

Analisa Kinerja Sistem

Pendahuluan

(Chapters 1 dan 2)

Buku Acuan

- Raj Jain. The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling, John Wiley and Sons, Inc., New York, NY, 1991.
- David J. Lilja. Measuring Computer Performance: A Practitioner's Guide, Cambridge University Press, New York, NY, 2000
- Douglas C. Montgomery. Design and Analysis of Experiments, 5th Edition, Wiley Text Books, June 30, 2000

Outline

- Objektif (berikut)
- Seni
- Kesalahan Umum
- Pendekatan Sistematik
- Studi Kasus

Objektif (1/6)

- Memilih *teknik* evaluasi, *metrik* kinerja dan *workload* untuk sistem yang tepat.
 - Teknik: pengukuran, simulasi dan pemodelan analitik
 - Metrik: kriteria untuk studi kinerja (mis: waktu respons)
 - Workloads: permintaan dari user/aplikasi terhadap sistem
- Contoh: Metrik kinerja apa yang seharusnya digunakan untuk sistem berikut?
 - a) Dua disk drives
 - b) Dua sistem pemrosesan transaksi
 - c) Dua algoritma retransmisi paket

Objektif (2/6)

- Melaksanakan pengukuran kinerja dengan benar
 - Membutuhkan dua tools: *load generator* dan *monitor*
- Contoh: Workload mana yang tepat untuk mengukur kinerja sistem berikut?
 - a) Utilisasi pada LAN
 - b) Waktu respons dari Web server
 - c) Kualitas audio dalam VoIP network

Objektif (3/6)

- Menggunakan teknik statistik yang sesuai untuk membandingkan beberapa alternatif
 - Menjalankan satu workload terkadang tak mencukupi
 - Banyak events komputer yg non-deterministik yang mempengaruhi kinerja
 - Membandingkan beberapa workload yang berjalan juga belum tentu mengarah kehasil yang benar
 - Khususnya jika variansnya tinggi
- Contoh: Packets lost pada link. Link mana yang lebih baik?

File Size	Link A	Link B
1000	5	10
1200	7	3
1300	3	0
50	0	1

Objektif (4/6)

- Pengukuran desain dan eksperimen simulasi untuk menghasilkan informasi dengan usaha yg sedikit.
 - Terkadang banyak faktor yg mempegaruhi kinerja. Memisahkan efek dari problem individual.
- Contoh: Kinerja sistem tergantung dari tiga faktor:
 - A) teknik garbage collection: G1, G2 tidak ada
 - B) tipe workload: editing, compiling, AI
 - C) Tipe CPU: P2, P4, SparcBerapa eksperimen yg dibutuhkan? Bagaimana kinerja dari tiap faktor dapat diestimasi?

Objektif (5/6)

- Melakukan simulasi dengan benar
 - Pilih bahasa yg tepat, Select correct language, mulai untuk random numbers, lamanya simulasi, dan analisa
 - Sebelum itu semua, mungkin perlu untuk memvalidasi simulator
- Contoh: Untuk membandingkan kinerja dua algoritma replacement cache:
 - A) berapa lama simulasi seharusnya dijalankan?
 - B) apa yg dapat dilakukan untuk mendapatkan akurasi yg sama dengan simulasi yg lebih singkat?

Objektif (6/6)

- Menggunakan model antrian sederhana untuk menganalisa kinerja sistem.
- Terkadang dapat memodelkan sistem komputer dengan rate service dan rate kedatangan load
 - Multiple servers
 - Multiple antrian
- Contoh:
 - Untuk Web request rate tertentu, apakah lebih efektif menggunakan 2 single-processor Web servers atau 4 single-processor Web servers?

Outline

- Objektif (sudah)
- Seni (berikut)
- Kesalahan Umum
- Pendekatan Sistematik
- Studi Kasus

Seni Evaluasi Kinerja

- Evaluasi tidak dapat dibuat secara mekanik
 - Membutuhkan pengetahuan detail sistem
 - Pemilihan metodologi, workload, tools dgn hati-hati
- Tidak ada satu jawaban yg benar, dua analyst kinerja dapat memilih metrik atau workload yang berbeda
- Seperti seni, ada teknik yg dapat dipelajari:
 - Bagaimana menggunakannya
 - Kapan menerapkannya

Contoh: Membandingkan Dua Sistem

- Dua sistem, dua workload, ukuran: transaksi per detik

Sistem	Work-load 1	Work-load 2
A	20	10
B	10	20

- Mana yang lebih baik?

Contoh: Membandingkan Dua Sistem

- Dua sistem, dua workload, ukuran: transaksi per detik

Sistem	Work-load 1	Work-load 2	Average
A	20	10	15
B	10	20	15

- Keduanya sama baiknya!
- ... tetapi apakah A lebih baik dari B?

Permainan Rasio

- Mengambil sistem B sebagai basis

Sistem	Work-load 1	Work-load 2	Average
A	2	0.5	1.25
B	1	1	1

- A lebih baik!
- ... tetapi apakah B lebih baik dari A?

Outline

- Objektif (sudah)
- Seni (sudah)
- Kesalahan Umum (berikut)
- Pendekatan Sistematik
- Studi Kasus

Kesalahan Umum (1/3)

- Goal Tidak Didefinisikan
 - Tidak ada untuk model general
 - Deskripsikan goal dan mendesain eksperimen
 - (Jangan tembak dan kemudian gambar target)
- Goal Bias
 - Jangan tunjukkan sistem KITA lebih baik dari MEREKA
 - (Analisa Kinerja seperti Juri)
- Workload Tidak Representatif
 - Seharusnya representatif dari bagaimana sistem bekerja sesungguhnya
 - Contoh: paket besar dan kecil? Jangan menguji dengan besar atau kecil saja

Kesalahan Umum (2/3)

- Teknik Evaluasi Salah
 - Gunakan yang paling sesuai: model, simulasi, pengukuran
 - (Jangan gunakan palu dan melihat semua sebagai paku)
- Level Detail Tidak Tepat
 - Dapat lebih banyak! Mis: modeling disk
 - Dapat lebih sedikit! Mis: model analitik untuk congested router
- Analisa Tidak Sensitif
 - Analisa adalah *bukti* bukan kenyataan
 - Perlu menentukan sesensitif apa hasil di-set

Kesalahan Umum (3/3)

- Presentasi Hasil Tidak Tepat
 - Ini bukan banyaknya grafik, tetapi banyaknya grafik yang membantu pengambilan keputusan
- Mengabaikan Asumsi dan Limitasi
 - Mis: ,mengasumsikan kebanyakan traffic TCP, dimana beberapa link memiliki traffic UDP yg signifikan
 - Menerapkan hasil dimana asumsi yang tidak pasti

Outline

- Objektif (sudah)
- Seni (sudah)
- Kesalahan Umum (sudah)
- Pendekatan Sistematik (berikut)
- Studi Kasus

Pendekatan Sistematik

1. Nyatakan goal dan tentukan batas
2. Pilih metrik kinerja
3. List parameter sistem dan workload
4. Pilih faktor dan nilai
5. Pilih teknik evaluasi
6. Pilih workload
7. Eksperimen desain
8. Analisa dan interpretasi data
9. Berikan hasil. Ulangi.

Nyatakan Goal dan Tentukan Batas

- Hanya dengan "mengukur kinerja" atau "melihat bagaimana kerjanya" adalah terlalu umum
 - Mis: goal adalah menentukan ISP mana yg memberikan throughput lebih baik
- Definisi sistem dapat tergantung dari goal
 - Mis: jika mengukur kecepatan instruksi CPU, sistem mungkin termasuk CPU + cache
 - Mis: jika mengukur waktu respons, sistem mungkin termasuk CPU + memory + ... + OS + user workload

Pilih Metrik

- Kriteria untuk membandingkan kinerja
- Secara umum, berhubungan dengan kecepatan, akurasi dan/atau availabilitas pelayanan sistem
- Mis: kinerja network
 - Kecepatan: throughput dan delay
 - Akurasi: error rate
 - Availabilitas: paket data terkirim yang sampai
- Mis: kinerja prosesor
 - Kecepatan: waktu untuk mengeksekusi instruksi

List Parameters

- List semua parameter yang mempengaruhi kinerja
- Parameter sistem (hardware and software)
 - Mis: tipe CPU, tipe OS, ...
- Parameter workload
 - Mis: Jumlah user, tipe requests
- List mungkin tidak lengkap pada awalnya, maka miliki list kerja dan biarkan tumbuh sesuai kemajuan.

Pilih Faktor untuk Studi

- Bagi parameter sesuai dengan apa yang akan dipelajari dan yang mana yang tidak
 - Mis: Tipe CPU dpt bervariasi tapi OS tetap
 - Mis: ukuran paket tetap tapi jumlah koneksi bervariasi
- Pilih level yg sesuai untuk tiap faktor
 - Ingin tipikal dan yg secara potensial memiliki high impact
 - Untuk workload terkadang range lebih kecil (1/2 atau 1/10th) dan lebih besar (2x atau 10x)
 - Mulai dengan kecil atau sejumlah tertentu dapat melampaui resource yg tersedia!

Pilih Tekni Evaluasi

- Tergantung dari waktu, resources dan level akurasi yang diinginkan
- Pemodelan Analitik
 - Cepat, keakuratan lebih rendah
- Simulasi
 - Usaha medium, akurasi medium
- Pengukuran
 - Umumnya usahanya besar, akurasi tinggi
- Catatan, semua diatas tipikal tetapi dapat kebalikannya di kasus tertentu!

Pilih Workload

- Tentukan service yg diminta sistem
- Tergantung dari teknik pengukuran
 - Model analitik dapat memiliki probabilitas berbagai request
 - Simulasi dapat memiliki kesamaan permintaan dari real system
 - Pengukuran mungkin memiliki scripts yang mengganggu transaksi
- Seharusnya representatif untuk real life

Eksperimen Desain

- Ingin hasil maksimal dengan usaha minimal
- Fase 1:
 - Banyak faktor, level sedikit
 - Melihat faktor mana yang penting
- Fase 2:
 - Faktor sedikit, level lebih banyak
 - Melihat jangkauan pengaruh untuk faktor tertentu

Analisa dan Intepretasi Data

- Membandingkan alternatif
- Memasukkan variabilitas hasil dalam pertimbangan
 - Teknik statistikal
- Interpretasi hasil.
 - Analisa tidak menghasilkan kesimpulan
 - Beda analyst mungkin beda kesimpulan

Tampilkan Hasil

- Buat mudah untuk dipahami understood
- Grafik
- Diseminasi (keseluruhan metodologi!)

"Tugas scientist tidak hanya melihat: melihat, memahami dan mengkomunikasikannya. Mengabaikan salah satu fase ini kita bukanlah."

Outline

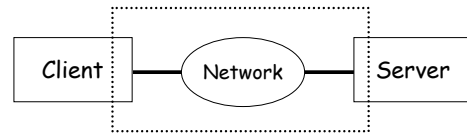
- Objektif (sudah)
- Seni (sudah)
- Kesalahan Umum (sudah)
- Pendekatan Sistimatik (sudah)
- Studi Kasus (berikut)

Studi Kasus

- Remote pipes (rpipe) versus remote procedure calls (rpc)
 - rpc seperti procedure call tetapi ditangani pada server remote
 - Client caller tertahan sampai return
 - rpipe seperti pipe tetapi server mendapatkan output pada mesin remote
 - Proses client dapat berlanjut, non-blocking
- Goal: studi kinerja aplikasi menggunakan rpipe dan aplikasi menggunakan rpc

Definisi Sistem

- Client, Server dan Network
- Komponen kunci "channel", baik rpipe atau rpc
 - Hanya subset client dan server yang menangani channel yang merupakan bagian dari sistem



- Usahakan untuk meminimalkan efek dari komponen diluar sistem

Service

- Ada beberapa service yang dapat terjadi menggunakan rpipe atau rpc
- Pilih transfer data sebagai yang umum, dengan data sebagai hasil tipikal untuk kebanyakan interaksi client-server
- Menentukan jumlah data besar atau kecil
- Maka, dua service:
 - Transfer data kecil
 - Transfer data besar

Metrik

- Batasi metrik untuk operasi yg benar saja (tdk ada failure atau errors)
- Pelajari rate service dan konsumsi resource
 - A) elapsed time per call
 - B) maximum call rate per unit waktu
 - C) Waktu CPU lokal per call
 - D) Waktu CPU remote per call
 - E) Jumlah byte terkirim per call

Parameter

Sistem	Workload
<ul style="list-style-type: none">• Kecepatan CPU<ul style="list-style-type: none">- Local- Remote• Network<ul style="list-style-type: none">- Kecepatan- Reliabilitas (retrans)• Operating system overhead<ul style="list-style-type: none">- Untuk interfacing dengan channel- Untuk interfacing dengan network	<ul style="list-style-type: none">• Waktu antar calls• Jumlah dan Ukuran<ul style="list-style-type: none">- parameter- hasil• Tipe channel<ul style="list-style-type: none">- rpc- Rpipe• Load lain<ul style="list-style-type: none">- Pada CPU- Pada network

Factor Kunci

- Tipe kanal channel
 - rpipe atau rpc
- Kecepatan network
 - Pilih dekat (LAN) lintas negara (WAN)
- Ukuran parameter
 - Kecil atau besar
- Banyaknya call
 - 11 nilai: 8, 16, 32 ...1024
- Parameter lainnya adalah tetap
- (Cat, coba jalankan saat load network "ringan")

Teknik Evaluasi

- Selama tidak ada prototipe, gunakan pengukuran
- Gunakan pemodelan analitik berdasarkan pada data terukur untuk nilai diluar jangkauan eksperimen yang dilakukan

Workload

- Program sintetik yg digenerasi menentukan request dari channel
- Juga memonitor resource yg dikonsumsi dan me-log hasil
- Gunakan request channel "null" untuk mendapatkan resource baseline yg dikonsumsi oleh logging
 - (Ingat prinsip Heisenberg!)

Desain Eksperimental

- Factorial penuh (semua kemungkinan kombinasi faktor)
- 2 channels, 2 kecepatan network, 2 ukuran, 11 banyaknya call
 - $\rightarrow 2 \times 2 \times 2 \times 11 = 88$ eksperimen

Analisa Data

- Analysis of variance akan digunakan untuk mengkuantifikasi tiga faktor pertama
 - Apakah mereka berbeda?
- Regresiakan digunakan untuk mengkuantifikasi efek dan n call yg konsekutif
 - Kinerja linier? atau eksponensial?