

# Analisa Kinerja Sistem

## Seleksi Teknik dan Metrik

(Chapter 3)

1

## Sekilas

- Satu atau lebih sistem, nyata atau hipotetikal
- Ingin mengevaluasi kinerjanya
- Teknik apa yang dipilih?
  - Pemodelan analitik?
  - Simulasi?
  - Pengukuran?

2

## Outline

- Menyeleksi Teknik Evaluasi (berikut)
- Memilih Metrik Kinerja
  - Studi kasus
- Metrik Kinerja yang Umum Digunakan
- Menentukan Requirement Kinerja
  - Studi Kasus

3

## Menyeleksi Teknik Evaluasi (1/4)

- Apa tahap life-cycle sistem?
  - *Pengukuran* hanya ketika sesuatu ada
  - Jika baru, opsinya hanyalah *pemodelan analitik* atau *simulasi*
- Kapan hasil dibutuhkan? (seringnya, kemarin!)
  - Pilihannya hanya *pemodelan analitik*
  - *Simulasi* dan *pengukuran* dapat sama
    - Tetapi hukum Murphy menekankan *pengukuran*
- Tool dan skill apa yang tersedia?
  - Mungkin bahasa untuk mendukung *simulasi*
  - Tool untuk mendukung *pengukuran* (mis: packet sniffers, source code untuk menambahkan monitoring)
  - Skills dalam *pemodelan analitik* (mis: teori antrian)

4

## Menyeleksi Teknik Evaluasi (2/4)

- Level akurasi diminta?
  - *Pemodelan analitik* kasar (jika ternyata akurat, analystnya juga mungkin terkejut!)
  - *Simulasi* lebih detail, tetapi abstraksi komponen sistem lebih detail
  - *Pengukuran* terdengar lebih nyata, tetapi workload, konfigurasi, dll., mungkin masih hilang
    - Akurasi dapat tinggi sampai rendah tanpa desain yang memadai
  - Meskipun dengan data yang akurat, masih perlu untuk menghasilkan kesimpulan yang sesuai
    - Mis: maka waktu tanggap adalah 10.2351 dengan 90% confidence. Jadi bagaimana? Apa yang dimaksud dengan ini?

5

## Menyeleksi Teknik Evaluasi (3/4)

- Apa alternatifnya?
  - Dapat mengeksplorasi trade-off lebih mudah dengan *pemodelan analitik*, *simulasi moderat*, *pengukuran* paling sulit
    - Mis: QFind - menyatakan pengaruh (tradeoff) dari RTT dan OS
    - Sulit untuk mengukur RTT tradeoff
    - Mudah untuk mensimulasi RTT tradeoff dalam network, tidak OS
- Biaya?
  - *Pengukuran* secara umum paling mahal
  - *Pemodelan analitik* termurah (pensil dan kertas)
  - *Simulasi* umumnya murah tetapi beberapa tools mahal
    - Generator traffic, simulator jaringan

6

## Menyeleksi Teknik Evaluasi (4/4)

- Saleabilitas?
  - Lebih mudah meyakinkan orang menggunakan dengan *pengukuran*
  - Kebanyakan orang skeptis terhadap hasil *pemodelan analitik* karena sulit untuk dipahami
    - Terkadang validasi dengan simulasi sebelum digunakan
- Dapat menggunakan satu atau lebih teknik
  - Validasi satu dengan lainnya
  - Kebanyakan tulisan analisa kinerja kualitas-tinggi memiliki model analitik + simulasi atau pengukuran

7

## Tabel Ringkasan untuk Seleksi Teknik Evaluasi

Kriteria	Pemodelan	Simulasi	Pengukuran
1. Tahap	Segala	Segala	Prototipe+
2. Waktu	Sedikit	Medium	Bervariasi
3. Tools	Analysts	Beberapa bahasa	Instrumentasi
4. Akurasi	Rendah	Moderat	Bervariasi
5. Trade-off evaluasi	Mudah	Moderat	Sulit
6. Biaya	Kecil	Medium	Tinggi
7. Saleabilitas	Rendah	Medium	Tinggi

8

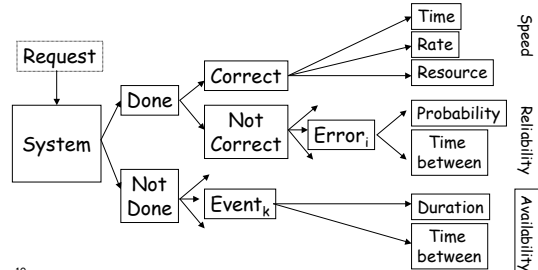
## Outline

- Menyeleksi Teknik Evaluasi (sudah)
- Menyeleksi Metrik Kinerja
  - Studi kasus
- Metrik Kinerja yang Umum Digunakan
- Menentukan Requirement Kinerja
  - Studi Kasus

9

## Menyeleksi Metrik Kinerja (1/3)

**response time** n. An unbounded, random variable ... representing the elapses between the time of sending a message and the time when the error diagnostic is received. – S. Kelly-Boote, *The Devil's DP Dictionary*



10

## Menyeleksi Metrik Kinerja (2/3)

- *Mean* yang biasanya diperhatikan
  - Tetapi *variance* untuk beberapa (mis: response time)
- Individual vs. Global
  - Meningkatkan individual dapat menurunkan global
    - Mis: response time pada biaya throughput
  - Meningkatkan global belum tentu yang paling rasional
    - Mis: throughput of cross traffic
- Optimisasi kinerja bottleneck paling berpengaruh,
  - Misal: Response time dari Web request
  - Client processing 1s, Latency 500ms, Server processing 10s → Total adalah 11.5 s
  - Meningkatkan client 50%? → 11 s
  - Meningkatkan server 50%? → 6.5 s

11

## Menyeleksi Metrik Kinerja (3/3)

- Mungkin lebih dari satu set metrik
  - Sumberdaya: Ukuran antrian, utilisasi CPU, Penggunaan memori ...
- Kriteria untuk menyeleksi subset, pilih:
  - Variabilitas rendah - memerlukan pengulangan lebih sedikit
  - Non redundancy - jangan menggunakan 2 jika satu cukup
    - Mis: ukuran antrian dan delay mungkin memberikan informasi yang identik
  - Completeness - harus menangkap tradeoff
    - Mis: satu disk mungkin lebih cepat tetapi mungkin memberikan lebih banyak error, maka tambahkan pengukuran reliabilitas

12

## Outline

- Menyeleksi Teknik (sudah)
- Menyeleksi Metrik Kinerja (sudah)
  - Studi Kasus (berikut)
- Metrik Kinerja yg Umum Digunakan
- Menentukan Requirement Kinerja
  - Studi Kasus

13

## Studi Kasus (1/5)

- Sistem komputer dari end-hosts mengirim paket lewat routers
  - Tumbukan terjadi ketika jumlah paket pada router melampaui kapasitas buffering (dropped)
- Goal: membandingkan dua algoritma kontrol tumbukan
- User sends blok paket ke destinasi
  - A) Beberapa terkirim berurut
  - B) Beberapa terkirim tidak berurut
  - C) Beberapa terkirim lebih dari satu kali
  - D) Beberapa dropped

14

## Studi Kasus (2/5)

- Untuk A), metrik:
  - 1) Response time: delay untuk paket individual
  - 2) Throughput: jumlah paket per unit waktu
  - 3) Waktu prosesor per paket pada sumber
  - 4) Waktu prosesor per paket pada destinasi
  - 5) Waktu prosesor per paket pada router
- Karena response time yg besar dapat mengakibatkan retransmisi ekstra:
  - 6) Variabilitas dalam response time dapat menyebabkan retransmisi ekstra

15

## Studi Kasus (3/5)

- Untuk B), tidak dapat disampaikan ke user dan sering dianggap dropped
  - 7) Probabilitas kedatangan diluar urutan
- Untuk C), menyita sumberdaya tanpa menggunakannya
  - 8) Probabilitas duplikasi paket
- Untuk D), alasan tidak jelas
  - 9) Probabilitas paket hilang
- Juga, loss yang berlebihan dapat menyebabkan diskoneksi
  - 10) Probabilitas diskoneksi

16

## Studi Kasus (5/5)

- Karena sistem multi-user dan ingin fairness:
  - Throughputs ( $x_1, x_2, \dots, x_n$ ):
$$f(x_1, x_2, \dots, x_n) = (\sum x_i)^2 / (n \sum x_i^2)$$
- Index antara 0 dan 1
  - Jika semua user mendapat sama, maka 1
  - Jika  $k$  user mendapat sama dan  $n-k$  mendapat nol, maka indeks adalah  $k/n$

17

## Studi Kasus (5/5)

- Setelah beberapa eksperimen (pilot tests)
  - Ditemukan *throughput* dan *delay* redundant
    - Lebih tinggi throughput memiliki lebih tinggi delay
    - Selain itu, gabungkan: *power* = thput/delay
  - Ditemukan variance dlm response time dengan *probabilitas duplikasi dan probabilitas diskoneksi*
    - Drop variance dalam response time
- Maka, tinggal sembilan metrik

18

## Outline

- Menyeleksi Teknik (sudah)
- Menyeleksi Metrik Kinerja (sudah)
  - Studi kasus (sudah)
- Metrik Kinerja yang Umum Digunakan (berikut)
- Menentukan Requirement Kinerja
  - Studi Kasus

19

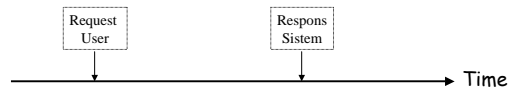
## Metrik Kinerja yang Umum Digunakan

- Response Time
  - Turn around time
  - Reaction time
  - Stretch factor
- Throughput
  - Operations/second
  - Capacity
  - Efficiency
  - Utilization
- Reliability
  - Uptime
  - MTTF (Mean Time to Failure)

20

## Response Time (1/2)

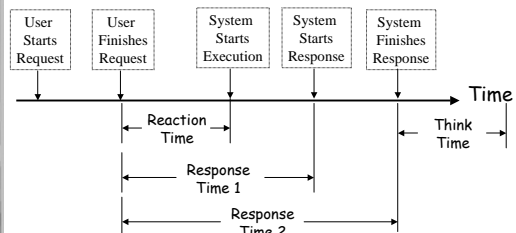
- Interval antara request user dan respons sistem



- Tetapi terlalu sederhana karena request dan respons tidak instant
  - Tipe user dan format sistem

21

## Response Time (2/2)



- Terdapat 2 pengukuran response time
  - Keduanya ok, tetapi yg ke 2 dianjurkan jika eksekusinya panjang
- *Think time* dapat menentukan load sistem

22

## Response Time+

- *Turnaround time* - waktu antara submisi job dan penyelesaian output
  - Untuk batch job systems
- *Reaction time* - Waktu antara submisi request dan mulainya eksekusi
  - Biasanya perlu untuk mengukur didalam sistem karena secara eksternal tidak tampak
- *Stretch factor* - rasion response time pada load terhadap response time pada load minimal
  - Kebanyakan sistem memiliki response time yang lebih tinggi begitu load meningkat

23

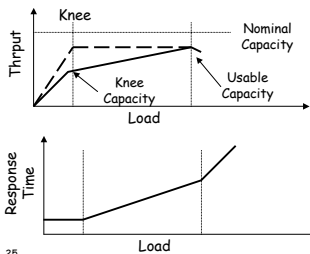
## Throughput (1/2)

- Rate dimana request dapat dilayani oleh sistem (requests per unit time)
  - Batch: jobs per second
  - Interaktif: requests per second
  - CPU
    - Millions of Instructions Per Second (MIPS)
    - Millions of Floating-Point Ops per Sec (MFLOPS)
  - Networks: pkts per second atau bits per second
  - Transactions processing: Transactions Per Second (TPS)

24

## Throughput (2/2)

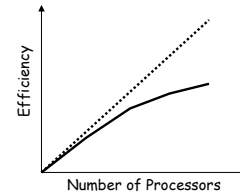
- *Throughput* naik dengan naiknya load, ke titik
- *Nominal capacity* adalah ideal (mis: 10 Mbps)
- *Usable capacity* adalah yang dapat dicapai (mis: 9.8 Mbps)
- *Knee* dimana response time naik dengan cepat untuk kenaikan sedikit dalam throughput



25

## Efisiensi

- Rasion throughput maksimum yang dapat dicapai (mis: 9.8 Mbps) terhadap kapasitas nominal (mis: 10 Mbps) → 98%
- Untuk multiprocessor, rasio  $n$ -processor terhadap satu-processor (in MIPS or MFLOPS)



26

## Utilisasi

- Biasanya, fraksi waktu sumberdaya sibuk untuk melayani request
  - Waktu tidak digunakan adalah *idle time*
  - Manager sistem sering ingin untuk menyeimbangkan sumberdaya agar memiliki utilisasi yang sama
    - Mis: load equal pada masing-masing CPU
    - Tetapi terkadang tidak memungkinkan. Mis: CPU ketika I/O bottleneck
- Mungkin bukan waktu
  - Processors - sibuk / total realistis
  - Memori - fraksi terpakai / total realistis

27

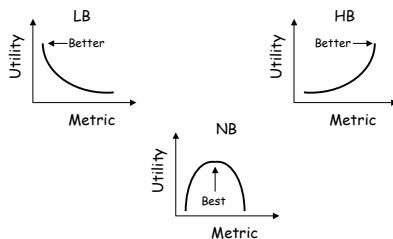
## Metrik Miscellaneous

- *Reliabilitas*
  - Probabilitas error atau mean time antara error (error-free seconds)
- *Availabilitas*
  - Fraksi waktu sistem tersedia untuk request service (fraksi tidak tersedia adalah *downtime*)
  - Mean Time To Failure (MTTF) adalah mean uptime
    - Berguna, karena availabilitas tinggi (downtime kecil) bisa sering terjadi dan tidak baik untuk request panjang
- Rasion Cost/Performance
  - Total cost / Throughput, untuk membandingkan 2 sistem
  - Mis: Untuk Transaction Processing system menginginkan Dollars / TPS

28

## Klasifikasi Utilitas

- *HB* - Higher is better (mis: throughput)
- *LB* - Lower is better (mis: response time)
- *NB* - Nominal is best (mis: utilization)



29

## Outline

- Menyeleksi Teknik (sudah)
- Menyeleksi Metrik Kinerja (sudah)
  - Studi kasus (sudah)
- Metrik Kinerja yang Umum Digunakan (sudah)
- Menentukan Requirement Kinerja (berikut)
  - Studi Kasus (berikut)

30

## Menentukan Requirement Kinerja (1/2)

- *Sistem seharusnya efisien dalam pemrosesan dan memori. Seharusnya tidak menciptakan overhead yang berlebihan*
- *Seharusnya memiliki probabilitas yang sangat rendah untuk network dapat menduplikasi paket, mengirimkan ke alamat yang salah, atau mengubah data*
- Mengapa?

31

## Menentukan Requirement Kinerja (2/2)

- Problem General
  - *Nonspecific* - tidak ada angka. Hanya kata kualitatif (jarang, rendah, tinggi sangat kecil)
  - *Nonmeasurable* - tidak ada cara untuk mengukur dan memverifikasi apakah sistem sesuai dengan requirement
  - *Nonacceptable* - angka berdasarkan pada apa yang tampaknya baik, tetapi sistem setup tidak cukup baik
  - *Nonrealizable* - angka berdasarkan pada apa yang tampaknya baik, tetapi setelah disetup terlalu tinggi
  - *Nonthorough* - tidak ada usaha yang dibuat untuk menentukan semua outcomes

32

## Menentukan Requirement Kinerja : Studi Kasus (1/2)

- Kinerja untuk high-speed LAN
- Speed - jika paket terkirim, waktu yang dibutuhkan penting
  - A) Delay akses harus lebih rendah dari 1 sec
  - B) Sustained throughput paling tidak 80 Mb/s
- Reliabilitas
  - A) Prob bit error lebih rendah dari  $10^{-7}$
  - B) Prob frame error lebih rendah dari 1%
  - C) Prob frame error yang tidak tertangkap  $10^{-15}$
  - D) Prob frame miss-delivered karena error yang tidak tertangkap  $10^{-18}$
  - E) Prob duplikasi  $10^{-5}$
  - F) Prob losing frame lebih rendah dari 1%

33

## Menentukan Requirement Kinerja : Studi Kasus (2/2)

- Availabilitas
  - A) Mean time to initialize LAN < 15 msec
  - B) Mean time between LAN inits > 1 minute
  - C) Mean time to repair < 1 hour
  - D) Mean time between LAN partitions >  $\frac{1}{2}$  week
- Semua nilai diatas harus diuji untuk realizeabilitas dengan pemodelan, menunjukkan bahwa sistem LAN memungkinkan untuk memenuhi requirement

34