



Lingkungan Pengembangan Aplikasi Internet

Erick Kurniawan, S.Kom, M.Kom

<http://actualtraining.wordpress.com>

erick.kurniawan@gmail.com



Masalah

- HTTP bersifat stateless
- Jika kita belanja pada situs e-commerce misal www.amazon.com dan kita membeli beberapa buku dan memasukannya kedalam basket chart
- Kemudian kita berpindah halaman yang lain untuk melihat buku yang lain
- Bagaimana amazon dapat mengingat buku apa saja yang sudah kita beli yang ada dalam basket chart?
- Bagaimana kita menyimpan state dari request ke request yang lain



Solusi yang ditawarkan

- Bisa menggunakan log file yang dicatat dalam web server?
- Tetapi HTTP bersifat *anonymous*
- Web Server tidak tahu user siapa yang sedang terkoneksi ke web server / mengakses web
- Web Server hanya dapat mencatat IP komputer saja
- Bagaimana kalau koneksinya melalui *proxy* (misal AOL)



Solusi yang ditawarkan

- Bagaimanapun juga anda juga harus mencatat informasi dari masing-masing individu yang terkoneksi
- Menggunakan *session ID*
- Yaitu menambahkan ID tertentu pada alamat URL dari semua halaman yang diakses
- <http://www.amazon.com/exec/obidos/ASIN/1588750019/103-9609966-7089404>
- Biru = nomer ISBN
- Hijau = *session ID*



Solusi yang ditawarkan

- Solusi lain yang ditawarkan adalah dengan menggabungkan semua content yg ada di basket chart
- Menggabungkan nomor ISBN-nya (misal: dari 3 buku yang ada di chart) dan mengenkripsi datanya
- Kemudian informasi tersebut ditambahkan kedalam URL
- Masalah yang timbul : keterbatasan panjang URL
- Hanya 255 bytes untuk browser lama, browser baru bisa lebih



Kesimpulan yang diambil

- Amazon tetap menggunakan *session ID* untuk mengatasi masalah ini
- Catatan : Masalah ini timbul jika user mengakses web sites tidak harus login terlebih dahulu
- Jika sudah melalui login, web server dapat membuat nama session secara spesifik sesuai dengan user name nya



Cookies

- *Cookies* adalah mekanisme umum dimana server side connection (CGI script) dapat menyimpan dan mengambil informasi pada sisi client (client side)
- Bagaimana cara kerjanya?
- Setelah user menambahkan buku kedalam shopping cart, server akan menuliskan
- Set-Cookie:
`cart_contents=1588750019; path=/`



Cookies

- Selama user belum keluar dari browser, untuk setiap request yang dilakukan ke server, browser akan menambahkan header
- `Cookie: cart_contents=1588750019`
- Script pada sisi server akan membaca header ini kemudian menampilkan isinya kedalam shopping cart



Diskusi

- Apakah ini merupakan solusi yang tepat?
- Cara ini termasuk *distributed database management system*
- Daripada menyimpan file log yang besar dalam server (memberatkan server)
- Bayangkan jika harus menyimpan log dari jutaan user yang terkoneksi ke web server



Masalah 1

- Keterbatasan untuk menyimpan cookies dalam browser
- Tidak boleh lebih 20 cookies untuk web server yang sama
- Ukuran setiap cookies tidak boleh lebih dari 4 kilobytes



Masalah 2

- Informasi yang disimpan di cookies akan dikirimkan kembali ke web server setiap kali page di load
- Jika file cookies besar dan jumlahnya banyak (misal 80 Kilobytes untuk 20 cookies)
- Usernya menggunakan dial-up modem
- Maka loading halaman web menjadi lambat



Masalah 3

- Tidak portabel untuk user
- Jika user berganti komputer maka data cookies yang sudah ada tidak tersimpan pada komputer yang baru
- Data shopping cart tersimpan pada memory di komputer yang digunakan



Masalah 4

- Masalah terakhir adalah banyak user yang mematikan / men-disable kan fasilitas cookies ini dengan alasan privacy



Solusi

- Digabungkan dengan data log yang disimpan di server (seperti session ID pada contoh amazon)



```
bash-2.03$ telnet www.eveandersson.com 80
```

```
Trying 64.94.245.206...
```

```
Connected to www.eveandersson.com.
```

```
Escape character is '^]'.  
GET / HTTP/1.0
```

```
HTTP/1.0 200 OK
```

```
Set-Cookie: ad_browser_id=3291092; Path=/  
Expires=Fri, 01-Jan-2010 01:00:00 GMT
```

```
Set-Cookie:
```

```
ad_session_id=3291093%2c0%2c6634C478EF46FC%2c1062215  
8; Path=/  
Max-Age=3600
```

```
Set-Cookie: last_visit=1071622158; path=/  
expires=Fri, 01-Jan-2010 01:00:00 GMT
```

```
Content-Type:  
text/html; charset=iso-8859-1
```

```
MIME-Version: 1.0
```

```
Date: Thu, 03 Feb 2005 00:49:18 GMT
```

```
Server: AOLserver/3.3.1+ad13
```

```
Content-Length: 8289
```

```
Connection: close
```



Server-Side Storage

- Dengan cookies anda dapat mengambil informasi ID dan mengirimkannya ke server
- Sekarang anda harus menemukan cara agar informasi yang diambil dari browser (cookies) terhubung dengan web server



Server-Side Storage

- Apa yang bisa digunakan untuk menyimpan informasi dalam table?
- Spreadsheet program (excel, calc)
- Masalahnya adalah spreadsheet program didesain untuk single user
- Pada web server kemungkinan user yang membaca dan menulis data ke database bisa ribuan orang dan dapat terus bertambah



Server-Side Storage

- Cara yang paling tepat adalah menggunakan RDBMS (relational database management system)
- Dapat melakukan perintah create, insert, update, delete, dan select (*query*)
- Mengatasi masalah concurency, misal buku tinggal satu kopi tapi mau dibeli 1000 orang



Mengapa Menggunakan RDBMS?

- Adanya declarative query language (SQL)
- Biasanya bahasa pemrograman bersifat procedural, perintahnya step by step (if then statement)
- Dapat bertambah kompleks secara cepat
- Hanya dapat dikembangkan dan di maintain oleh professional programmer
- Alternative style yang lain adalah declarative
- Kita meminta kepada komputer apa yang diinginkan



Mengapa Menggunakan RDBMS?

- Misal 'cari user yang sudah terdaftar selama satu tahun tetapi tidak menjawab pertanyaan pada discussion forum'
- Kita tidak memerintahkan RDBMS untuk membaca tabel user terlebih dahulu kemudian membaca tabel discussion
- Kita hanya menentukan hasil report yang kita inginkan secara spesifik
- RDBMS yang akan mengaturnya



Mengapa Menggunakan RDBMS?

- Alasan yang kedua adalah adanya isolation untuk data yang penting dari kesalahan yang dilakukan oleh programmer
- RDBMS membatasi programmer dengan perintah (insert,update,delete)
- Dapat dilihat log-nya user siapa yang melakukan perintah tersebut (verifikasi user)



Mengapa Menggunakan RDBMS?

- Alasan yang ketiga adalah performa yang bagus untuk menangani banyak user (ribuan bahkan lebih) secara bersama-sama
- Jenis RDBMS: IBM DB2, Oracle, MS SQL Server, PostgreSQL



Langkah yang dilakukan

- Mengembangkan data model, informasi apakah yang akan disimpan dan bagaimana informasi tersebut akan ditampilkan
- Mengembangkan legal transaction (perintah yang diijinkan) pada model (table) tersebut (insert, update)
- Mendisain page flow, bagaimana user berinteraksi dengan sistem, perintah apa yang boleh dilakukan user / halaman yg boleh diakses
- Mengimplementasikan individual pages, menulis script (HTML dan bahasa server side) dan query information (data model)



Kesimpulan yang diperoleh

- Mau tidak mau kita harus menentukan pilihan untuk tools apa yang akan digunakan untuk penyimpanan data.
- Kita menggunakan RDBMS
- Untuk implementasi pembuatan halaman (pada step 4) kita harus menentukan pilihan penggunaan teknologinya
- Terlebih dahulu kita harus tahu dasar penggunaan HTML



Penjelasan HTML dan XHTML



Bagaimana HTML dibaca oleh browser?

- `<html>` tag menunjukkan bahwa 'ini adalah dokumen HTML' misal disimpan dengan `index.html`
- Ketika user merequest dokumen yang diminta, maka web server mengecek bahwa dokumen tersebut mempunyai ekstensi `.html`
- Web server menambahkan pada MIME (Multipurpose Internet Mail Extention) header untuk memberi tahu kepada browser tipe file yang dikirimkan oleh web server (`type='text/html'`)



Memilih programming environment

- Karena kita menggunakan RDBMS untuk backend aplikasi Internet maka kita harus menentukan RDBMS mana yang akan digunakan
- Ada beberapa pilihan, yang disarankan pada buku ini adalah (MS SQL Server, Oracle, dan PostgreSQL)



Membandingkan RDBMS

- Untuk programmer windows yang sudah berpengalaman, MS SQL Server merupakan RDBMS yang mudah diinstal dan di-admini
- Jika anda memutuskan untuk berkarir di lingkungan pengembangan berbasis Microsoft maka anda harus mempelajarinya
- Concurrency adalah salah satu keunggulan Oracle dibanding kompetitornya
- Reader tidak pernah menunggu writer dan writer tidak pernah menunggu reader

Membandingkan RDBMS

Time	Publisher	Public WebApp
12:00 PM	Memulai 90 menit query untuk mencetak laporan tahunan terdahulu	-
12:30 PM	Proses query masih berjalan	Ada user 'erick' mengupdate email dari 'erick@yahoo.com' menjadi 'erick@hotmail.com'
12.45 PM	Query sampai pada user 'erick', oracle menjalankan rollback segment dan menuliskan 'erick@yahoo.com' pada laporan	
1:30 PM	Query report selesai dilakukan	



Membandingkan RDBMS

- Jika kita menggunakan SQL Server maka:
- Ketika kita membaca table, maka secara otomatis table akan terkunci dan tidak dapat ditulis oleh user lain sampai kita selesai mengakses tabel tersebut
- Ketika kita menulis ke table, maka secara otomatis table akan terkunci, dan user lain tidak dapat membaca tabel tersebut sebelum kita selesai menulis ke table



Membandingkan RDBMS

- Alternatif lain menggunakan RDBMS yang bersifat open source (gratis untuk digunakan) yaitu PostgreSQL
- Lebih sulit digunakan daripada RDBMS komersial lainnya
- Mempunyai fasilitas concurrency mirip Oracle
- Tetapi tidak untuk sekala besar karena tidak ada yang berani menanggung resiko jika mengalami hardware failures
- Contoh sql pada buku ini menggunakan Oracle sintaks