

Desain dan Pemrograman Web

Semester 4

Sks : 3

Oleh : Lambang Probo Sumirat

Konsep Pemrograman Web

Konsep Web

World Wide Web ("WWW", atau singkatnya "Web") adalah suatu ruang informasi dimana sumber-sumber daya yang berguna diidentifikasi oleh pengenal global yang disebut Uniform Resource Identifier (URI).

WWW sering dianggap sama dengan Internet secara keseluruhan, walaupun sebenarnya hanyalah bagian daripadanya.

Hiperteks dilihat dengan sebuah program bernama browser web yang mengambil informasi(disebut"dokumen" atau"halamanweb") dari server web dan menampilkannya, biasanya disebuah monitor. Kita lalu dapat mengikuti pranala disetiap halaman untuk pindah ke dokumen lain atau bahkan mengirim informasi kembali kepada server untuk berinteraksi dengannya. Ini disebut "*surfing*" atau "*berselancar*" dalam bahasa Indonesia. Halaman web biasanya diatur dalam koleksi material yang berkaitan yang disebut "situsweb".

- Pemrograman web : membuat aplikasi berbasis web
- Aplikasi berbasis web : Aplikasi yang dibuat dengan memanfaatkan mekanisme dan aplikasi yang sudah ada pada sistem web (WWW)
- Sistem web sebenarnya merupakan aplikasi yang:
 - ber arsitektur client-server
 - software web browser disisi client
 - software web server disisi server
 - menggunakan protokol HTTP dalam komunikasi antara client dan server
 - mempunyai fungsi untuk mengambil / menjalankan isi file dokumen web di server dan menampilkannya disisi client

- Membuat aplikasi berbasis web berarti:

Memperkaya fungsi web server dengan cara menambahkan program pada dokumen web yang akan di eksekusi oleh server ketika file dokumen web tersebut diakses oleh web server.

Misalnya, program yang mengambil data ke basis data untuk ditampilkan ke web browser.

- Memperkaya interaktivitas dokumen dengan cara menambahkan program pada dokumen web yang akan dieksekusi oleh web browser ketika file dokumen tersebut ditampilkan oleh web browser.

Misalnya, program yang memvalidasi data masukan pada form sebelum disubmit ke web server

Kelebihan & Kekurangan Aplikasi Berbasis Web

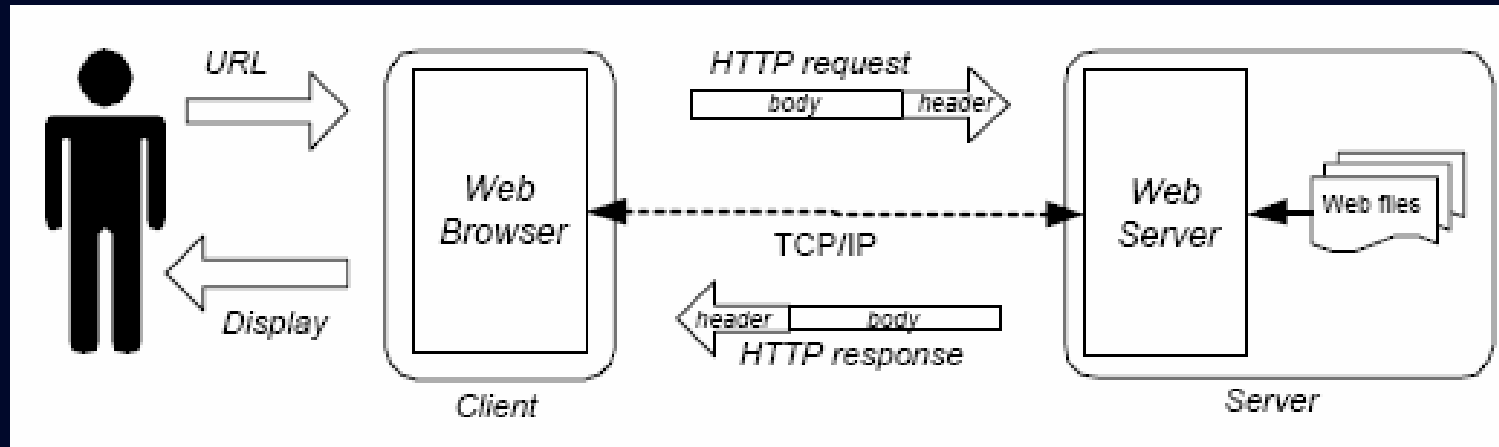
- Kelebihan:

- Dapat diakses kapanpun dan darimanapun selama ada internet
- Dapat diakses hanya dengan menggunakan web browser (umumnya sudah tersedia di PC, PDA, dan handphone terbaru), tidak perlu menginstall aplikasi client khusus

- Kekurangan:

- Antarmuka yang dapat dibuat terbatas sesuai spesifikasi standar untuk membuat dokumen web dan keterbatasan kemampuan web browser untuk menampilkannya
- Terbatasnya kecepatan internet mungkin membuat respon aplikasi menjadi lambat

Arsitektur Web



- Bagaimana Web Bekerja?
- user menetik URL di browser
- browser menghubungi server yang tersebut pada URL
- Setelah terhubung, browser mengirimkan HTTP request
- server menjawab dengan mengirim HTTP response (berisi header dan isi dokumen)
- Untuk dokumen yang terdiri atas beberapa file (misalnya dokumen bergambar), browser harus mengirimkan HTTP request lagi untuk setiap file
- browser menampilkan semua isi dokumen kepada user

Yang perlu dipelajari untuk membuat aplikasi berbasis web

- Disisi client:
 - Sintaks pembuatan dokumen web (HTML & CSS)
 - Client side scripting (JavaScript)
- Disisi server
 - Mekanisme pemanggilan program dan pengambilan output program oleh web server (CGI)
 - Server side scripting (PHP, JSP, ASP, dll. Untuk contoh: PHP)
- Penghubung
 - Sintaks pengalamatan dokumen web (URL)
 - Protokol komunikasi (HTTP)

Client Side Programming

Dalam jaringan komputer, kata “client side” mengacu kepada operasi yang dilakukan oleh client pada suatu hubungan “client-server”. Secara umum, suatu client adalah satu aplikasi komputer, seperti web browser yang berjalan pada satu komputer lokal pengguna atau workstation dan terhubung kesatu server.

Operasi dapat dilakukan client - side karena operasi tersebut membutuhkan akses keinformasi atau fungsi yang tersedia pada client tetapi tidak pada server, karena pengguna membutuhkan observasi terhadap operasi tersebut atau menyediakan input, atau server kekurangan kekuatan pemrosesan untuk melakukan operasi yang tepat waktu untuk seluruh client yang harus dilayaninya.

Sebagai tambahan, jika operasi dapat dilakukan oleh client tanpa mengirim data melalui jaringan, makahal itu memakan waktu lebih sedikit, menggunakan lebih kecil bandwidth dan mengurangi resiko keamanan.

- **Client - side scripting** umumnya mengacu kepada kelas dari program komputer pada web yang dieksekusi client - side, oleh web browser nya pengguna. Tipe pemrograman komputer adalah bagian penting dari konsep Dynamic HTML (DHTML), yang memungkinkan web pages di-script; yakni, agar memiliki perbedaan dan perubahan isi yang bergantung pada masukan pemakai, kondisi lingkungan (seperti waktu dalam hari), atau variabel lain.
- **Pembuat web (web author)** membuat script pada client - side dalam bahasa seperti JavaScript (Client-side JavaScript) atau VBScript, yang didasarkan pada beberapa standar berikut : HTML scripting, HTTP, Document Object Model

- Client-side scripts seringkali disisipkan dalam sebuah dokumen HTML, tetapi ia mungkin juga dimasukkan dalam file terpisah, yang ditunjuk oleh dokumen (atau dokumen-dokumen) yang menggunakannya. Selama request / permintaan, file-file yang diperlukan dikirim ke komputer pemakai oleh web server (atau server) dimana mereka berada. web browser pemakai menjalankan script, kemudian menampilkan dokumen, termasuk semua output yang mungkin dari script tsb. Client-side scripts mungkin juga berisi instruksi – instruksi bagi browser untuk diikuti jika pemakai berinteraksi dengan dokumen dalam suatu cara tertentu, contoh, click button tertentu. Instruksi ini dapat diikuti tanpa komunikasi lebih lanjut dengan server.
- Dengan menampilkan file yang berisi script, pemakai-pemakai mungkin dapat melihat source code-nya. Banyak pembuat web belajar bagaimana membuat sebagian client-side scripts dengan menguji source code untuk pembuat script lain.

Sebaliknya, server-side scripts, ditulis dalam bahasa seperti Perl dan PHP, yang dijalankan oleh web server ketika pemakai meminta / request suatu dokumen. Mereka menghasilkan output dalam bentuk yang dapat dimengerti oleh web browser (biasanya HTML), yang kemudian mengirim kekomputer pemakai. Pemakai tidak dapat melihat source code-nyascript (kecuali pembuat mempublikasikan code secara terpisah), dan bahkan mungkin secara tidak sadar bahwa script telah dieksekusi. Dokumen dihasilkan oleh server-side script, tentu saja, berisi client-side scripts.

Client-side script memiliki akses lebih besar ke informasi dan fungsi yang tersedia pada komputer pengguna, sedangkan server-side script memiliki akses lebih besar ke informasi dan fungsi yang tersedia pada server. Server-side script mengharuskan menginterpretasikan bahasa yang di pasang di server, dan menghasilkan output yang sama tanpa memandang browser client, sistem operasi, atau sistem lainnya yang lebih detail.

Client-side script tidak memerlukan software tambahan pada server (membuat mereka populer dengan Author yang tidak memiliki akses administratif ke server), namun mereka memang mengharuskan web browser pengguna mengerti bahasa scripting dimana script ditulis. Oleh karena itu tidak praktis untuk seorang Author untuk menulis script dalam bahasa yang tidak didukung oleh browser web yang digunakan oleh banyak pengguna mayoritas atau audiens

Server Side Programming

- *Pada jaringan komputer*, istilah server-side mengacu pada operasi - operasi yang dilakukan oleh server dalam hubungan client-server.
- Khususnya, suatu server adalah suatu software program, seperti web server, yang berjalan pada suatu remote server, yang dapat dijangkau dari komputer lokal pemakai atau workstation. Operasi - operasi mungkin dilakukan server-side karena mereka membutuhkan akses ke informasi atau fungsi yang tidak tersedia pada client, atau memerlukan perilaku khusus yang tidak dapat dipercaya ketika itu dilakukan client-side.
- Operasi-operasi Server-side juga mencakup pemrosesan dan penyimpanan data dari suatu client kesuatu server, yang dapat di lihat oleh suatu kelompok / group client.

•Server-side scripting adalah suatu teknologi web server yang mana sebuah permintaan pemakai dipenuhi dengan menjalankan suatuscript secara langsung pada web server untuk menghasilkan dynamic HTML pages. Hal ini biasanya digunakan untuk menyediakan web site yang interaktif yang merupakan interface kedatabase atau simpanan data lain. Hal ini berbeda dengan client-side scripting dimana scripts dijalankan oleh viewing web browser, biasanya dalam JavaScript. Keuntungan utama server-side scripting adalah kemampuan untuk respon sangat tinggi berdasarkan kebutuhan pengguna, hak akses, atau melakukan query ke dalam database.

Dimasa lalu web hal ini hampir eksklusif dilakukan penggunaan suatu kombinasi dari C program, Perl script dan Shell script yang menggunakan Common Gateway Interface (CGI). Script-script tsb dijalankan oleh operating system, dan hasil hanya dilayani kembali oleh web server. Saat ini, Bahasa online scripting lain seperti ASP dan PHP dapat dijalankan secara langsung oleh web server sendiri atau oleh extension modules (contoh mod_perl atau mod_php) ke web server. Baik format scripting (yaitu. CGI atau pelaksanaan yang langsung) dapat digunakan untuk membangun multi-page site yang kompleks, tetapi umumnya mengarahkan pelaksanaan yang mengakibatkan penurunan ongkos eksploitasi dalam kaitan dengan ketiadaan panggilan keinterpreter yang eksternal. Website yang dinamis kadang-kadang juga diperkuat oleh server aplikasi web tertentu, sebagai contoh Python "Base HTTP Server" library, walaupun beberapa tidak mempertimbangkan hal ini untuk menjadi server-side scripting

Daftar teknologi server-side scripting

- Programming language manapun dapat menghasilkan web pages melalui CGI atau suatu modul extension atau application server framework. Bagian ini daftar teknologi yang dirancang terutama atau khusus untuk server-side scripting, biasanya oleh embedding instruksi langsung dalam halaman web template.
- ASP Microsoft dirancang sebagai solusi yang memungkinkan berbagai bahasa (meskipun umumnya VBscriptis digunakan) di dalam halaman HTML seperti luar, terutama digunakan pada Windows tapi dengan platform supporton terbatas lainnya.
- Cold Fusion, Lintas platform tag berbasis server-side scripting sistem yang komersial.

- JSP, Sistem berbasis Java untuk menyisipkan system kedalam code di halaman HTML.
- Lasso, Sebuah sumber data netral diinterpretasikan bahasa pemrograman dan server lintas platform.
- SSIA Sistem Dasar yang merupakan bagian dari web server apache. Bukan merupakan bahasa pemrograman tingkat lanjut, tapi masih berguna untuk hal-hal sederhana seperti menu yang umum.
- PHP Solusi open source berbasis code dalam bahasa umum ke dalam halaman HTML.
- Server-side Java Script, Sebuah bahasa yang umumnya digunakan pada sisi client tetapi juga kadang-kadang pada sisi server.
- SMX Lisplike bahasa open source dirancang untuk dimasukkan ke dalam halaman HTML .

Web Server Software

- Server web adalah sebuah server yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan browser web dan mengirimkan kembali hasilnya dalam bentuk halaman – halaman web yang umumnya berbentuk dokumen HTML. Server web yang terkenal diantaranya adalah Apache dan Microsoft Internet Information Service(IIS). Apache merupakan server web multi - platform, sedangkan IIS hanya dapat beroperasi di sistem operasi Windows.
- Server web juga dapat berarti komputer yang berfungsi seperti definisi diatas.

- XAMPP adalah suatu paket software free berisi Apache HTTP Server, MySQL database dan tool yang perlu untuk menggunakan PHP dan Perl programming languages. Program di - release dibawah naungan GNU General Public License dan layanan bersifat bebas, mudah digunakan web server, mampu memberi halaman – halaman yang dinamis. Saatini, XAMPP sudah tersedia untuk Windows, Linux, Sun Solaris dan Mac OS X(X dalam namanya dapat bermakna dapat digunakan untuk salah satu dari operating systems ini).
- Secara resmi, XAMPP adalah hanya diharapkan untuk penggunaan sebagai alat pengembangan, mengizinkan para perancang dan para programmer website untuk menguji pekerjaan mereka pada komputer mereka sendiri tanpa akses ke Internet. Dalam Praktek, bagaimanapun, XAMPP adalah kadang – kadang digunakan untuk benar-benar melayani halaman web pada World Wide Web, dan dengan beberapa modifikasi umumnya disepakati cukup menjamin keamanan.

HTTP

- HTTP (Hyper Text Transfer Protocol) adalah yang dipergunakan untuk mentransfer dokumen dalam World Wide Web (WWW). Protokol ini adalah protokol ringan, tidak berstatus dan generik yang dapat dipergunakan berbagai macam tipe dokumen.
- Pengembangan HTTP dikoordinasi oleh Konsorsium World Wide Web (W3C) dan kelompok kerja Internet Engineering Task Force (IETF), bekerja dalam publikasi satu seri RFC, yang paling terkenal RFC 2616, yang menjelaskan HTTP/1,1, versi HTTP yang digunakan umum sekarang ini.

- HTTP adalah sebuah protokol meminta / menjawab antara client dan server. Sebuah client HTTP seperti web browser, biasanya memulai permintaan dengan membuat hubungan TCP/IP ke port tertentu di tuan rumah yang jauh (biasanya port 80). Sebuah server HTTP yang mendengarkan di port tersebut menunggu client mengirim kode permintaan (request), seperti "GET/ HTTP/1.1" (yang akan meminta halaman yang sudah ditentukan), diikuti dengan pesan MIME yang memiliki beberapa informasi kode header yang menjelaskan aspek dari permintaan tersebut, diikuti dengan body dari data tertentu. Beberapa kepala (header) juga bebas ditulis atau tidak, sementara lainnya (seperti tuan rumah) diperlukan oleh protokol HTTP/1,1. Begitu menerima kode permintaan (dan pesan, bila ada), server mengirim kembali kode jawaban, seperti "200 OK", dan sebuah pesan yang diminta, atau sebuah pesan error atau pesan lainnya.

Sejarah

- Protokol HTTP pertama kali dipergunakan dalam WWW pada tahun 1990. Pada saat tersebut yang dipakai adalah protokol HTTP versi 0.9. Versi 0.9 ini adalah protokol transfer dokumen secara mentah, maksudnya adalah data dokumen dikirim sesuai dengan isi dari dokumen tersebut tanpa memandang tipe dari dokumen.
- Kemudian pada tahun 1996 protokol HTTP diperbaiki menjadi HTTP versi 1.0. Perubahan ini untuk mengakomodasi tipe – tipe dokumen yang hendak dikirim beserta encoding yang dipergunakan dalam pengiriman data dokumen.
- Sesuai dengan perkembangan infrastruktur internet maka pada tahun 1999 dikeluarkan HTTP versi 1.1 untuk mengakomodasi *proxy*, *cache* dan koneksi yang persisten.

- Contoh Transaksi
- S = Server C = Client
- C: (Inisialisasi koneksi)
- C: GET /index.htmHTTP/1.1
- C: Host: www.wikipedia.org
- S: 200 OK
- S: Mime-type: text/html
- S:
- S: --data dokumen --
- S: (close connection)

Mekanisme CGI (Common Gateway Interface)

- Jika dokumen web yang diminta oleh web browser merupakan file program atau file HTML yang disisipi program, maka web server akan menjalankan (*run*) file tersebut dengan bantuan interpreter atau sistem operasi
- Informasi yang diberikan oleh web server kepada program:
 - ✓ **Server variables**: berbagai informasi yang ada pada *HTTP request* yang sedang diterima, nama dan path file program, kapabilitas web server, dll
 - ✓ **Environment variables**: informasi yang telah diset (mungkin oleh aplikasi lain) pada lingkungan sistem operasi
 - ✓ **Cookie**: isi cookie yang disimpan diweb browser
 - ✓ **Request parameter**: input dari user yang dikirimkan bersamaan dengan *HTTP request* oleh web browser
- Hasil output program ke *standard output (screen)* akan diambil oleh web server untuk dikirimkan kepada web browser sebagai *HTTP*

response

CGI pada PHP

- Pada PHP:

- Server variables diletakkan pada predefined variable `$_SERVER`

- `$_SERVER["PHP_SELF"]` = path dan nama file yang sedang dieksekusi

- `$_SERVER["SERVER_NAME"]` = nama host / server

- `$_SERVER["REQUEST_METHOD"]` = jenis metode request

- `$_SERVER["HTTP_USER_AGENT"]` = identitas web browser yang melakukan request

- `$_SERVER["REMOTE_ADDR"]` = nomor IP user

- dll

- Environment variables diletakkan pada predefined variable `$_ENV`

- Cookie diletakkan pada predefined variable `$_COOKIE`

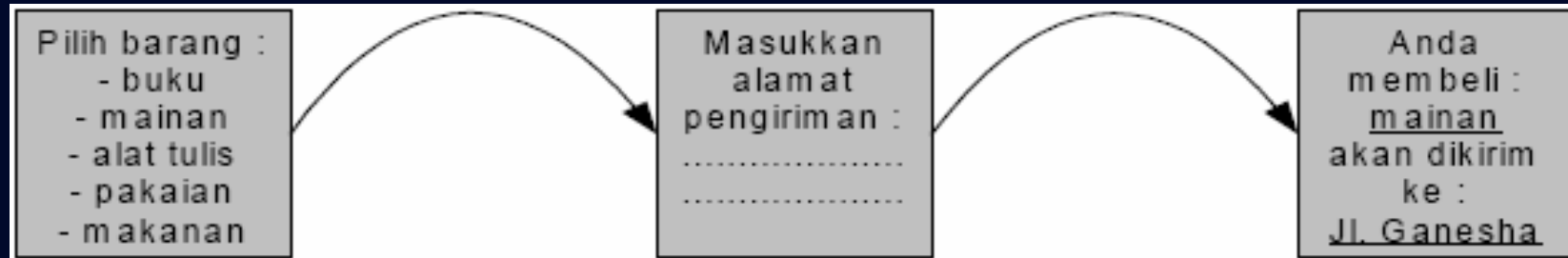
- Request parameter diletakkan pada predefined variable `$_GET`, `$_POST`, `$_FILES`

- Output menggunakan perintah `echo`

Penanganan *State*

Konsep “*task*” dalam aplikasi berbasis web

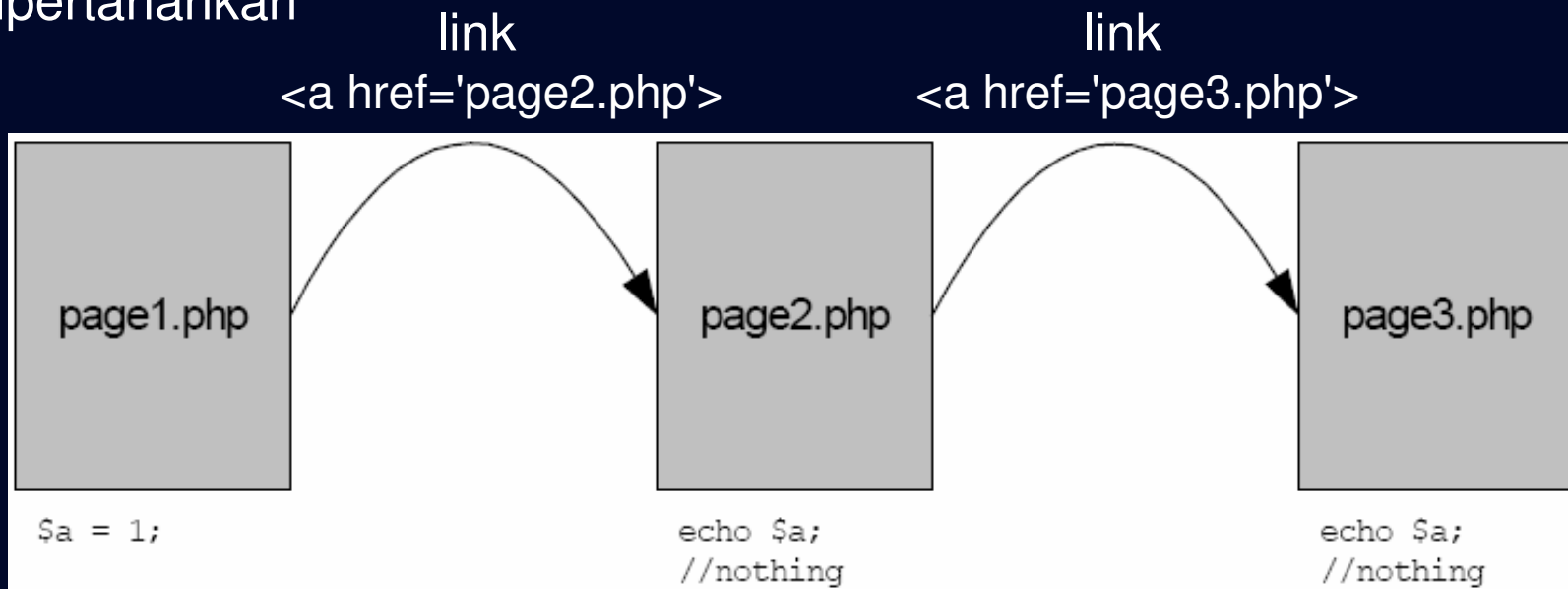
- Aplikasi mempunyai sejumlah fitur / fungsi yang memungkinkan user untuk melakukan sejumlah *task*
- Untuk memenuhi suatu *task*, dapat digunakan satu atau lebih halaman web sebagai antarmuka
- Contoh *task* yang menggunakan satu halaman web:
 - Membaca artikel nomor tertentu
 - Menambahkan entry baru pada guest book
- Contoh *task* yang menggunakan lebih dari satu halaman web:
 - Membeli barang secara online
 - halaman 1: menampilkan daftar barang yang ada untuk dipilih
 - halaman 2: mengisi alamat pengiriman
 - halaman 3: menampilkan informasi transaksi yang baru saja dilakukan



- Antara halaman web yang satu dengan yang lain harus ada komunikasi mengenai *state* dari *task* (misalnya, halaman 3 butuh informasi mengenai barang yang telah dipilih pada halaman 1)

Stateless HTTP

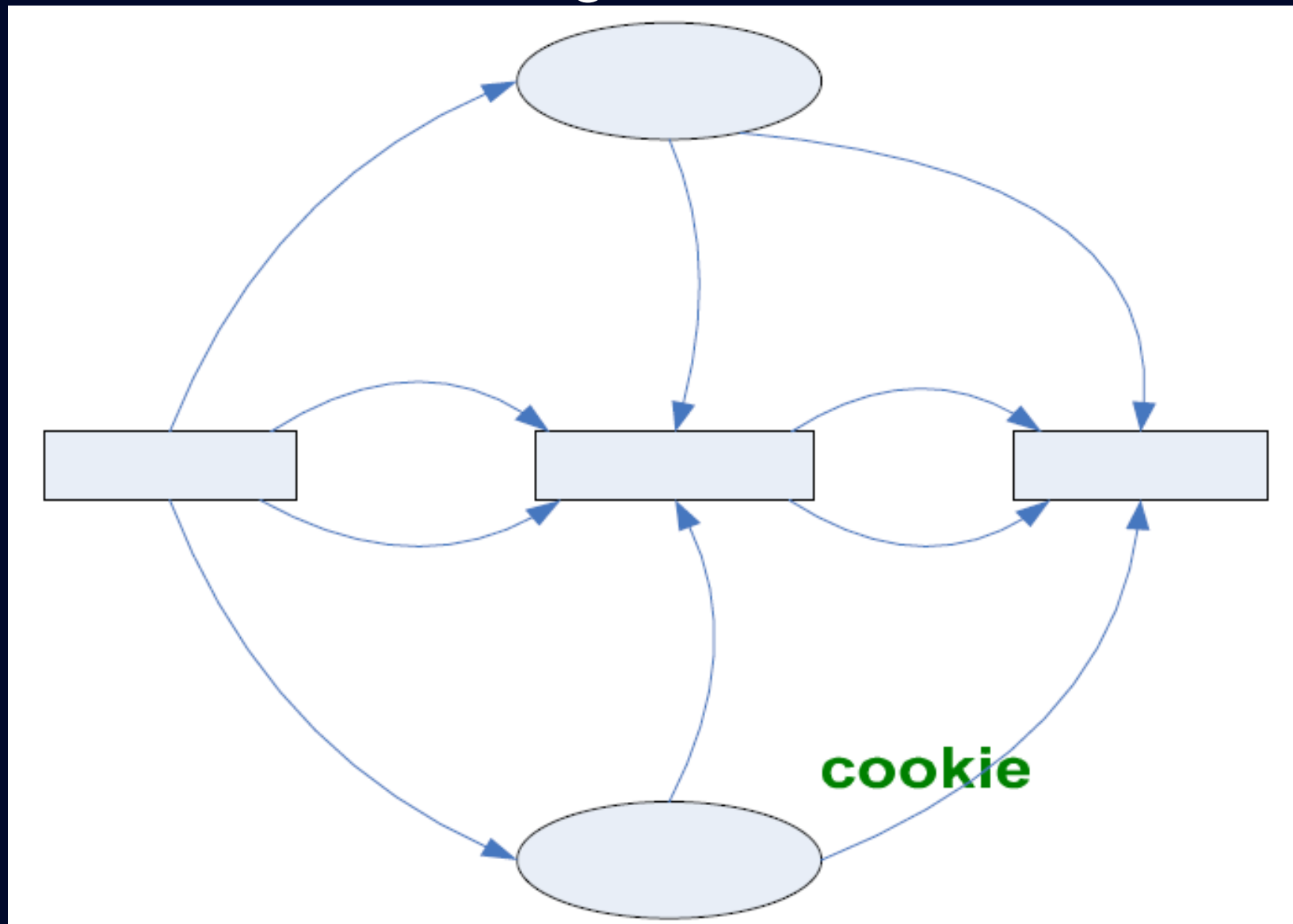
- Permasalahan : HTTP bersifat *stateless*: satu request saling independen terhadap request lainnya, tidak ada *state* (variabel) yang dipertahankan



Cara untuk mengatasi HTTP yang stateless:

- Message passing via URL / Form
- Cookie
- Session

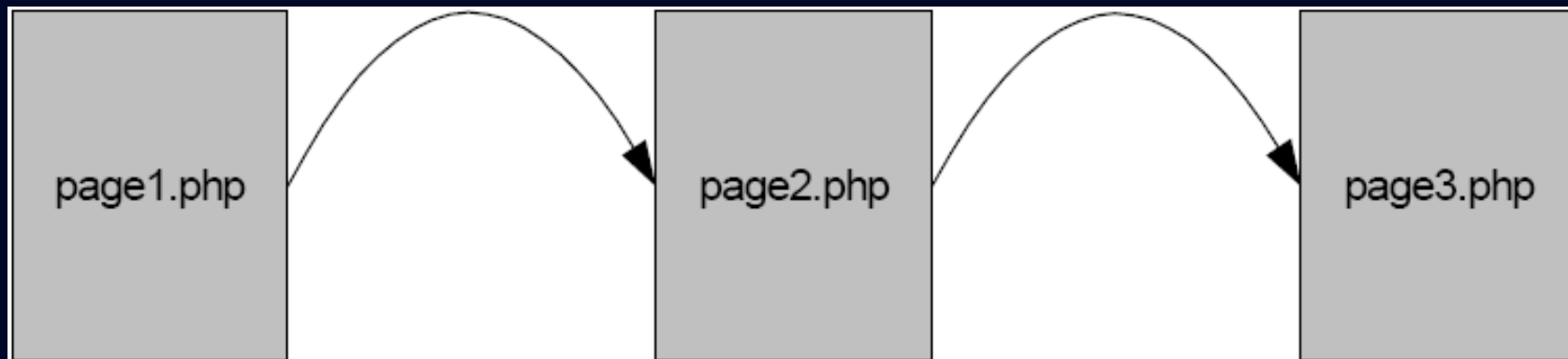
Penanganan *State*



Message Passing via URL

- Nilai variabel di lewatkan melalui parameter URL
- KIRIM : mengkonstruksi parameter pada URL
- TERIMA: akses variabel \$_GET

link
link
``
``



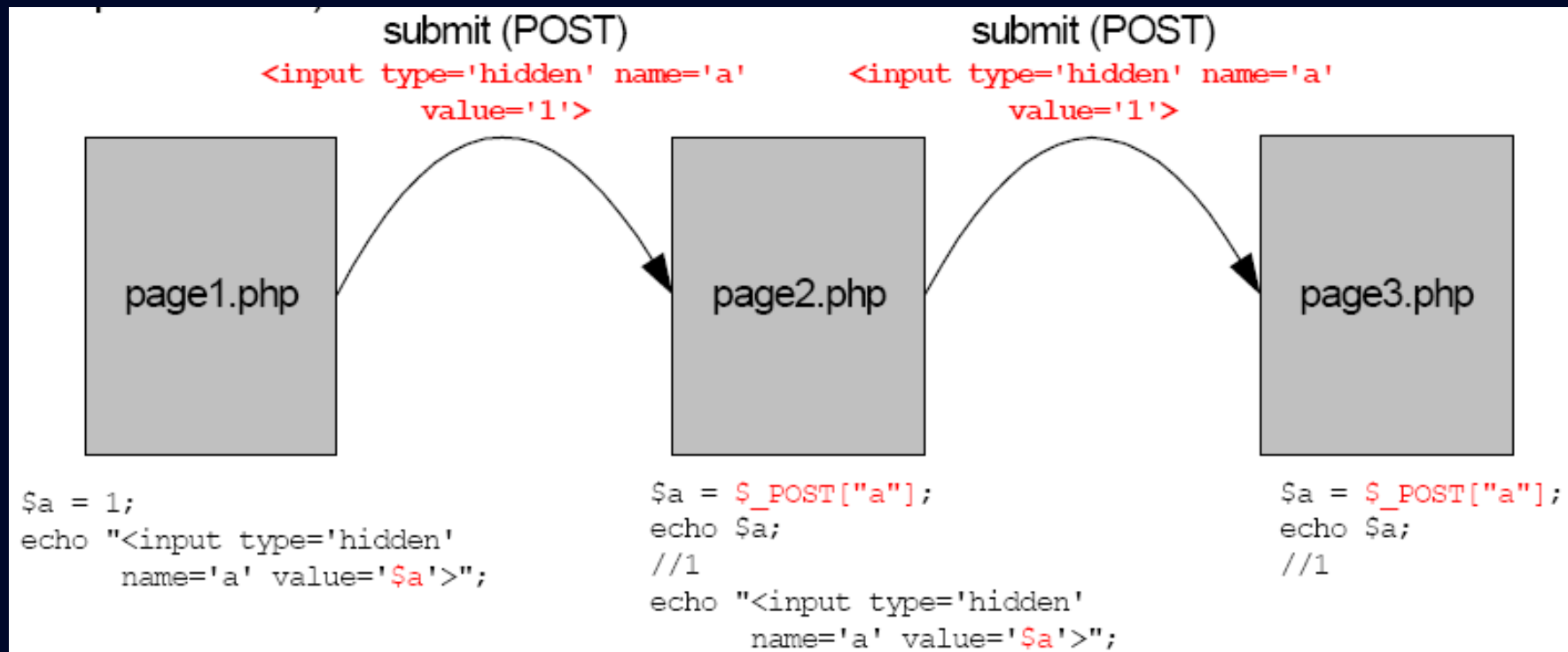
```
$a = 1;
echo "<a href='page2.php?a=$a'>";
```

```
$a = $_GET["a"];
echo $a;
//1
echo "<a
href='page3.php?a=$a'>";
```

```
$a = $_GET["a"];
echo $a;
//1$
```

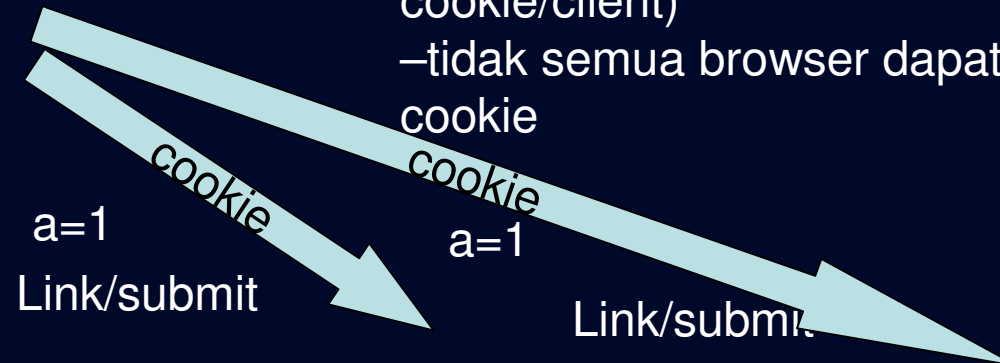
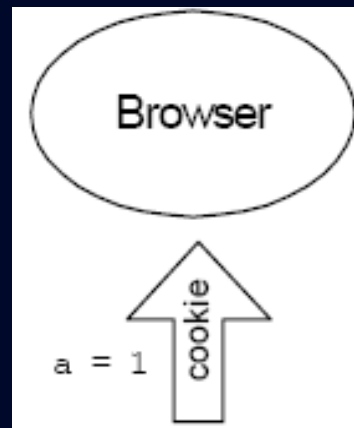
Message Passing via Form

- Nilai variabel dilewatkan melalui input tipe hidden pada form HTML
- KIRIM: mengkonstruksi input tipe hidden dalam form
- TERIMA: akses variabel `$_GET` atau `$_POST` (tergantung method pada form)



Cookie

- Nilai variabel disimpan disuatu tempat penyimpanan (file) di browser
 - KIRIM : `setcookie ("name", "value");`
 - TERIMA : akses variabel `$_COOKIE`
 - Keterbatasan:
 - cookie hanya dapat menyimpan sejumlah kecil data (4 kb/cookie, 20 cookie/domain, 300 cookie/client)
 - tidak semua browser dapat/mau menerima cookie



Session

- Nilai variabel disimpan disuatu tempat penyimpanan (file) diserver
- KIRIM dan TERIMA: akses variabel `$_SESSION`
 - Fungsi untuk session handling : `session_start()`, `session_id()`, `session_destroy()`, dll.
 - Keterbatasan:
 - Session bergantung pada mekanisme state handling lain untuk menyimpan ID session

