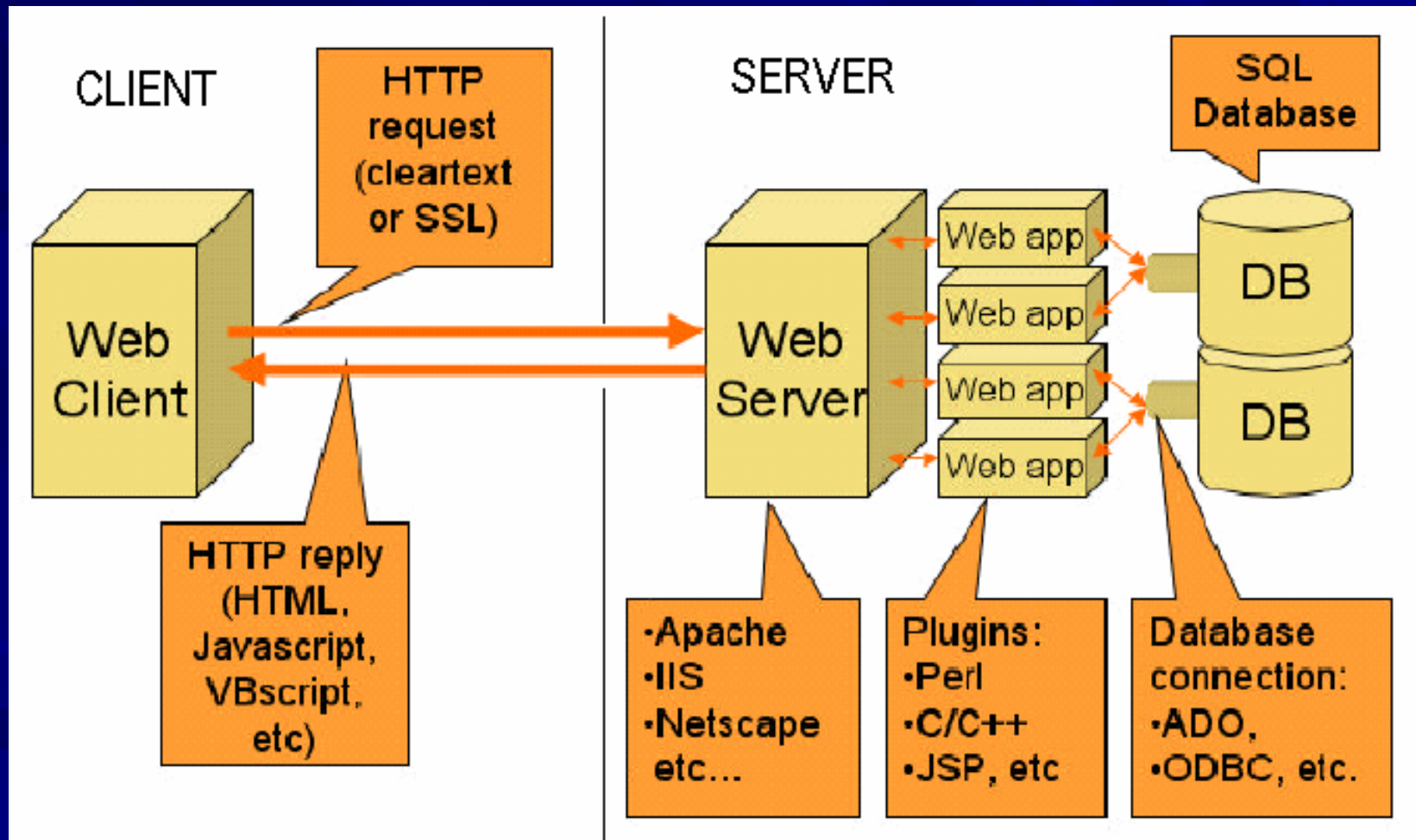
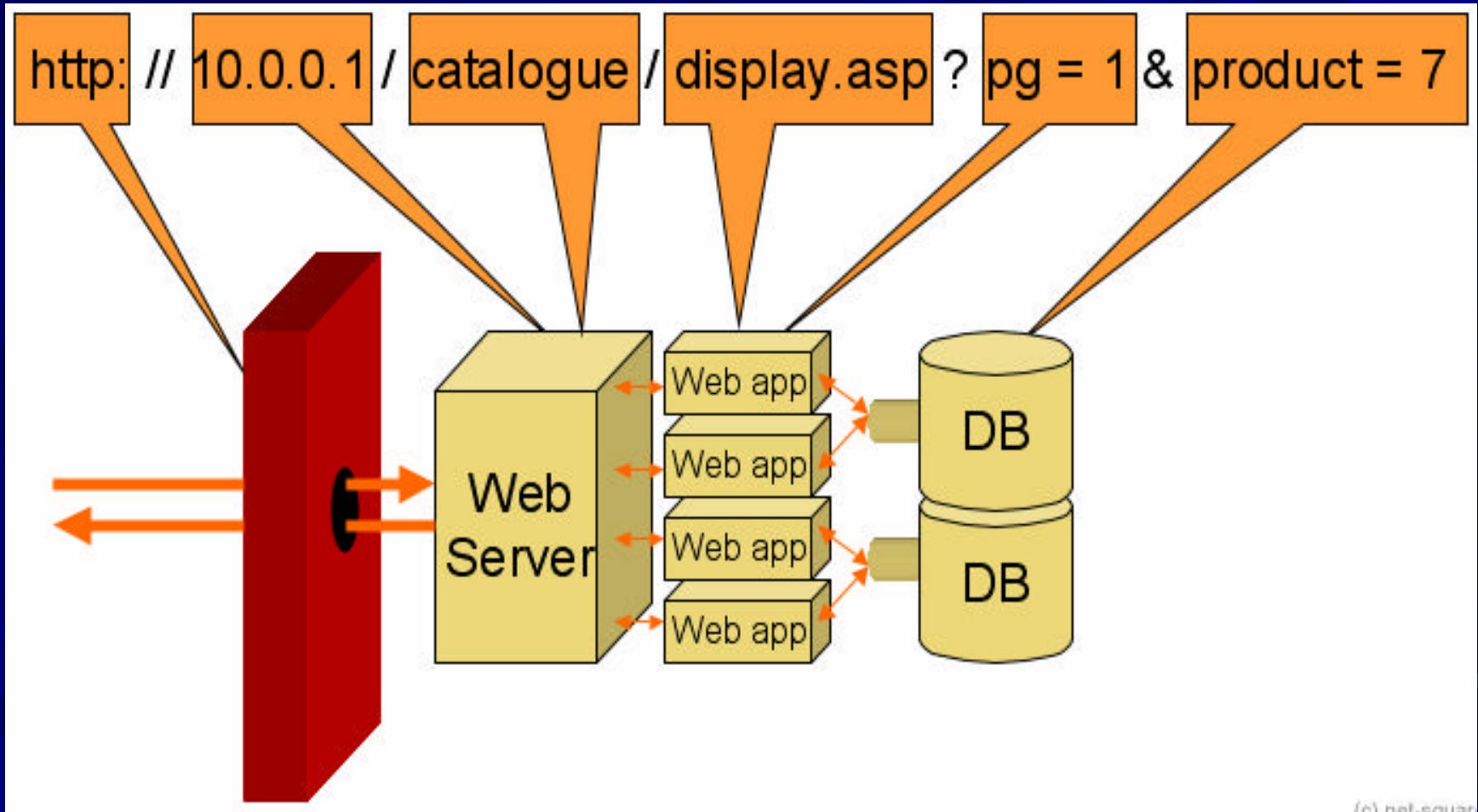


Web server

Komponen System Web



Mapping URL System Web



Web Server vs Web Application

■ Web Application:

- Menggunakan bahasa Pemrograman(misal ASP, PHP, Java, .Net, Perl atau C)

■ Web Server:

- Melayani permintaan client dan meneruskan ke aplikasi yang sesuai selanjutnya diproses oleh aplikasi yang sesuai (misal IIS, Apache, thttpd dll.)

■ Web Application tidak bisa jalan tanpa Web Server

■ Web Server bisa bekerja tanpa Web Application (Tapi hanya bisa melayani web dengan content statis)

URLs

- Each file managed by a server has a unique name called a URL (Universal Resource Locator)
- URLs for static content:
 - `http://www.cs.cmu.edu:80/index.html`
 - `http://www.cs.cmu.edu/index.html`
 - `http://www.cs.cmu.edu`
 - Identifies a file called `index.html`, managed by a Web server at `www.cs.cmu.edu` that is listening on port 80.
- URLs for dynamic content:
 - `http://www.cs.cmu.edu:8000/cgi-bin/adder?15000&213`
 - Identifies an executable file called `adder`, managed by a Web server at `www.cs.cmu.edu` that is listening on port 8000, that should be called with two argument strings: 15000 and 213.

Instalasi dan konfigurasi

- Download file `httpd-2.0.55.tar.gz` dari `apache.org`
- Extract file tersebut
`#tar -xzvf httpd-2.0.55.tar.gz`
- Pindah ke directory `httpd-2.0.55`
`#cd httpd-2.0.55`
- `#!/configure`
tunggu sampai keluar prompt lagi
- `#make`
tunggu sampai keluar prompt
- `#make install`
setelah keluar prompt kembali, program `apache2` sudah terinstall di `/usr/local/apache2`

Susunan directory

- bin → file-file program
- build
- cgi-bin → script cgi
- conf → tempat konfigurasi
- error → error message
- htdocs → dokumen yang akan dipublish
- icons → gambar-gambar kecil
- include
- lib
- logs
- man → manual pages
- manual → dokumentasi
- modules → module

Directory yang digunakan

- bin
- htdocs
- conf

Konfigurasi apache2

- Diletakkan di dalam sub dir **conf**
- Di dalam file `httpd.conf`
- Menggunakan directives

Konfigurasi Dasar

- ServerRoot → Root dari apache2
- Listen → Port http (biasanya 80)
- DocumentRoot → lokasi dokumen HTML
- Users → apache akan dijalankan sebagai user dan group ini
- Group
- ServerName → assign Nama mesin
- ServerAdmin → email address admin
- DirectoryIndex → file default yang akan dijalankan

Menjalankan Apache2

- Cukup mengganti Konfigurasi dasar dengan isian yang sesuai, apache2 sudah siap untuk dijalankan dengan kemampuan dasar yaitu bisa di browse
- Menjalankan :
`#!/usr/local/apache2/bin/apachectl start`
- Mengentikan:
`#!/usr/local/apache2/bin/apachectl stop`

Meletakkan Dokumen HTML

- Dokumen-dokumen HTML yang akan dipublikasikan ke Internet, bisa diletakkan di dalam Directory yang disebutkan di DocumentRoot

Per User Web Directories

- Menggunakan module `mod_userdir`
- Secara default directory user yang bisa di-browse adalah directory user yang di bawah `/home`
- Directive `UserDir` diisi dengan nama directory yang diinginkan

Per User web Directory

- jika path yang diberikan tidak didahului dengan slash, maka diasumsikan bahwa itu adalah directory path yang relative terhadap home dir dari user tertentu
- **UserDir public_html**
- `http://example.com/~rbowen/file.html` → `/home/rbowen/public_html/file.html`

Per User Web Directory

- jika path diberikan dengan slash, maka directory path akan dibuat menggunakan path tsb, plus username yang ditentukan
- **UserDir /var/html**
- `http://example.com/~rbowen/file.html` →
`/var/html/rbowen/file.html`

Per User Web Directory

- jika sebuah path di sediakan dengan * (asterisk) maka * tersebut akan diganti dengan username yang ditentukan
- **UserDir /var/www/*/docs**
- <http://example.com/~rbowen/file.html> -->
/var/www/rbowen/docs/file.html

Virtual Host

■ IP Based

- Berbasis IP
- Satu IP untuk tiap Website

■ Name Based

- Berbasis Nama
- Lebih dari satu website per IP address

■ Dalam Kasus ini kita akan menggunakan Name Based VH

Name Based Virtual Host

- Name based VH, memungkinkan kita untuk bisa mempunyai banyak site walaupun kita hanya punya satu IP
- Per site identik dengan satu Block directive VirtualHost

Per user home directory

- Edit httpd.conf
- Cari UserDir public_html hapus tanda #
- Hapus baris yang ada "UserDir disable"
- Keluar dari httpd.conf
- Restart apache dengan `/etc/rc.d/init.d/httpd restart`

- Menambahkan user:
- `Useradd <namauser>`
- `Passwd <namauser>`
- Rubah mode access homedir user tersebut :
 - `#chmod a+x /home/<namauser>`

Buatlah directory public_html di dalam homedir user
Masukkan file HTML ke dalam direktori tersebut

Name Based Virtual Host

■ NameVirtualHost *:80

<VirtualHost *:80>

```
ServerName www.domain.tld
ServerAlias domain.tld *.domain.tld
DocumentRoot /www/domain
ErrorLog /www/domain/error
```

■ </VirtualHost>

<VirtualHost *:80>

```
ServerName www.otherdomain.tld
DocumentRoot /www/otherdomain
ErrorLog /www/otherdomain/error
```

■ </VirtualHost>

Konfigurasi VH

■ <VirtualHost *:80>

 ServerName [nama_mesin]

 ServerAlias [alias]

 DocumentRoot [letak directory site]

 ErrorLog [file untuk mencatat error]

■ </VirtualHost>