × SE 3320

Testbed systems and equipment 3/4

- **Audio – video infrastructure**
  - 4 × ProHD JVC, 2 × full HD JVC, 2 × ProHD Sony cameras
  - 5 × Sony BRC300 remote controled cameras
  - 4 × video mixer (with audio, graphics, title insertion, chroma keying,..)
  - 2 × audio mixers
  - 2 × audio-video matrix 16×16
  - Lightning and other audio equipment
- **Apple equipment (video production)**
  - 12 × iMac 27” i7 2,93 GHz, 8GB RAM,
  - 5 × MAC Pro (2×6 core 2,66 GHz, 16 GB RAM, 4TB disc, Cinema Display 27”, AJA kona 3)
  - 4 × MacBook PRO (15,4”, C2D 2,8 GHz, 4GB RAM)
  - 2 × Mac Mini
  - 1 × Xserve (2×4 core 2,66 GHz, 12GB RAM, 3TB HDD, Fiber channel)
  - ES_6216SX storage 32TB (Fiber Channel)

Testbed systems and equipment 4/4

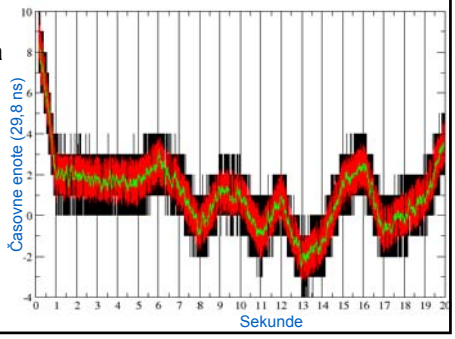
- **Broadcasting infrastructure (producting and measuring)**
  - DVB-T/H RF probe and Digital spectrum analyzer
  - DVB-T/H VHF and UHF modulators and demodulators
  - DVB Multiplexer
  - DVB-T reciver (redistributor, dual channel, diversity mode)
  - DVB Analayzer
  - Multipurpose ASI / IP Stream station with analyzing tools
  - ASI/SDI streamer
  - Harris VTM4100 video analyzer (Waveform, Vector, Gamut, Picture, and Timing screens with digital and analog module)
  - Reference monitors JVC and Sony with waveform and vectroscope
- **Production system**

  Video-audio system, Apple equipment and broadcasting infrastructure can be combined and conected in whole TV production system with Non-Linear editing and live channel production.

  Apple equipment is connected in cluster mode infrastructure to ensure efficient workflow.

Dag merilna kartica

- Točno časovno žigosanje datagramov
- Razvoj skupine The Waikato Applied Network Dynamics Group (WAND) od leta 1997 naprej
- Merjenje na OC-192 povezavah (10 Gbit/s)
- 64-bitni zapis časovnega žiga:
  - 32 bitov časovnega žiga je namenjeno štetju sekund (68 let)
  - 32 bitov pa predstavlja decimalni del (ločljivosti četrtinko ns)
- Urin impulz generira kristalni oscilator
  - frekvenca 33554432 Hz
  - pokriva 25 bitov decimalnega dela
  - časovno ločljivost okrog 30 ns
  - s stabilnostjo nekaj ppm
- Sinhronizacija; impulz vsako s
  - GPS
    - natančnost, boljšo od 100 ns
  - CDMA
    - nedoločljiva zakasnitev prehoda signala v celici



TKI

Sprint IP Monitoring System

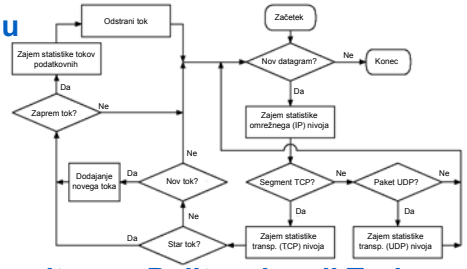
- Zelo zmogljiv sistem za naknadno analizo:
- Sistem za zajem podatkov: IPMON
  - Robne zmogljivosti današnjih osebnih računalnikov
- Oblika zapisa zajetih podatkov
  - Časovni žig
  - Prvih 44 oktetov datagrama IP: zajeta IP, navadno tudi glava transp. protokola
  - Skupaj – 64 oktetov
- Kapaciteta naprav za shranjevanje sledi in fizične zahteve sistema
  - Prenosne hitrosti na medijih:

	OC-3	OC-12	OC-48
podatkovna hitrost (Mb/s)	155	622	2480
velikost enourne sledi (GB)	11	42	176

  - Podpora merjenju velikim prenosnim hitrostim
    - Pisanje na disk (najslabši primer – OC48): 396,8 Mb/s = 50 MB/s
    - 5 SCSI diskov, RAID
    - 64 bitno vodilo PCI s 66 MHz uro 4224 Mb/s
  - Izvedba časovnega žigosanja paketov: Dag kartica;
  - Ocenjena napaka absolutne ure 5 µs
  - Časovna sinhronizacija (GPS), analiza zakasnitev

Tstat

- Sistem za tekočo pasivno analizo podatkov IP, UDP in TCP
- Razvit v letu 2001, predstavljen na GlobeCom-u 2002
- Podprtih več različnih zapisov sledi
- Problematiki časovnega žigosanja se ne posvečajo
  - Uporabljajo časovne žige osebnega računalnika
  - Uporabljajo časovne žige Dag katic
  - Promet zajemajo s `tcpdump`, `libpcab`
- Program napisan v C-ju, perlu



- Uporabili so ga za izvajanje meritev na Politecnico di Torino

Hvala za pozornost.

Vprašanja?

DPI – Deep Packet Inspection

mag. Roman Kotnik

Univerza v Ljubljani
Fakulteta za elektrotehniko
Laboratorij za telekomunikacije

Ljubljana, 10. maj, 2011

Razlogi za DPI

- Novi viri prihodkov
  - ISP-ji, podjetja
- Reševanje problemov tipa "ozko grlo"
- Zaznavanje vdorov
  - napad DOS
  - „Buffer overflow"
- Ciljno oglaševanje
- Izboljšanje kakovosti storitev
  - preprečevanje P2P prometa
  - povratne informacije od uporabnikov

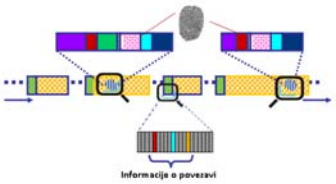
Kaj je DPI

- DPI – Deep Packet inspection
- Vpogled v vsebino komunikacij
  - pomembna tehnologija za identificiranje in preverjanje istovetnosti protokolov in aplikacij
  - analiza se izvaja glede na vsebino in ne glede na glavo IP-paketa – analogija s poštnim sporočilom

4-bit	8-bit	16-bit	32-bit
Različica	Dolžina glave	Tip storitve	Celotna dolžina
Identifikacija		Zastavice	Zamik
Čas življenja paketa	Protokol	Vsota za preverjanje paketa	
Naslov izvora			
Naslov ponora			
Možnosti in zamiki			
Vsebina oz. payload			

Osnovni mehanizmi za DPI

- Kaj je elektronski podpis?
  - elektronski podpis so vzorci, ki so izbrani za unikatno identifikacijo aplikacije ali protokola



- Napake pri podpisih
  - "False negative"
  - "False positive"

Kaj omogoča DPI

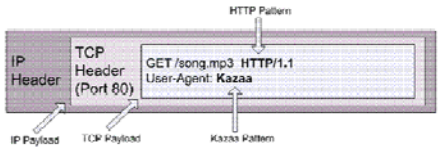
- Nadzor nad uporabniki in omrežjem
  - poznamo vsebino, posledično poznamo uporabnike in omrežje
- Zaračunavanje glede na vsebino
  - interes ISP – konflikt internetne nevtralnosti
- Nadzor prometa
  - prioritiziranje določenega tipa prometa
- Izboljšave elektronske varnosti
  - interes podjetij – preprečevanje napadov
  - preprečevanje uhajanja vsebin
- Legalni nadzor

Analiza glede na vrata

- Najenostavnejša in najbolj poznana
- Aplikacije večinoma uporabljajo privzeta vrata
  - POP3 – vrata 110
  - SMTP – vrata 125
- „Sindrom vrat 80"
  - Druge aplikacije delujejo na vratih 80 s tem namenom, da se pretvarjajo, kot da so HTTP promet.
- Nekatere aplikacije uporabljajo naključna vrata, da bi prikrila svoje delovanje
- Analiza ni zadostna in jo moramo uporabljati v kombinaciji z drugimi

Analiza glede na ujemanje z besedno zvezo

- Iskanje sekvenc tekstovnih znakov znotraj vsebine paketa
- Aplikacija, kot je npr. Kazaa, uporablja besedo „Kazaa“ znotraj HTTP GET zahteve
- Uporaba regularnih izrazov



Analiza glede na numerične lastnosti

- Preiskovanje aritmetičnih in numeričnih karakteristik znotraj paketa ali več paketov
  - dolžina vsebine paketa (payload)
  - število poslanih paketov
  - numerični zamik neke besede
- Primer povezava preko UDP v Skype

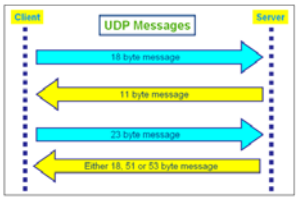


Figure 4: Skype (versions prior to 2.0) numerical properties analysis

Analiza glede na obnašanje in hevristiko

- Deluje glede na delovanje protokola
- Iskanje statističnih parametrov
- Iskanje vzorkov in posledic, ki se ujemajo
- Zahtevna analiza, ki zahteva poznavanje delovanja aplikacije
- Primerjava HTTP in P2P prometa

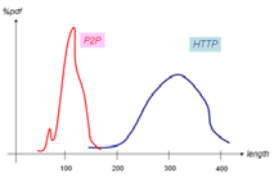


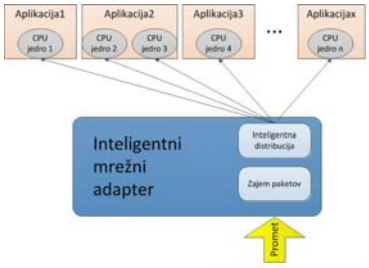
Figure 5: HTTP vs. P2P

Enkripcija in obfuskacija

- Namen: prikrivanje podatkov
- Enkripcija:
  - kodiranje
  - matematični algoritmi
  - uporaba ključa
  - RC4 enkripcija – Bittorrent
- Obfuskacija
  - „zameglitev“
  - dodajanje nepomembnih vsebin z namenom prikrivanja tiste, bistvene

Izvajanje DPI

- DPI sestavljen iz dveh glavnih funkcionalnosti
  - zajem paketov
  - procesiranje paketov
- Uporaba PC strežnikov



DPI za mobilne operaterje

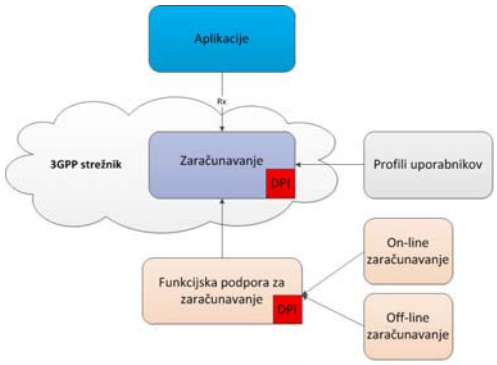
- Porast uporabe pametnih telefonov (HSDPA, HSUPA)
- Trenutno 5 milijard uporabnikov mobilnih telefonov
- Statistika prometa
  - 5 % uporabnikov uporablja 90% kapacitet
  - vsakih 6 mesecev se prenosi povečajo 2×
  - do 2014 – video predstavlja 66 % prometa
- Aplikacije so podatkovno požrešne (YouTube)
- Problematika zagotavljanja pasovne širine
- ISP-ji želijo poznati uporabnike, da lahko zagotovijo QOS
  - to jim omogoča DPI

DPI za mobilne operaterje

- Operaterji morajo zagotoviti pasovno širino in QOS
- Glavni izziv je zagotoviti delovanje omrežja z nizkimi investicijami
  - mesečno zaračunavanje je rizično – z večjo porabo porabnika ne dobimo več denarja – "efekt škarij"
- DPI omogoča mobilnim operaterjem, da se zavedajo vsebin, ki se pretakajo po njihovih omrežjih v realnem času
  - primer prenosa videa
- DPI "throttling"
  - povečevanje in zniževanje prioritet posameznega prometa

DPI in zaračunavanje

Implementacija DPI v mobilno omrežje



DPI pri podjetjih

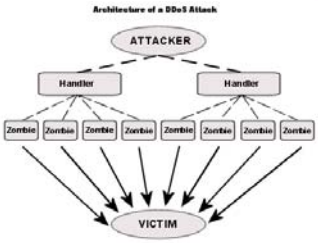
- Elektronska varnost
  - zaščita pred vdori, ki lahko povzročajo milijonske škode
- Nadzor nad zaposlenimi
- Preprečitev uhajanja informacij
- Zaščita avtorskih vsebin
- Proizvajalci DPI opreme
  - Cisco
  - Pcube
  - Nokia
  - Ericsson
  - Juniper
  - Ellacoya

Možnosti DPI pri zaračunavanju

- Trenutno deluje sistem le glede na prenesene podatke
  - 100 MB/mesec = x €, 1 GB/mesec = x + €
- Uporabniki, ki ne uporabljajo veliko aplikacij plačujejo za tiste, ki jih uporabljajo v velikih količinah
- S pomočjo DPI poznamo profil uporabnika
  - lahko mu zaračunamo glede na aplikacije, ki jih uporablja
    - lahko bi segmentirali posamezne pakete glede na aplikacije
      - npr: P2P paket, online gaming paket …
  - sistem cenejši za tiste, ki ne uporabljajo veliko aplikacij
- Možnost zaračunavanja glede na čas prenosa podatkov
  - npr. brezplačen prenos ponoči

Uporaba DPI – preprečevanje vdorov

- Požarni zidovi ne opazujejo vsebine paketov
  - napadi se selijo iz omrežnega nivoja na aplikacijski nivo
- S pomočjo DPI lahko zaznamo in izločimo promet, ki je potencialno nevaren
- Delovanje v realnem času
- IDS/IPS sistemi delujejo na osnovi DPI
- Napadi:
  - DOS
  - prenapolnjenje spomina

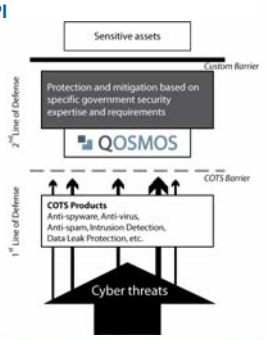


Uporaba DPI – Filtracija prometa

- Pri filtraciji pogledamo vsebino vsakega paketa in se na podlagi le-te odločimo, ali bomo paket pustili naprej ali ne
- Tipi filtriranja
  - statično – se ne zaveda ali je paket prvi ali zadnji ali vmes
  - dinamično – zavedanje ali gre za zahtevek ali odgovor, pomembno predvsem, ko je dovoljen UDP promet
- Klasifikacija prometa pri ponudnikih internetnih storitev
  - občutljiv promet – VoIP, Online gaming, video-konference
  - "Best-effort"– P2P, e-mail – pomembno samo, da pride čez
  - neželjen promet – spam, črvi, botnet, napadi
- Onemogočanje ilegalnih, piratskih vsebin

Uporaba DPI – Kibernetska varnost

- Poleg obstoječih aplikacij za zaščito lahko uporabimo tudi DPI
- Zaščita pred neželeno pošto, virusi in zlonamernimi programi (spyware, malware)



Uporaba DPI – Data retention

- Shranjevanje podatkov o povezavah
  - naslovniki in prejemniki poštnih sporočil
  - telefonski klici (VoIP)
  - obiskane spletne strani
- Glavni namen je analiza prometa in masovni nadzor
- Izvajajo ga vladne organizacije in večja podjetja
- Različno legalno ozadje v državah
  - EU – 6 mesecev do 2 leti shranjevanja
- Argumenti proti izvajanju
  - ni zasebnosti
  - veliko strojne opreme za shranjevanje vseh podatkov
  - zlouporaba

Dodatne uporabe DPI

- Starševski nadzor (parental control)
  - URL filtracija ni več dovolj
  - filtracija IM, P2P, IPTV
  - uporabnik sam nastavlja želene filtre
- Uveljavljanje uporabniške politike
  - nadzor nad avtentikacijami in avtorizacijami
- Filtriranje avtorsko zaščitenega materiala
  - veliko povpraševanja s strani YouTube za odstranjevanje avtorsko zaščitenih vsebin
- Ciljno oglaševanje
  - v kolikor poznamo strani, ki jih obiskuje uporabnik mu lahko ponudimo primerne oglase
- Legalni nadzor

Primeri uprabe DPI 1/3

- Proaktivna optimizacija storitev glede na navade in vrednost stranke/uporabnika
  - Pomembnost določenih storitev za uporabnika
  - Alarmiranje v primeru poslabšanja parametrov omrežja
  - Sprememba uporabnikovih nastavitev
    - Spremembe prioritet
  - Druge akcije glede na uporabnikovo vedenje
    - Prestavitev uporabnika v drugo omrežje (če je na voljo)
    - Alarmiranje (prenehanje uporabe storitve, nove naprave)
    - Možnost uporabe dodatnih storitev glede na obnašanje podobnih uporabnikov
  - Prilagajanje in ponujanje paketov za uporabnike ki uporabljajo malo storitev

Primeri uprabe DPI 2/3

- Starševski nadzor
  - Pametni telefoni in sirokopasovnia mobilna omrežja
  - Nadzor nad aktivnostjo otrok vedno bolj problematičen
  - Priložnost za razširitev in spremembo starševskega nadzora z uporabo DPI.
  - Operaterji lahko omogočijo:
    - Filtriranje neželjenih aplikacij in spletnih strani
    - Onemogočanje določenih aplikacij glede na lokacijo in čas
    - Obveščanje staršev o novih nevarnostih
    - Obveščanje o porabi
    - Obveščanje o neželjenih vsebinah
    - Analizo obnašanja uporabe interneta za otroke
  - Starševski nadzor kot aplikacija

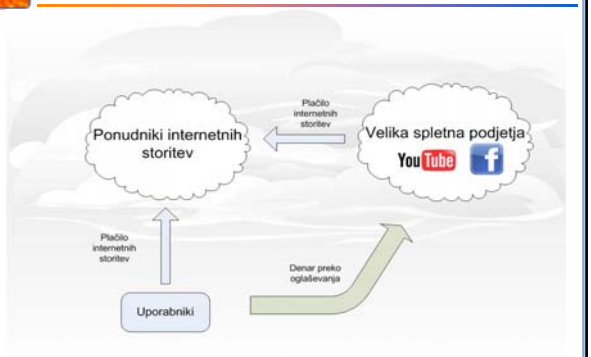
Primeri uprabe DPI 3/3

- Analize in podatki
  - Cilj mobilnih operaterjev je sodelovanje z zunanjimi ponudniki vsebin, razvijalci aplikacij in web portali
  - Operaterji imajo veliko podatkov ki so pomembni za razvijalce in ponudnike vsebin.
  - Vprašanja na katere lahko odgovorijo
    - Čas uporabe določene aplikacije.
    - QoE.
    - Kje je storitev uporabljena, na kateri napravi, kdaj?
    - Kaj uporabniki kupujejo, katere druge storitve uporabljajo?
    - Nasveti za vsebine in storitve
  - Informacije podlaga za
    - QoS
    - Ciljno oglaševanje
    - Avtomatsko prilagajanje na napravo ali lokacijo

Internetna nevtralnost

- Glavno načelo – ves promet na internetu je obravnavan enako
  - ISP-ji lahko prioritizirajo promet in sami določajo pravila po katerih bo internet deloval
  - imajo dober izgovor – preprečujejo napade
- Ekonomsko stališče
  - oskrba – pasovna širina – ISPji
  - povpraševanje – aplikacije – uporabiki
- ISP-ji prodajo več, kot lahko zagotovijo
  - iščejo rešitve za reševanje problema velikega porasta v povpraševanju po pasovni širini

Internetna nevtralnost



Primeri DPI aplikacij

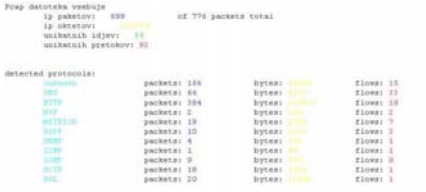
- Argus
  - nadziranje mrežnih storitev in strežnikov
  - odprtokodna rešitev
  - podpora IPv4 in IPv6, izrisovanje grafov
  - spletni vmesnik
- Arpwatch
  - spremlja ethernet – IP tabelo
  - uporaba libpcap
  - odprtokodna rešitev
- Barnyard
  - plugin za SNORT – pospešuje delovanje IDS/IPS sistema
- SILK (System for Internet-Level Knowledge)
  - za velika omrežja, srednje velike ISP

Primeri DPI aplikacij

- L7 – filter
- IPP2P
- HiPPIE
- nProbe
  - odprtokodna rešitev sonde za zaznavanje in zajemanje velike količine prometa
- Ntop
  - podobno kot linux ukaz top poraba mrežnih kapacitet
  - Libpcap, spletni vmesnik
- SANCP (Security Analyst Network Connection Profiler)
  - nadzira podatke o povezavah
  - zmožnost označevanja povezav
- OpenDPI

OpenDPI

- Odprtokodna DPI rešitev
- Prednosti
  - podpora za velik nabor protokolov
  - odprtokodna skupnost
- Slabosti
  - konzolno upravljanje
  - post-procesiranje



IWGDPT proti uporabi DPI

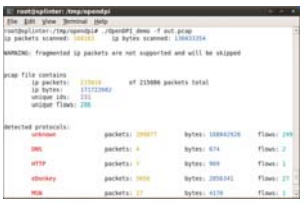
- Mednarodna delovna skupina za varstvo osebnih podatkov v telekomunikacijah (International Working Group on Data Protection in Telecommunications - IWGDPT),
- Mednarodna delovna skupina IWGDPT je v mnenju poudarila, da operaterji elektronskih komunikacij ne smejo na noben način posegati v zaupnost in celovitost komunikacije, če to ni izrecno dovoljeno ali zahtevano z zakonodajo. V luči navedenega mednarodna delovna skupina IWGDPT izrecno opozarja operaterje elektronskih komunikacij naj se ne poslužujejo uporabe DPI tehnologij za ciljno oziroma vedenjsko oglaševanj

Kontakt:

roman.kotnik@ltfe.org

OpenDPI

- **Odprtokodna DPI rešitev**
- **Prednosti**
  - **podpora za velik nabor protokolov**
    - **standardni (HTTP, FTP …), P2P, VoIP, IM, strujanje, tuneliranje, igralski …**
  - **odprtokodna skupnost**
- **Slabosti**
  - **konzolno upravljanje**
  - **post-procesiranje**
- **Delovanje obstoječe rešitve**
  - **zajem datoteke – Wireshark**
  - **izpis rezultatov OpenDPI**



Realne uporabe DPI danes

- **Traffic management control**
  - **Fair usage control/mandate/policy**
  - **IP session (DHCP model)**
  - **Assured QoS for third party applications and content (OTT)**
  - **Turbo button**
  - **Bill shock prevention (EU)**
- **Charging control**
  - **Variable charging and pricing (time of day, content downloads)**
  - **Quota limiting (EU mobile broadband roaming in FMC)**
- **Targeted advertising**
- **Security**
  - **Parental control**
  - **LI**
  - **Detection (DOS, DDOS, "buffer overflow")**

Upravljanje IP omrežij z uporabo SNMP

doc. dr. Iztok HUMAR
prof.dr. Janez BEŠTER

Upravljanje omrežij

- Upravljanje omrežij pomeni
  - razvijanje
  - integracijo
  - koordiniranje

  strojne in programske opreme ter človeških virov s cilji
  - nadzorovanja
  - testiranja
  - konfiguriranja
  - analiziranja
  - vrednotenja
  - krmiljenja

  omrežja in njegovih elementov, da z zmogljivostmi zagotavljajo delovanje v realnem času in v skladu z zahtevami QoS.

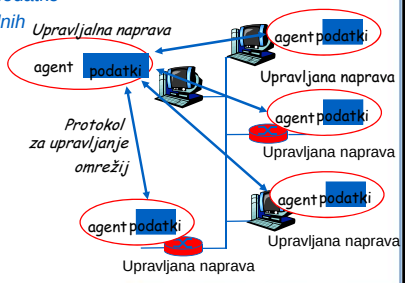
Agenda

- Upravljanje omrežij
  - namen
  - glavne komponente
- SNMP
  - MIB: management information base
  - SMI: data definition language
  - SNMP in komunikacija preko sporočil
  - varnost

Infrastruktura za upravljanje omrežij

- Definicije pojmov:

Upravljana naprava vsebuje *upravljane objekte* ki podatke zbirajo v *bazi upravljalnih podatkov (MIB)*



Upravljanje omrežij

- Omrežja in podobne avtonomne sisteme sestavlja več sto oziroma več tisoč med seboj sodelujočih enot, ki vsebujejo različno strojno in programsko opremo.
- Večina ostalih sistemov je večina krmiljena iz centralnega nadzornega in upravljalskega sistema. Primer:
  - jedrska elektrarna
  - letalo

Organizacije za standardizacijo

- IETF: Internet Engineering Task Force
  - Operation and Management Area:
    - Simple network Management protocol
- ISO: International Standardization Organization
  - ISO-IEC / JCT 1 / WG4:
    - Common Management Information Protocol
    - Common Management Information Service

Standardi za upravljanje omrežij

OSI CMIP
- Common Management Information Protocol
- Izdelan v osemdesetih – zamišljen kot poenoten standard za upravljanje omrežij
- Predolg proces pisanja standardov
- Zelo kompleksen
- Redka uporaba

IETF SNMP
- Simple Network Management Protocol
- Nastal zaradi Internet
- Prvotna standardizacija zelo enostavna
- Razvit in sprejet izjemno hitro
- Rast: velikosti in kompleksnosti protokola,
- Trenutno: SNMP v3
- *de facto* stanard za uporavljanje omrežij

Pregled SNMP: 4 ključni elementi

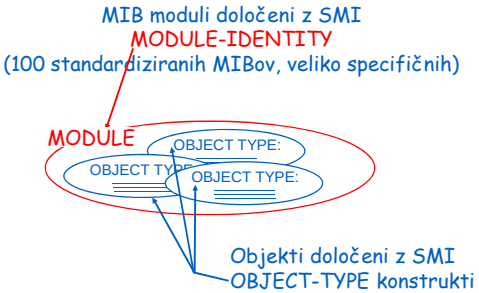
- **Management information base (MIB):**
  - porazdeljeno skladišče za informacije za upravljanje omrežja
- **Structure of Management Information (SMI):**
  - jezik za definicijo podatkov za MIB objekte
- **SNMP protokol**
  - prenos informacij in instrukcij med upravljalcem in upravljanimi objekti
- **Varnost, zmožnost administracije**
  - nove zmožnosti v SNMPv3

SNMP (RFC 1157) in njegova arhitektura

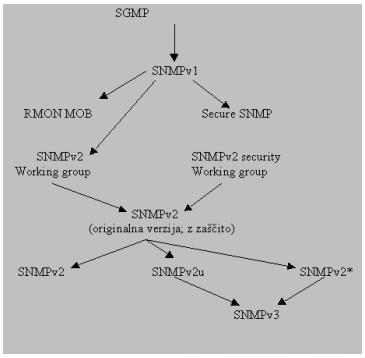
- SNMP = Simple Network Management Protocol
- Protokol za upravljanje TCP/IP in drugih omrežij
- Prednik: SGMP = Simple Gateway Monitoring Protocol (RFC 1028, 1987)
- Razvil IETF – Internet Engineering Task force, RFC 1157
- Razširil leta 1993

- Arhitektura odjemalec/strežnik (C/S)
- Odjemalec = Upravljalec
  - navadno aplikacija na osebnem računalniku
  - lahko implementiran na komunikacijsko napravo (sodeluje z drugimi)
- Strežnik (SNMP agent) = Upravljan
  - komunikacijska naprava (Router, Switch)
  - lahko tudi spletni strežnik, osebni računalnik

MIB



Razvoj protokola



SMI: Jezik za definicijo podatkov

- Namen: Nedvoumna in dobro definirana sintaksa in semantika za tvorjenje upravljalskih podatkov
- Osnovni podatkovni tipi:
  - Text, Integer, OCTET STRING, Opaque
  - Counter, Integer32, OBJECT IDENTIFIED,
  - Time Ticks Unsigned32, IPaddress,
- OBJECT-TYPE
  - vsebujejo statistične upravljalske podatke
  - podatkovni tip, status, semantika upravljanega objekta
  - diskretni, tabelirani
  - ASN.1 Notacija (Abstract Syntacx Notation)
- MODULE-IDENTITY
  - standardni nizi, definirani v RFC 1213
  - skupina logično povezanih objektov je združena v MIB modul
  - za posamezne naprave definirane vnaprej, lahko pa jih dodajamo sami

SMI primer: definicija objekta, modula

OBJECT-TYPE: ipInDelivers

```
ipInDelivers OBJECT TYPE
  SYNTAX       Counter32
  MAX-ACCESS   read-only
  STATUS   current
  DESCRIPTION
    "The total number of input
     datagrams successfully
     delivered to IP user-
     protocols (including ICMP)"
::= { ip   9}
```

MODULE-IDENTITY: ipMIB

```
ipMIB MODULE-IDENTITY
  LAST-UPDATED "941101000Z"
  ORGANZATION "IETF SNPv2
               Working Group"
  CONTACT-INFO
    " Keith McCloghrie
    ....."
  DESCRIPTION
    "The MIB module for managing IP
     and ICMP implementations, but
     excluding their management of
     IP routes."
  REVISION "019331000Z"
  .........
::= {mib-2 48}
```


MIB primer: UDP modul

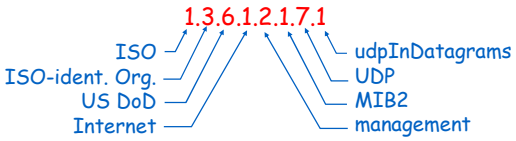
Object ID	Name	Type	Comments
1.3.6.1.2.1.7.1	UDPInDatagrams	Counter32	total # datagrams delivered at this node
1.3.6.1.2.1.7.2	UDPNoPorts	Counter32	# underliverable datagrams no app at portl
1.3.6.1.2.1.7.3	UDInErrors	Counter32	# undeliverable datagrams all other reasons
1.3.6.1.2.1.7.4	UDPOutDatagrams	Counter32	# datagrams sent
1.3.6.1.2.1.7.5	udpTable	SEQUENCE	one entry for each port in use by app, gives port # and IP address

Poimenovanja v SNMP

vprašanje: **kako poimenovati vsak možen standardni objekt (protokol, podatek, ...) v vsakem izmed razvitih omrežnih standardov?**

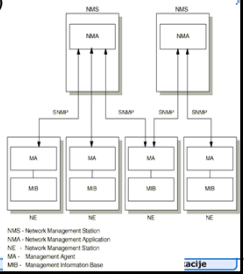
odgovor: ***ISO Object Identifier tree:***

- **hierarhično poimenovanje vseh objektov**
- **vsaka veja ima svoje ime, številko**



Delovanje SNMP

- **Strežnik ima podatkovno bazo MIB (Managment Information Base)**
- **Upravljalec pošilja ukaze, ki vsebujejo**
  - **MIB identifikator (ime MIB spremenljivke)**
  - **navodilo, kaj storiti s spremenljivko**
  - **dostop s Comunity Name, določi admin**



Poimenovanja v SNMP

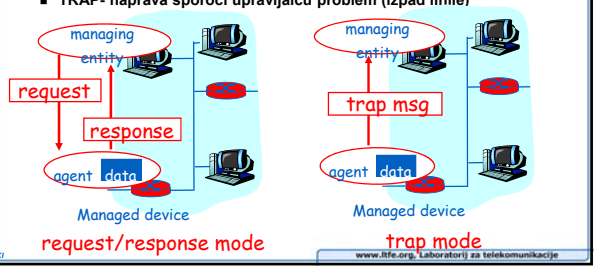
- **Več na: http://www.alvestrand.no/harald/objectid/top.html**

Dostop do MIB

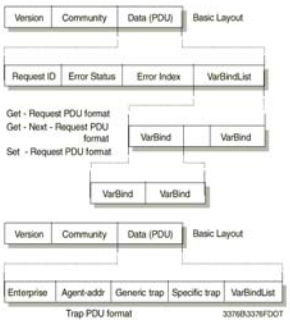
- **Dostop do MIB objektov:**
  - **RO**
  - **RW**
  - **WO**
- **Struktura (standard):**
  - **mgmt**
  - **private**
  - **experimental & directory**

Formati sporočila

- Sporočila kodirana v PDU (protocol data unit)
- Standardni formati sporočila SNMP
  - GET – prebere vrednost niza; ugotovi stanje naprave
  - GET NEXT – preleti vse vrednosti objektov (ugotovi imena)
  - SET – izvajamo aktivnosti (onemogočimo vmesnike), konfiguracija
  - TRAP- naprava sporoči upravljalcu problem (izpad linije)



SNMP sporočilo (II):



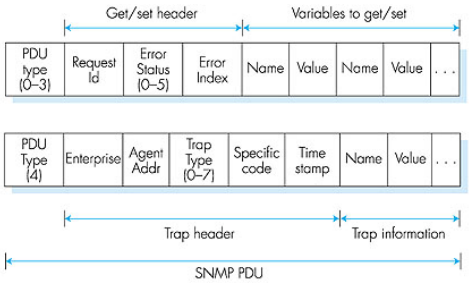
SNMP protocol: message types

Tip sporočila	Namen
GetRequest GetNextRequest GetBulkRequest	Mgr-to-agent: "želim podatke" (instance, next in list, block)
InformRequest	Mgr-to-Mgr: pošiljajo vrednosti MIB
SetRequest	Mgr-to-agent: nastavi vrednost MIB
Response	Agent-to-mgr: vrednost, odziv na zahtevo.
Trap	Agent-to-mgr: obvesti mngr o problematičnem dogodku

Prednosti / slabosti

- Prednosti:
  - velika popularnost
  - fleksibilnst in razširljivost
- Slabosti
  - kljub "Simple" je relativno kompliciran (SNMP–v2)
  - ni učinkovit (velika količina redundančnih informacij)

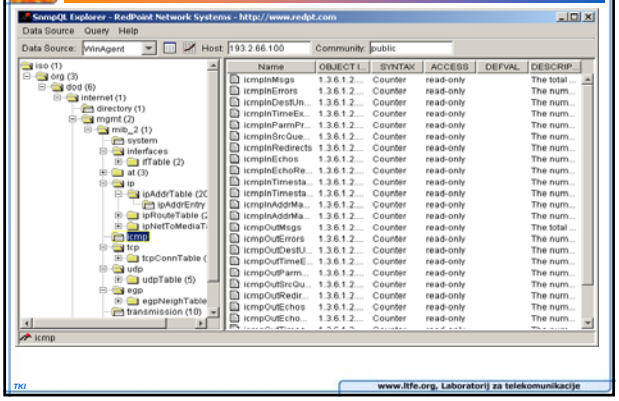
SNMP sporočilo (I):



SNMP v3: varnost in administracija

- Šifriranje: DES-šifriranje SNMP sporočilo
- Avtentikacija:
  - izračunamo in pošljemo MIC(m,k):
  - izračunamo hash funkcijo (MIC) over message (m),
  - skriti ključ (k), ki ga poznata obe strani
- Zaščita proti ponovni uporabi ključev
- Kontrola dostopa do objektov:
  - SNMP entiteta vzdržuje podatkovno bazo o pravicah za dostop do objektov, (različne pravice za različne uporabnike)
  - podatkovna baza se uporablja kot ostali objekti

Primer uporabe



HVALA ZA POZORNOST,
VPRAŠANJA !?

Primer uporabe: mrtg

- Aplikacija za zbiranje statističnih podatkov
- Podatke zajema preko SNMP protokola
- Podatke zajemamo v zaporednih časovnih intervalih
- Zajemamo podatke
  - o prometu
  - o času delovanja naprave
- Statistično obdelani, prikazani s pomočjo grafov
- Opazujemo lahko:
  - promet v omrežju
  - izpade naprav
- Možnost prirejanja različnim aplikacijam (customization)