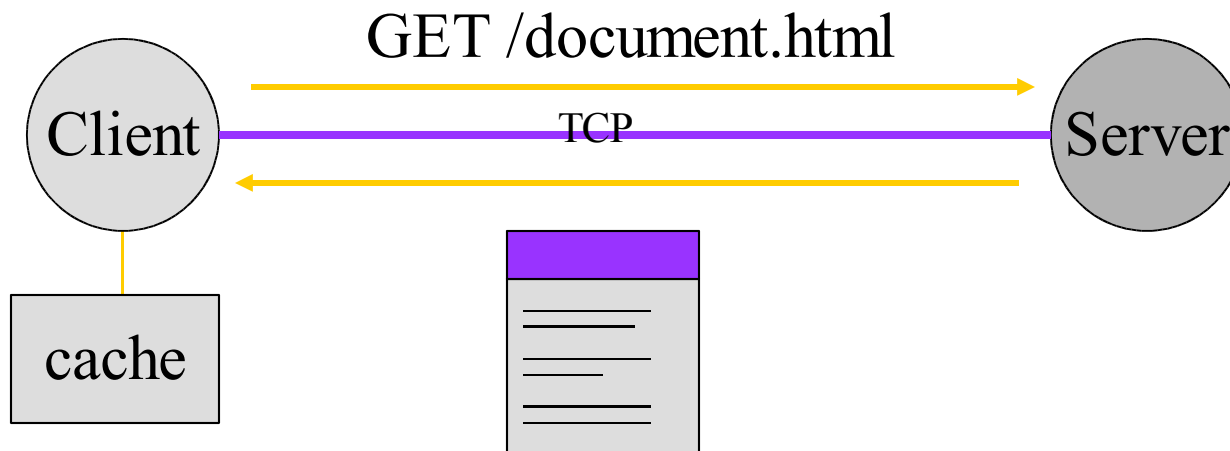


Mengenal HTTP

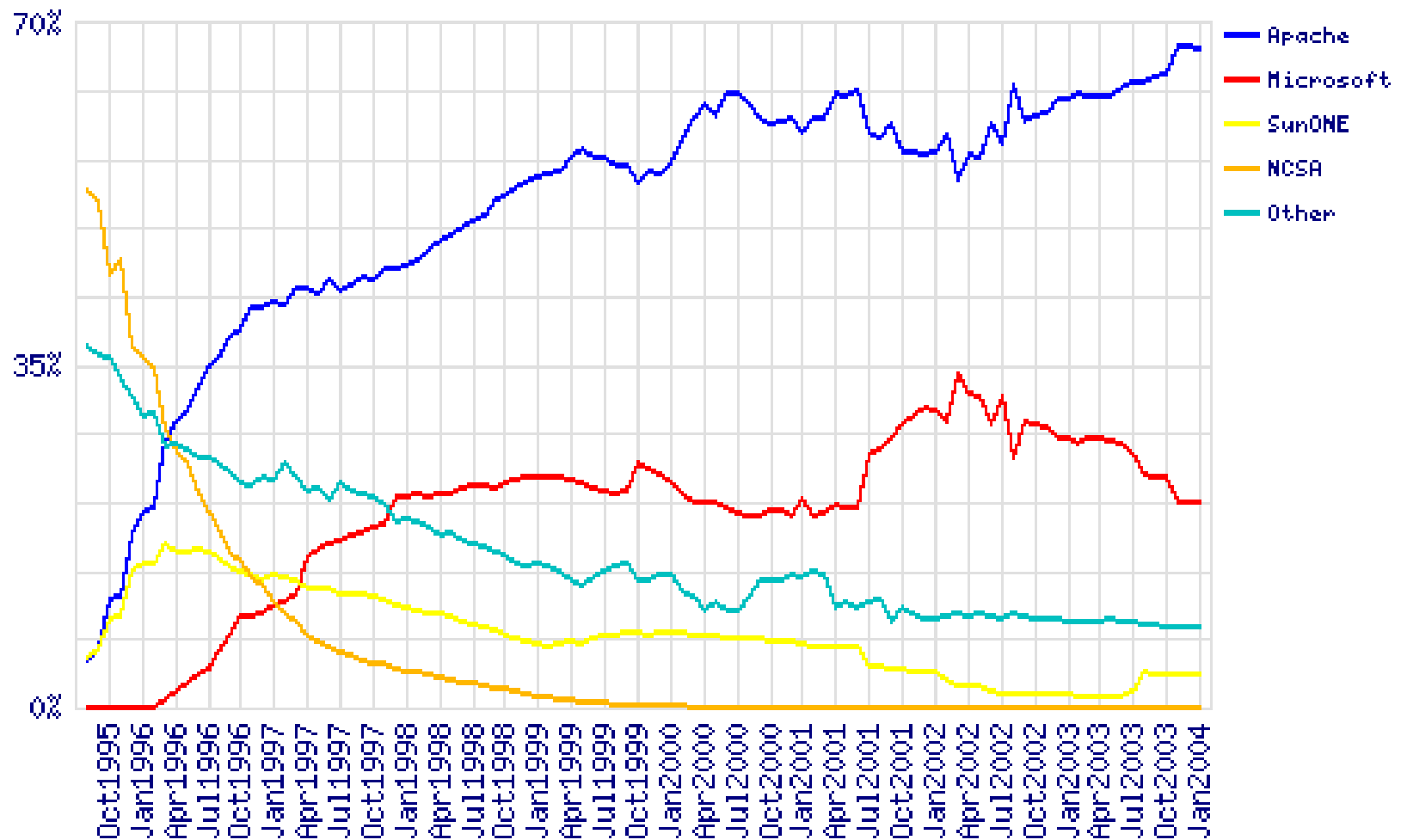
Budi Susanto

budsus@ukdw.ac.id, budsus@yahoo.com

The Web



Market Share dari 46,067,743 web site



Dasar HTTP

Definisi HTTP

- Pada RFC 2616 didefinisikan :

“The Hypertext Transfer Protocol (HTTP) is an application-level protocol for distributed, collaborative, hypermedia information systems.”
- HTTP dimulai digunakan oleh WWW pada tahun 1990 dengan versi HTTP/0.9
- Dikembangkan menjadi HTTP/1.0 dengan RFC 1945, yang memungkinkan dokumen diformat MIME
- HTTP juga merupakan protokol yang digunakan antara user agent dengan proxy/gateway ke sistem internet lain, misal SMTP, FTP, NNTP, FTP, Gopher dan WAIS.

Struktur Request

<METHOD> <URL> <HTTPVERSION>\r\n

<HEADERNAME>: <HEADERVAL>\r\n

<HEADERNAME>: <HEADERVAL>\r\n

...

\r\n

<DATA, IF POST>

```
GET /test/index.html?foo=bar+baz&name=steve HTTP/1.0\r\n
```

```
Connection: Keep-Alive\r\n
```

```
User-Agent: Mozilla/4.07 [en] (X11; I; Linux 2.0.36 i686)\r\n
```

```
Host: ninja.cs.berkeley.edu:5556\r\n
```

```
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*\r\n
```

```
Accept-Encoding: gzip\r\n
```

```
Accept-Language: en\r\n
```

```
Accept-Charset: iso-8859-1,*,utf-8\r\n
```

```
\r\n
```

Struktur Response

<HTTPVERSION> <STATUS CODE> <MSG>\r\n

<HEADERNAME>: <HEADERVERVAL>\r\n

<HEADERNAME>: <HEADERVERVAL>\r\n

...

\r\n

<DATA, IF NECESSARY>

HTTP/1.0 200 OK

Server: Netscape-Enterprise/2.01

Date: Thu, 04 Feb 1999 00:28:19 GMT

Accept-ranges: bytes

Last-modified: Wed, 01 Jul 1998 17:07:38 GMT

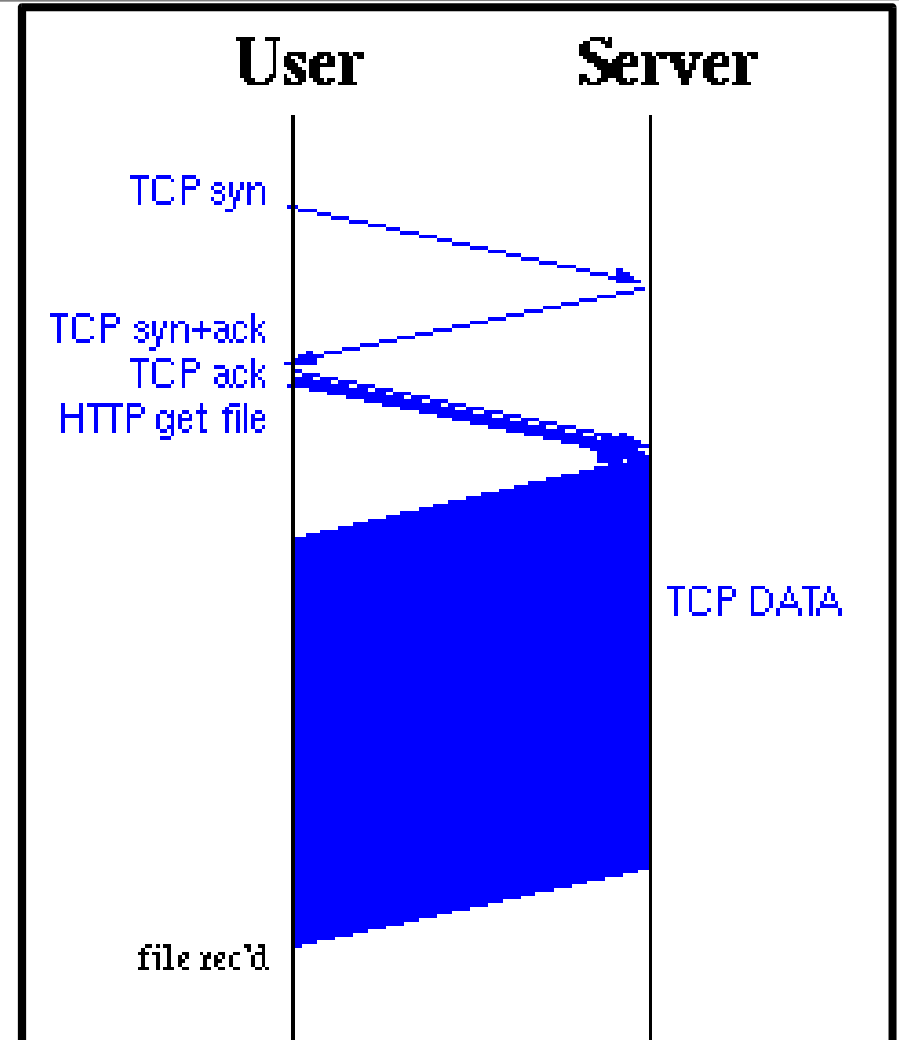
Content-length: 1848

Content-type: text/html

TCP level

Permasalahan:

- TCP threeway handshake
- Lakukan request, data dikirim
- Koneksi baru untuk tiap inline image



Interesting TCP gotchas

Solusi :

- Persistent HTTP
 - Pada HTTP/1.0, gunakan “**Connection: Keep-Alive\r\n**” pada header
 - Pada HTTP/1.1, sudah ada P-HTTP
- Memungkinkan membawa beberapa perintah dalam satu koneksi

Metode HTTP/1.0 Client

- GET
 - Mengambil dan menerima sebuah dokumen
 - URL digunakan untuk mengirimkan data
 - GET /foo/bar.html?x=bar&bam=baz
- POST
 - Mengirim sebuah form, dan menerima response
- HEAD
 - Seperti GET, namun hanya menerima header HTTP saja.
Digunakan untuk caching
- PUT, DELETE, LINK, UNLINK
 - Jarang sekali digunakan

Kode Status HTTP/1.0

- Ada 5 tipe kode status HTTP/1.0
 - 1xx: informational
 - 2xx: successful, e.g. 200 OK
 - 3xx: redirection
 - 301 Moved Permanently
 - 304 Not Modified
 - 4xx: Client Error
 - 400 Bad Request
 - 401 Unauthorized
 - 403 Forbidden
 - 404 Not Found
 - 5xx: Server Error
 - 501 Not Implemented
 - 503 Service Unavailable

Header HTTP/1.0 (case insensitive?)

- **Allow** – dihasilkan dari server
 - **Allow: GET, HEAD**
 - Tidak pernah digunakan
- **Authorization** – dikirim oleh client
 - **Authorization: <credentials>**
 - “Basic Auth” secara umum digunakan
 - **<credentials> = Base64(username:password)**
- **Content-Encoding** – dikirim oleh keduanya
 - **Content-Encoding: x-gzip**
 - Memilih encoding untuk transport, tidak untuk isi

Header HTTP/1.0 (lanjutan)

- **Content-Length** – dikirim keduanya
 - **Content-Length: 56**
 - Berapa besar yang dikirim?
 - Dibutuhkan untuk persistent HTTP, atau POST
- **Content-Type** – dikirim oleh server
 - **Content-Type: text/html**
 - Tipe MIME dari dokumen
 - Jika lebih dari satu : **multipart/mixed**
- **Date**
 - **Date: Tue, 15 Nov 1994 08:12:31 GMT**
 - 3 format tanggal(RFC 822, RFC 850, asctime())

Header HTTP/1.0 (lanjutan)

- **Expires** – dikirim oleh server
 - **Expires: Thu, 01 Dec 1994 16:00:00 GMT**
 - Merupakan mekanisme masa hidup cache
 - Tidak dapat memaksa client untuk refresh tampilan saat ini, hanya ketika dilakukan refresh saja
- **From** – dikirim oleh client
 - **From: gribble@cs.berkeley.edu**
 - Tidak selalu digunakan
- **If-Modified-Since** – dikirim oleh client
 - **If-Modified-Since: Sat, 29 Oct 1994 19:43:31 GMT**
 - Server mengembalikan data jika termodifikasi, selain itu “304 Not Modified”

Header HTTP/1.0 (lanjutan)

- **Last-Modified** – dikembalikan oleh server
 - **Last-Modified: Sat, 29 Oct 1994 19:43:31 GMT**
 - Secara semantik tidak tepat – modifikasi file? *Record timestamp?* Tanggal file yang dihasilkan secara dinamis?
 - Digunakan dengan **If-Modified-Since** dan **HEAD**
- **Location** – dikembalikan oleh server
 - **Location: <http://www.cs.ubc.ca>**
 - Digunakan ketika redirection (**3xx**)
- **Pragma** – dikirim oleh client atau server
 - **Pragma: no-cache**

Header HTTP/1.0 (lanjutan)

- **Referer** – dikirim oleh client
 - **Referer: http://www.xxx-smut.com**
 - Alamat dimana request dihasilkan
- **Server** – dihasilkan oleh server
 - **Server: Netscape-Enterprise/2.01**
 - Identifikasi server.
- **User-Agent** – dikirim oleh client
 - **User-Agent: Mozilla/4.07 [en] (X11; I; Linux 2.0.36 i686)**
 - Identifikasi software client

HTTP/1.0 Server headers

- **WWW-Authenticate** - sent by server
 - **WWW-Authenticate:** <challenge>
 - Memberitahukan ke client untuk mengirim ulang request dengan header **Authorization:**.
- Header ini dapat bertambah
 - **Accept:** image/gif, image/jpeg, text/*, */*
 - **Accept-Encoding:** gzip
 - **Accept-Language:** en
 - **Retry-After:** (date) or (seconds)
 - **[Set-]Cookie:**
Part_Number="Rocket_Launcher_0001";
Version="1"; Path="/acme"
 - **Title:** (title)

Contoh HTTP Request

GET /SE452/index.html HTTP/1.0

Connection: Keep-Alive

User-Agent: Mozilla/4.72 [en]

Host: venus.cs.depaul.edu

Accept: image/gif, image/jpeg, */*

Accept-Encoding: gzip

Accept-Language: en

Accept-Charset: iso-8859-1,*,utf-8

Contoh HTTP Reply

HTTP/1.1 200 OK

Accept-Ranges: bytes

Content-Length: 110

Connection: close

Content-Type: text/html

<html>

<head>

<title>Hello World!</title>

</head>

<body bgcolor="white">

<h1>Hello World!</h1>

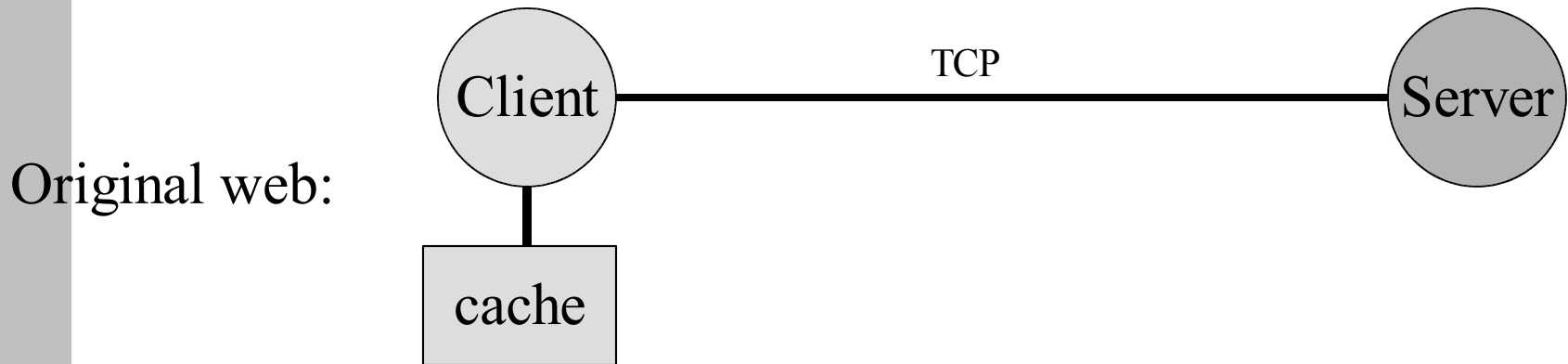
</body>

</html>

Penambahan pada HTTP/1.1

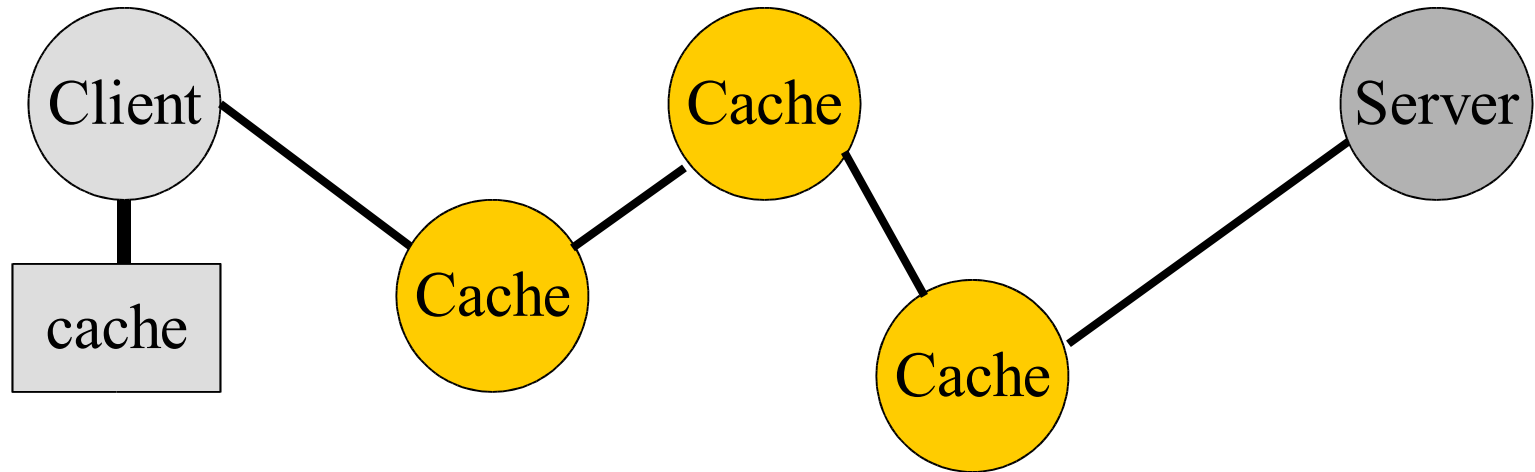
- Beberapa kelemahan pada HTTP/1.0
 - HTTP/1.0 belum mampu digunakan pada konfigurasi hierarchical proxies, caching, kebutuhan persistent connections, atau virtual hosts.
- HTTP/1.1 dikembangkan untuk mengatasi masalah tersebut
 - Merupakan protokol yang lebih komplek
 - Merupakan protokol yang sulit diimplementasikan
- Kunjungi :
<http://www.w3.org/Protocols/rfc2616/rfc2616.html>

Caches



- Server melayani permintaan dokumen yang sama berkali-kali untuk dicache di lokal

Solusi: Cache Hierarchy



- NLANR (National Laboratory for Applied Network Research) cache hierarchy dikembangkan
 - Secara informal menggunakan Squid cache
 - root server melakukan caching sampai 30GB per hari
 - Setiap dapat meminta...
- Baca artikel di <http://www.nlanr.net/Papers/Cache96/>

Beberapa hal baru HTTP/1.1

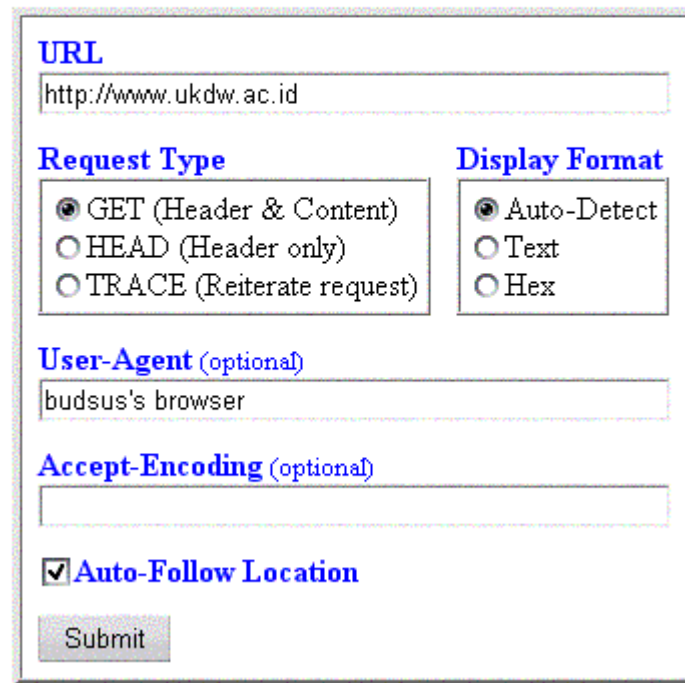
- **Host:** www.ninja.com
 - clients HARUS mengirim ini – perbaikan untuk multihome
 - Untuk client 1.0 dan 1.1 sudah siap
- **Range:** bytes=300-304,601-993
 - Berguna untuk perbaikan koneksi yang putus *broken connection*
- **Age:** <seconds, date>
 - expiration dari caches
- **Etag:** fa898a3e3
 - unique tag untuk mengidentifikasi dokumen
- **Cache-control:** <command>
 - Menandai dokumen bersifat private (jangan disimpan pada cache)
- “chunked” transfer encoding
 - Segmentasi dokumen – tidak perlu menghitung panjang dokumen keseluruhan. Berguna untuk response yang dinamis.

Contoh tool untuk pengecekan HTTP

- <http://www.rexswain.com/httpview.html>

Rex Swain's HTTP Viewer

See *exactly* what an HTTP request returns to your browser



The image shows a web form for 'Rex Swain's HTTP Viewer'. It has a title 'URL' in blue. Below it is a text input field containing 'http://www.ukdw.ac.id'. There are two columns of radio buttons. The first column is titled 'Request Type' in blue and contains three options: 'GET (Header & Content)' (selected), 'HEAD (Header only)', and 'TRACE (Reiterate request)'. The second column is titled 'Display Format' in blue and contains three options: 'Auto-Detect' (selected), 'Text', and 'Hex'. Below these is a text input field for 'User-Agent (optional)' containing 'budsus's browser'. There is another empty text input field for 'Accept-Encoding (optional)'. At the bottom left is a checked checkbox for 'Auto-Follow Location'. At the bottom center is a 'Submit' button.

URL
http://www.ukdw.ac.id

Request Type

- ☒ GET (Header & Content)
- ☐ HEAD (Header only)
- ☐ TRACE (Reiterate request)

Display Format

- ☒ Auto-Detect
- ☐ Text
- ☐ Hex

User-Agent (optional)
budsus's browser

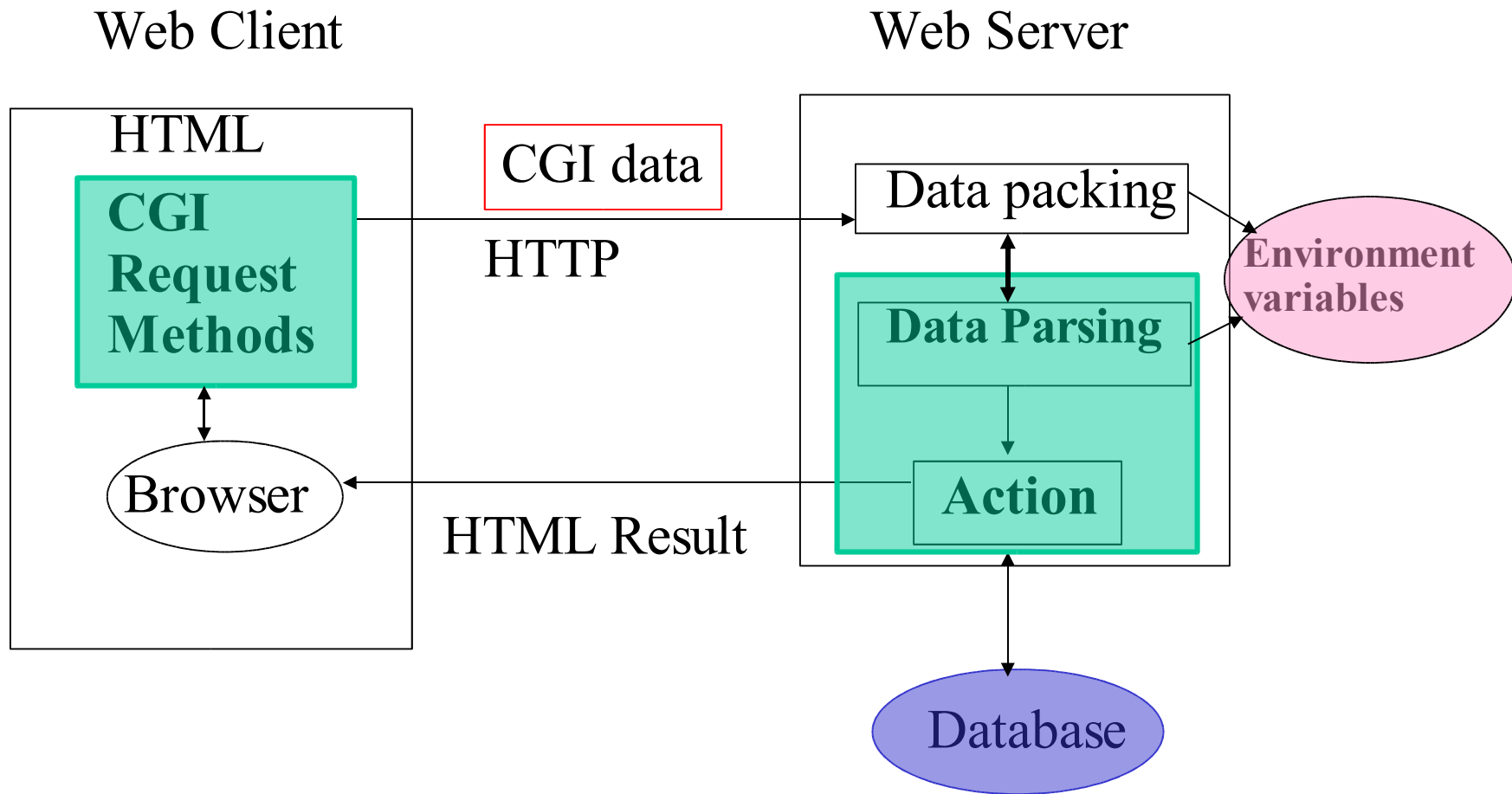
Accept-Encoding (optional)

☒ **Auto-Follow Location**

Submit

Common Gateway Interface (CGI)

Pemrosesan CGI



CGI Environment

AUTH_TYPE	Jika server mendukung authentication user; dan script diprotek; variabel ini digunakan untuk validasi user.
CONTENT_LENGTH	Panjang dari input data yang diberikan oleh client
CONTENT_TYPE	Memberikan nilai tipe MIME dari data yang dikirim ke aplikasi CGI jika menggunakan metode POST
GATEWAY_INTERFACE	Interface gateway yang digunakan. Format : CGI/revision. Contoh : CGI/1.1.
PATH_INFO	Informasi path tambahan yang diberikan client.
PATH_TRANSLATED	Server menyediakan suatu penerjemah versi dari PATH_INFO; yang akan mengambil path dan melakukan pemetaan dari virtual-to-physical terhadap path tersebut.

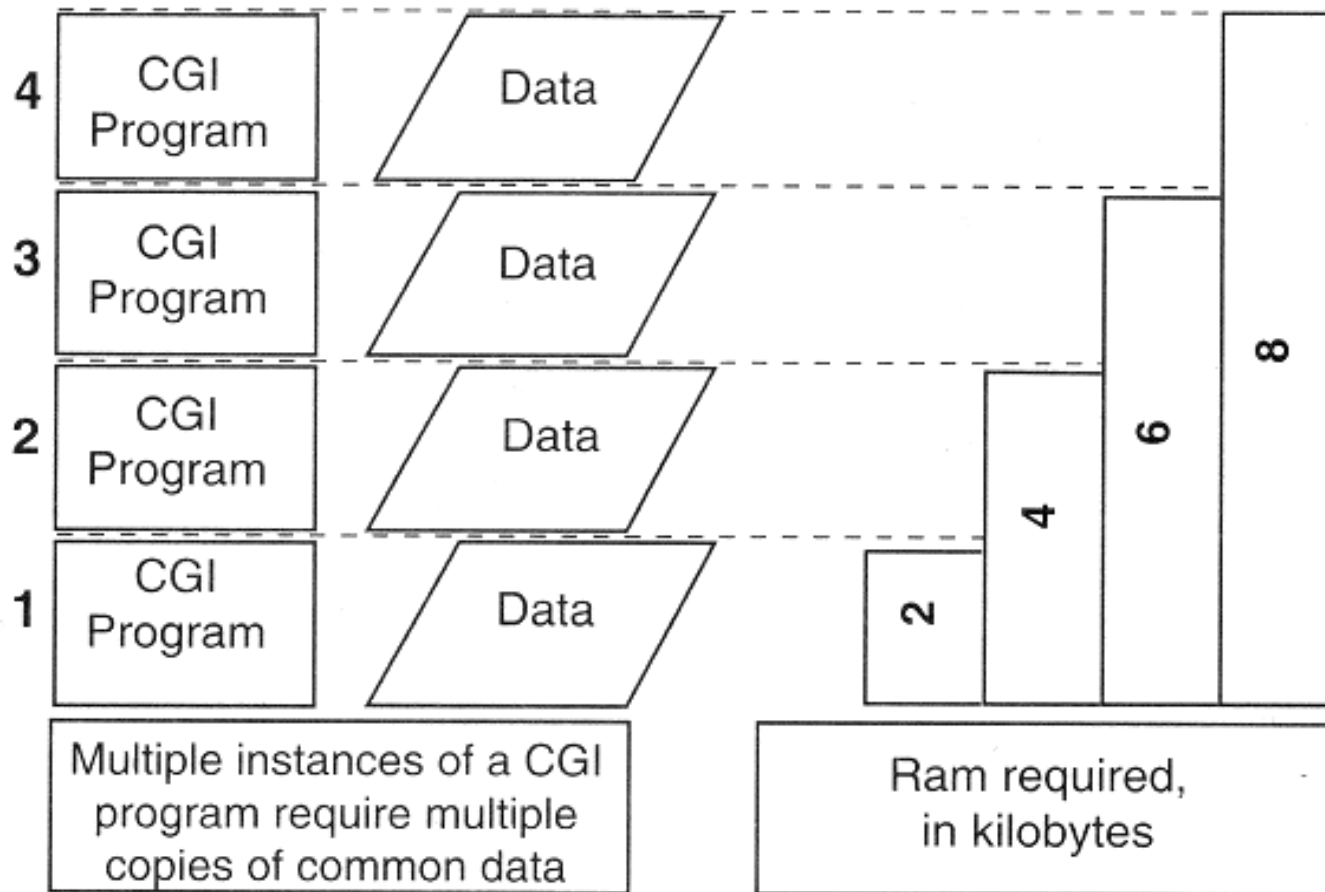
QUERY_STRING	Berisi informasi setelah tanda ? pada URL. Input string ini adalah URL-encoded (spasi diganti + dan beberapa karakter diganti karakterescape).
REMOTE_ADDR	Alamat IP dari remote host yang membuat request
REMOTE_HOST	Hostname yang membuat request.
REMOTE_IDENT	Berisi user name yang diberikan oleh ident protocol; yang menerapkan RFC (Request For Comment) 931.
REMOTE_USER	Berisi user name yang ter-authentication.
REQUEST_METHOD	Berisi metode yang digunakan
SCRIPT_NAME	Berisi nama script yang panggil; contoh /cgi-bin/test.cgi
SERVER_NAME	Domain name yang menjalankan web server; contoh www2.ukdw.ac.id
SERVER_PORT	Port number dimana request dikirimkan.
SERVER_PROTOCOL	Berisi informasi nama dan revisi protokol request yang dikirim.
SERVER_SOFTWARE	Nama dan versi dari software server. Format : name/version

Contoh script CGI/Perl

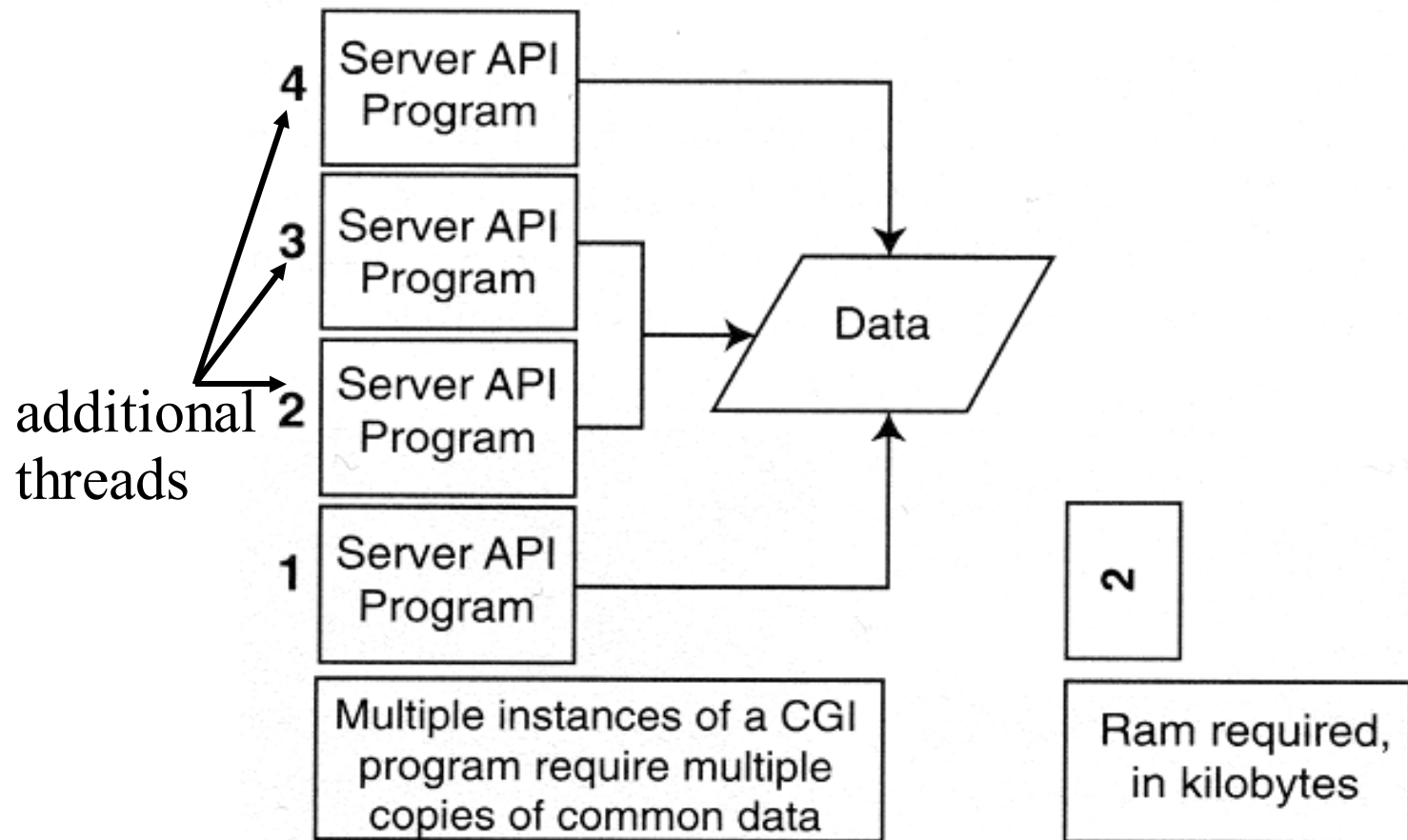
```
#!/usr/local/bin/perl
```

```
print "Content-type: text/html\n\n";  
print "<tt>\n";  
foreach $key (sort keys(%ENV)) {  
    print "$key = $ENV{$key}<p>";  
}
```

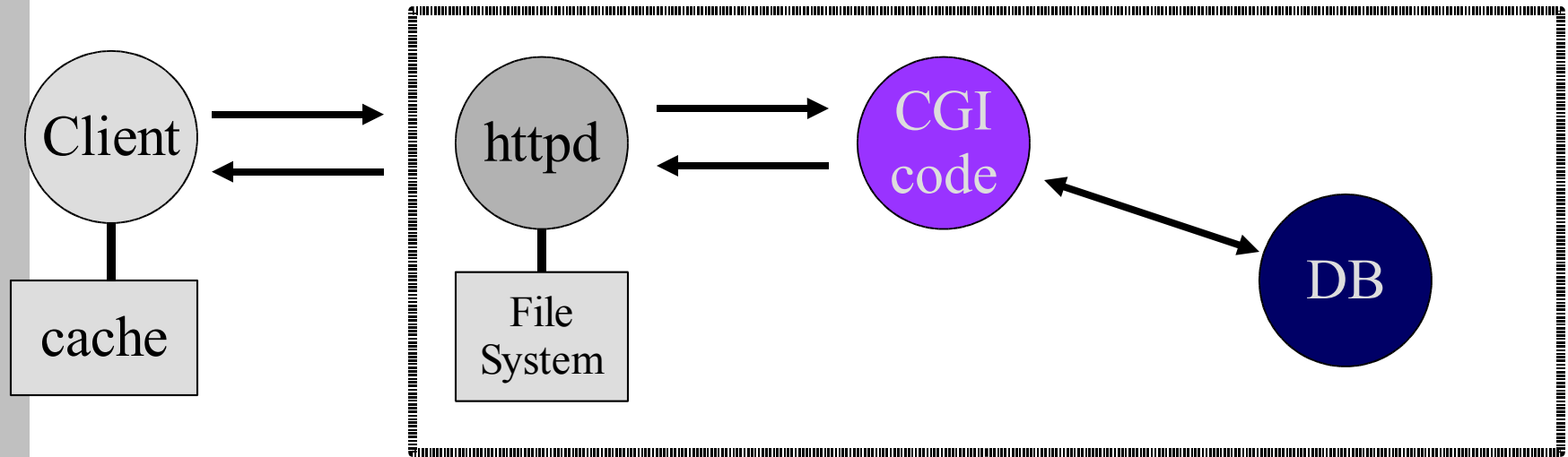
n CGI program butuh $N * 2K$ RAM



HTTP Server API hanya butuh 2K



CGI-BIN to DB gateways



- **JDBC/ODBC gateways**
 - single-node DB, biasanya berjalan pada host terpisah
 - blocking operations
 - Masalah transaksi 'nakal' – bagaimana client tahu apakah operasi dari CGI berhasil atau tdak?

cgi-bin security

- Banyak hal kelemahan dengan program CGI-BIN
 - buffer overflows
 - shell metacharacter expansion
 - Apa yang terjadi jika Anda memberikan
``cat /etc/passwd``
pada field Form?
 - Pengiriman mail, pembacaan files
 - redirection – memungkinkan melewati alamat IP tertentu