

Webbutveckling med PHP

# PHP

Utbildare: Mikael Olsson

[mikael.olsson@emmio.se](mailto:mikael.olsson@emmio.se)

076-174 90 43

# NACKADEMIN

# Dagens ämnen

- Formalia
- Variabler
- Utskrift
- Kontrollstrukturer
- Utvecklingsmiljö

# Kursplan

# PHP

- "PHP [...] är ett populärt skriptspråk som främst körs på webbservrar för att driva internetsajter med dynamiskt innehåll (det vill säga innehåll som genereras från till exempel databas eller besökarens formulärdata). Det finns dock också en CLI-applikation för PHP som gör att skript kan köras direkt, utan webbrowser." - <https://sv.wikipedia.org/wiki/PHP>

# Skriptspråk

- "Skriptspråk är inom datavetenskap en benämning på "små" högnivåspråk inriktade på specialiserade uppgifter inom redan befintliga miljöer, i motsats till systemspråk, som används för programmering av tillämpningsprogram. Det finns ingen exakt avgränsning mellan systemspråk och skriptspråk, men skriptspråk är ofta interpreterande programspråk som använder dynamisk typning."
  - <https://sv.wikipedia.org/wiki/Skriptspr%C3%A5k>

# Skriptspråk

- "I en interpreterande miljö tolkas programkoden samtidigt som programmet körs, till skillnad mot en kompilerande miljö, där koden översätts i en separat process till maskinkod."
  - [https://sv.wikipedia.org/wiki/Interpreterande\\_programspr%C3%A5k](https://sv.wikipedia.org/wiki/Interpreterande_programspr%C3%A5k)
- "Dynamiskt typade variabler kan referera till vilken datatyp som helst."
  - [https://sv.wikipedia.org/wiki/Typsystem#Dynamisk\\_typning](https://sv.wikipedia.org/wiki/Typsystem#Dynamisk_typning)

# PHP

- PHP-skript kan köras på webb och i terminalen, men en del saker fungerar olika på olika "plattformar".

# PHP & Visual Studio Code

- <https://code.visualstudio.com/docs/languages/php>
- För CLI-skript kan man oftast köra PHP-filer på samma sätt som vi gjorde i Node. VCS måste då känna till sökvägen till PHP, t ex:

Windows:

```
{  
  "php.validate.executablePath": "c:/php/php.exe"  
}
```

Linux and macOS:

```
{  
  "php.validate.executablePath": "/usr/bin/php"  
}
```



# Local by Flywheel

- Vi har tidigare använt Local by Flywheel. Bra lösning för att enkelt sätta upp siter.
- Nackdelen är att Local använder en virtuell dator som den installerar alla program på, så vi kommer inte att nå "php-programmet" från t ex Visual Studio Code.
- Vi får inte heller tillgång till webbservern.
- En lösning kan vara att manuellt installera PHP och en webbserver så att man har PHP installerat både på sin egen dator och på Locals virtuella dator.
- En annan lösning kan vara att installera en stack. ->

# Stack

- LAMP - Linux, Apache, MySQL, PHP/Perl/Python
- Uppsättning med de OpenSource-komponenter som traditionellt använts för att sätta upp en webbservice.
- Finns i massor av varianter med olika operativsystem, webbservrar, databaser och skriptspråk.

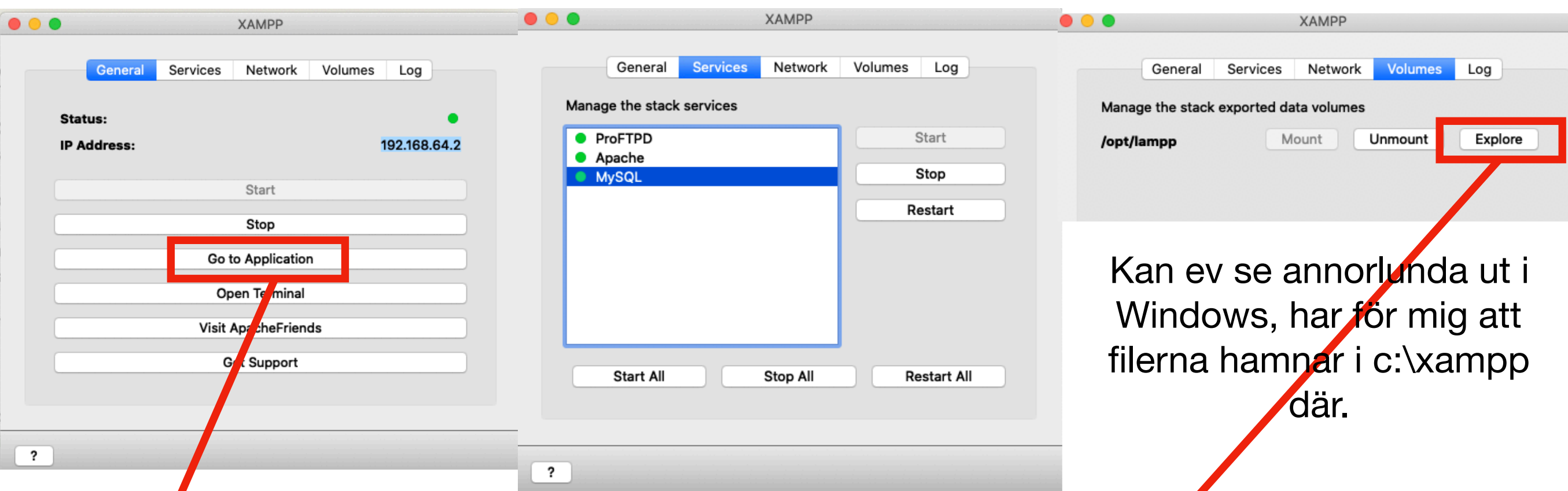
# Installera miljö

- Det finns massor av sätt att sätta upp fungerande miljöer. Ett vanligt sätt i utvecklingsmiljöer är att installera en stack, t ex:
  - XAMPP: <https://www.apachefriends.org/>
  - MAMP: <https://www.mamp.info/en/>
- Alla lösningar har sina starka och svaga sidor.

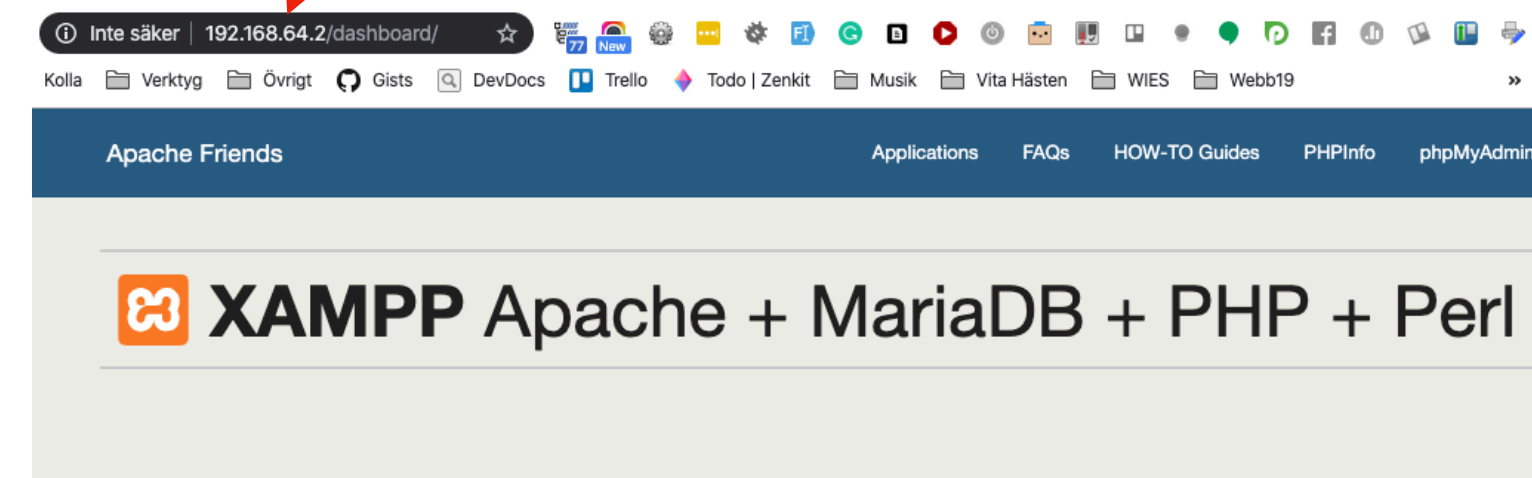
# Installera miljö

- Ni kan använda Local by Flywheel till projekt eller liknande om ni vill, men för att vi ska kunna undersöka hela processen behöver ni ändå att installera en stack av något slag, typ XAMPP, MAMP eller liknande.
- Er uppgift är att installera en stack. Det kan finnas fördelar med att vi väljer samma stack så vi kan hjälpa varandra bättre.

# XAMPP



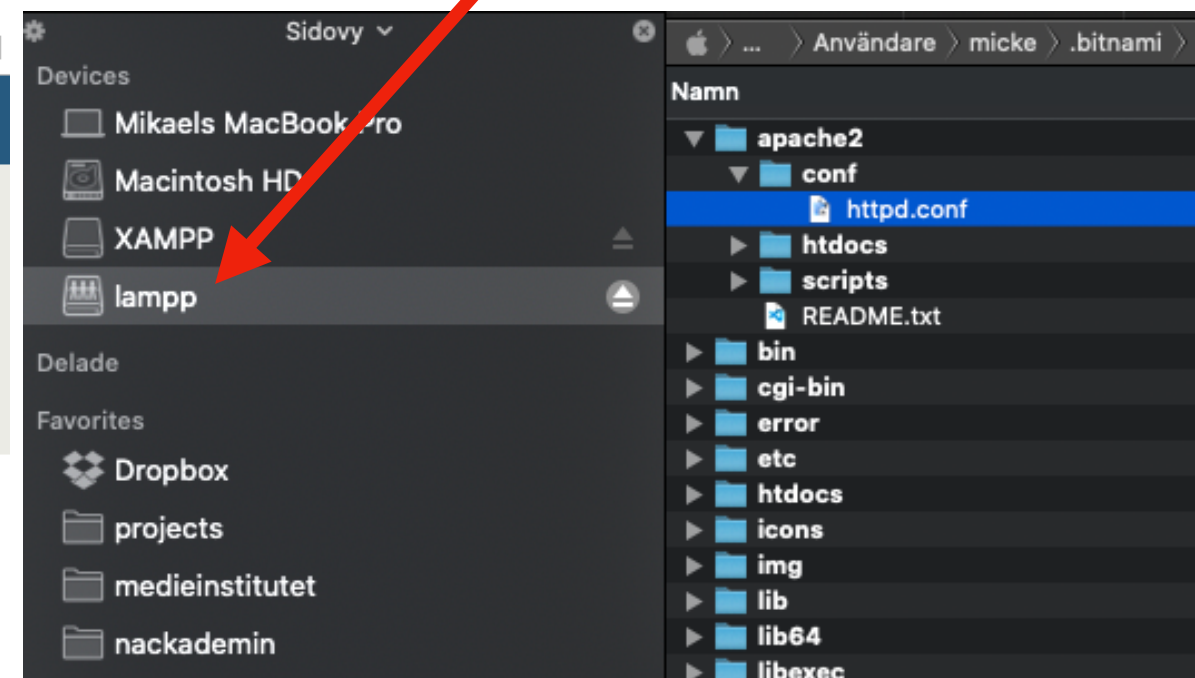
Kan ev se annorlunda ut i Windows, har för mig att filerna hamnar i c:\xampp där.



## Welcome to XAMPP for 7.4.1-0

You have successfully installed XAMPP on this system! Now you can start using Apache, MariaDB, PHP and other components. You can find more info in the [FAQs](#) section or check the [HOW-TO Guides](#) for getting started with PHP applications.

XAMPP is meant only for development purposes. It has certain configuration settings that make it easy to develop locally but that are



# Internets uppbyggnad

- Decentraliserat
- Kunna koppla upp via olika vägar
- IP-adress
- DNS



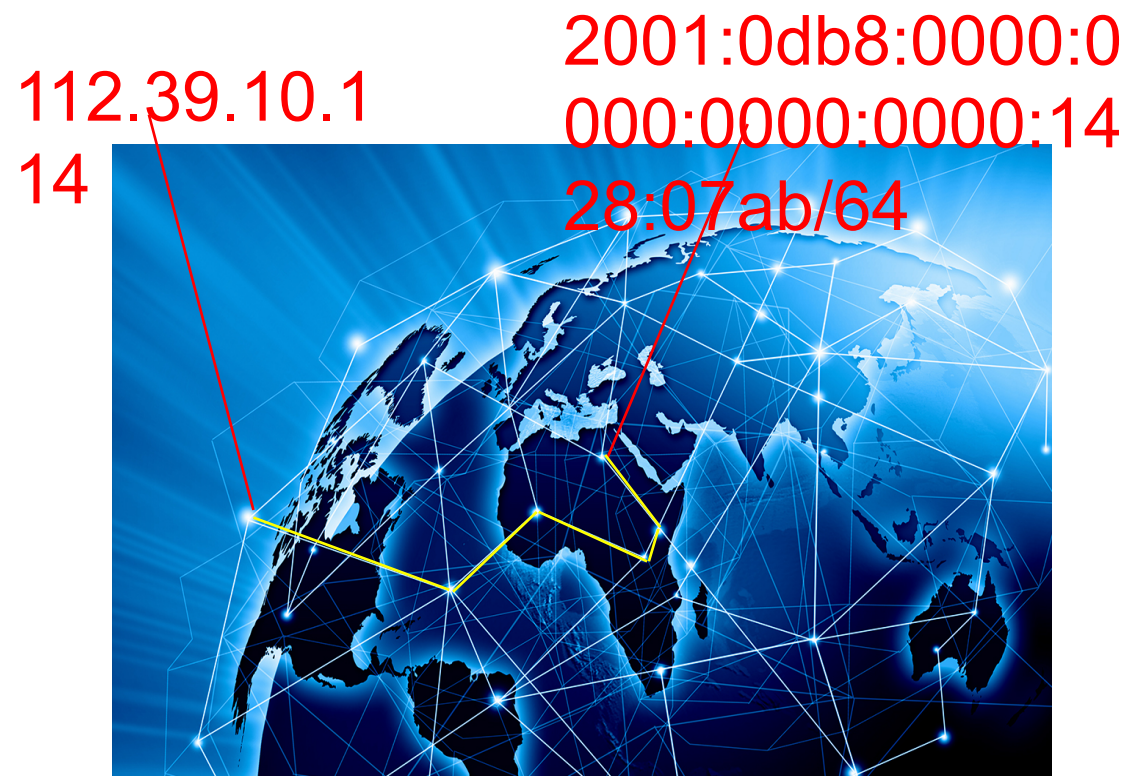
# Internet - decentraliserat

- Utvecklat av militären
- Om en stad wipas ut måste informationen fortfarande kunna komma fram





# Internet - IP-adress



- Alla maskiner som vill kunna kommunicera över internet behöver en adress.
- v4: nnn.nnn.nnn.nnn
- v6:  
xxxx:xxxx:xxxx:xxxx:xxxx:xxx  
x:xxxx:xxxx (/nn)



# Internet - DNS

- Domain Name System
- Adressbok för IP-adresser

Mickes dator

Sweet! Nu vet jag vilken IP-adress jag ska till.

[www.aftonbladet.se](http://www.aftonbladet.se)?

Yes, den har 104.20.55.70

[www.aftonbladet.se](http://www.aftonbladet.se)  
104.20.55.70

Jag har din sida, kommer här!

Nice! Sparar det ett tag.

Ingen aning, frågar vidare.



# Internet - http

- HyperText Transfer Protocol
  - används för att överföra webbsidor
  - definierar åtta kommandon
    - GET
    - HEAD
    - POST
    - PUT
    - DELETE
    - TRACE
    - OPTIONS
    - CONNECT
  - Svaret från webbservern innehåller en HTTP-statuskod
- 200: OK
- 301: Moved Permanently
- 401: Unauthorized
- 403: Forbidden
- 404: Not Found
- 500: Internal Server Error

# Internet - http

- Exempel klientförfrågan

```
GET /index.html HTTP/1.1  
Host: www.example.com
```

- Exempel serversvar
- (följs av en blankrad och texten i det efterfrågade dokumentet)

```
HTTP/1.1 200 OK  
Date: Mon, 23 May 2005 22:38:34 GMT  
Server: Apache/1.3.27 (Unix) (Red-Hat/Linux)  
Last-Modified: Wed, 08 Jan 2003 23:11:55 GMT  
Etag: "3f80f-1b6-3e1cb03b"  
Accept-Ranges: bytes  
Content-Length: 438  
Connection: close  
Content-Type: text/html; charset=UTF-8
```

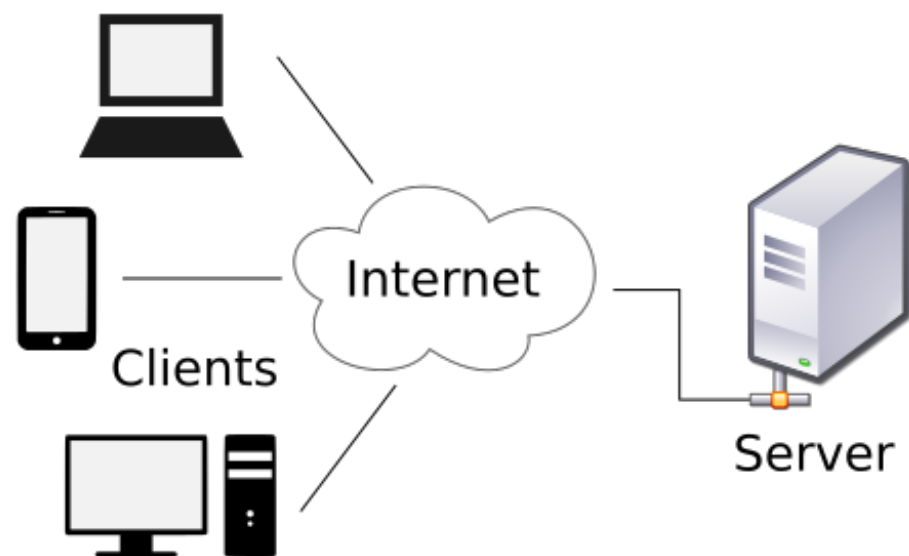
# Internet - https

- HyperText Transfer Protocol Secure
- Krypterad http
- Använder TLS, tidigare SSL
- Betrodd tredje part tillhandahåller ett undertecknat digitalt certifikat. Certifikatet installeras på webbservern och kontrolleras av webbläsaren med hjälp av den tredje partens lokalt installerade certifikat.
- Har varit dyrt och prestandaförsämrade, inte längre.

# Internet - http/2

- Nyaste versionen av http
- Utvecklades från Googles SPDY för att vara snabbare och säkrare
- https är en förutsättning

# Klient/server-modellen



Server utför något slags arbete, delar resurser

Klient är en beställare.

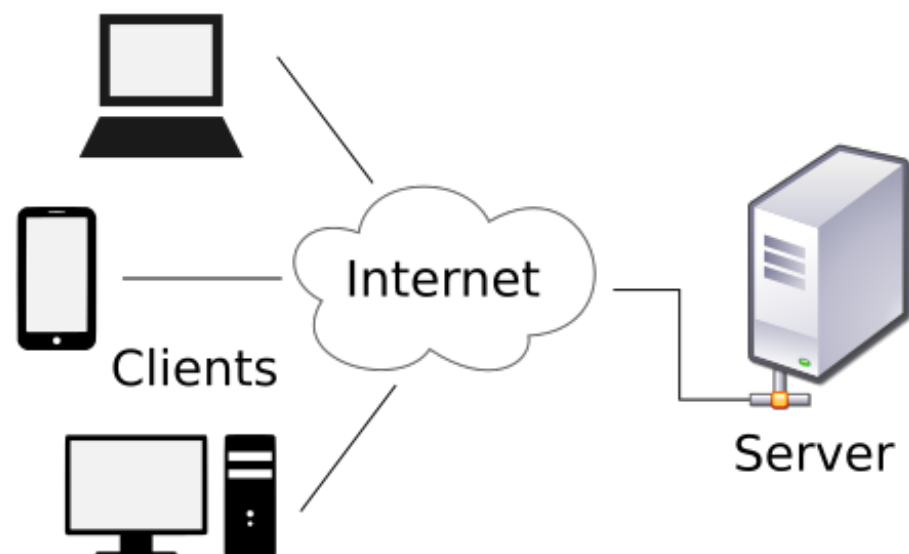
Ofta över ett nät, men inte alltid.

Klienten delar inte med sig av sina resurser, den begär en tjänst.

Klienten initierar kommunikation, servern inväntar den.

Ex: mail, nätverksskrivare, www.

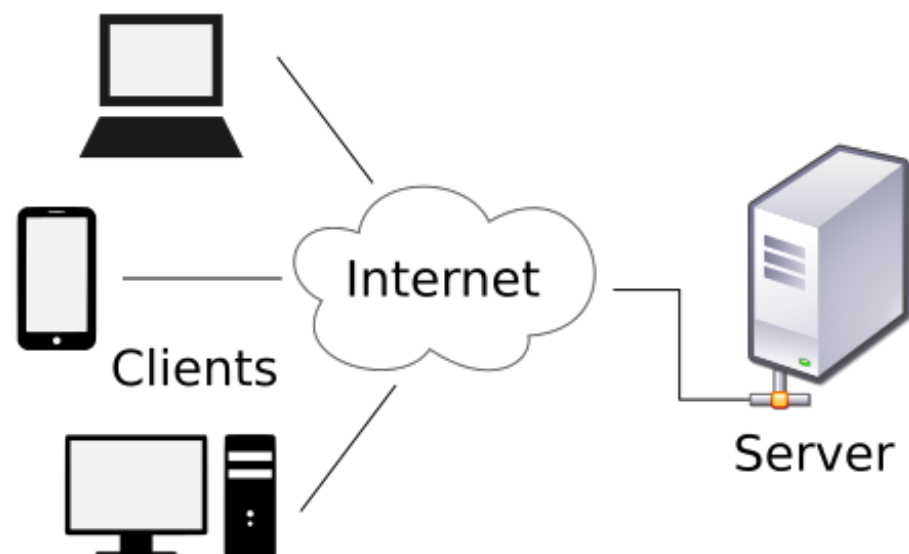
# Klient/server-modellen



- Server är inte en slags dator.
- Server är ett program.
  - Lyssnar på requests
  - Kan i princip köras på vilken dator som helst
  - Datorn som kör server-programmet kallas ofta för server.
  - Man kan köra en server på sin egen dator. Då agerar samma dator både server och klient.

**Förvirrande!**

# Klient/server-modellen



- Vad finns det för klienter?
  - Browsers
  - Mailklienter
  - Ftp-klient
  - SSH-klient



# Internet - webbrowser

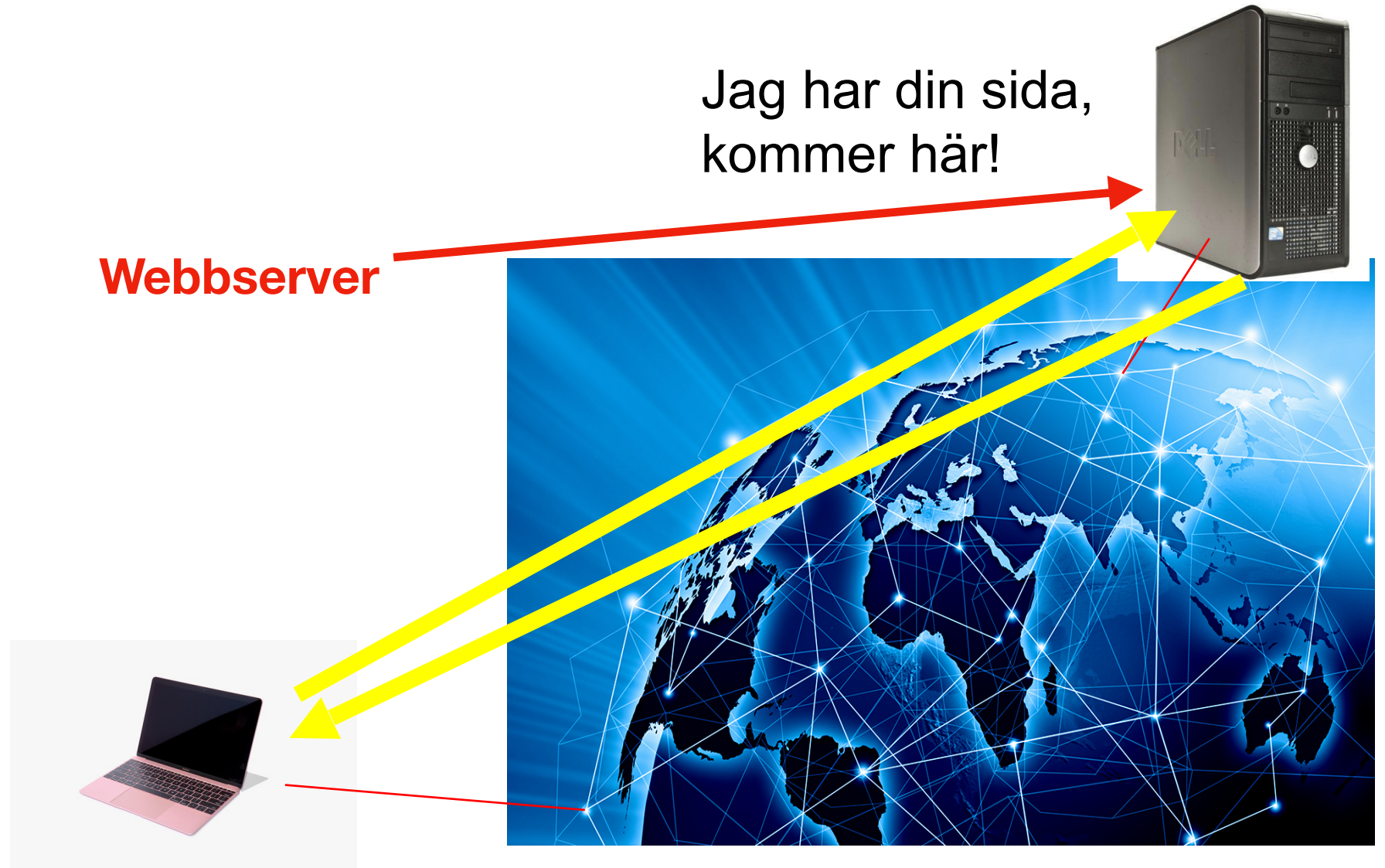
[www.aftonbladet.se](http://www.aftonbladet.se)

104.20.55.70

Jag har din sida,  
kommer här!

Webbserver

Mickes  
dator



# Lokal utvecklingsmiljö

- Webbserver
  - Lyssnar på anrop
  - Avgör vilken typ av resurs (fil) som efterfrågas.
  - Statiska filer (html, css, js, bilder) skickas tillbaka direkt.
  - PHP-filer måste *parsas* först.

# Lokal utvecklingsmiljö

- Databasserver
  - Hanterar databas.
  - Kan oftast kommunicera med webserver.

# Lokal utvecklingsmiljö

- LAMP
  - Linux
  - Apache
  - MySQL
  - PHP
- Finns i många varianter.
  - [https://en.wikipedia.org/wiki/LAMP \(software bundle\)](https://en.wikipedia.org/wiki/LAMP_(software_bundle))
- Populär variant för Windows: XAMPP
  - <https://www.apachefriends.org/>
- Mac: MAMP
  - <https://www.mamp.info/en/>

# Lokal utvecklingsmiljö

- Annan variant: Virtuellt dator
  - VirtualBox - <https://www.virtualbox.org/>
    - Låter användare installera en virtuellt dator på datorn.
  - Vagrant - <https://www.vagrantup.com/>
    - Låter användare skipta ihop utvecklingsmiljöer på en virtuellt dator.
  - <https://puphpet.com/>

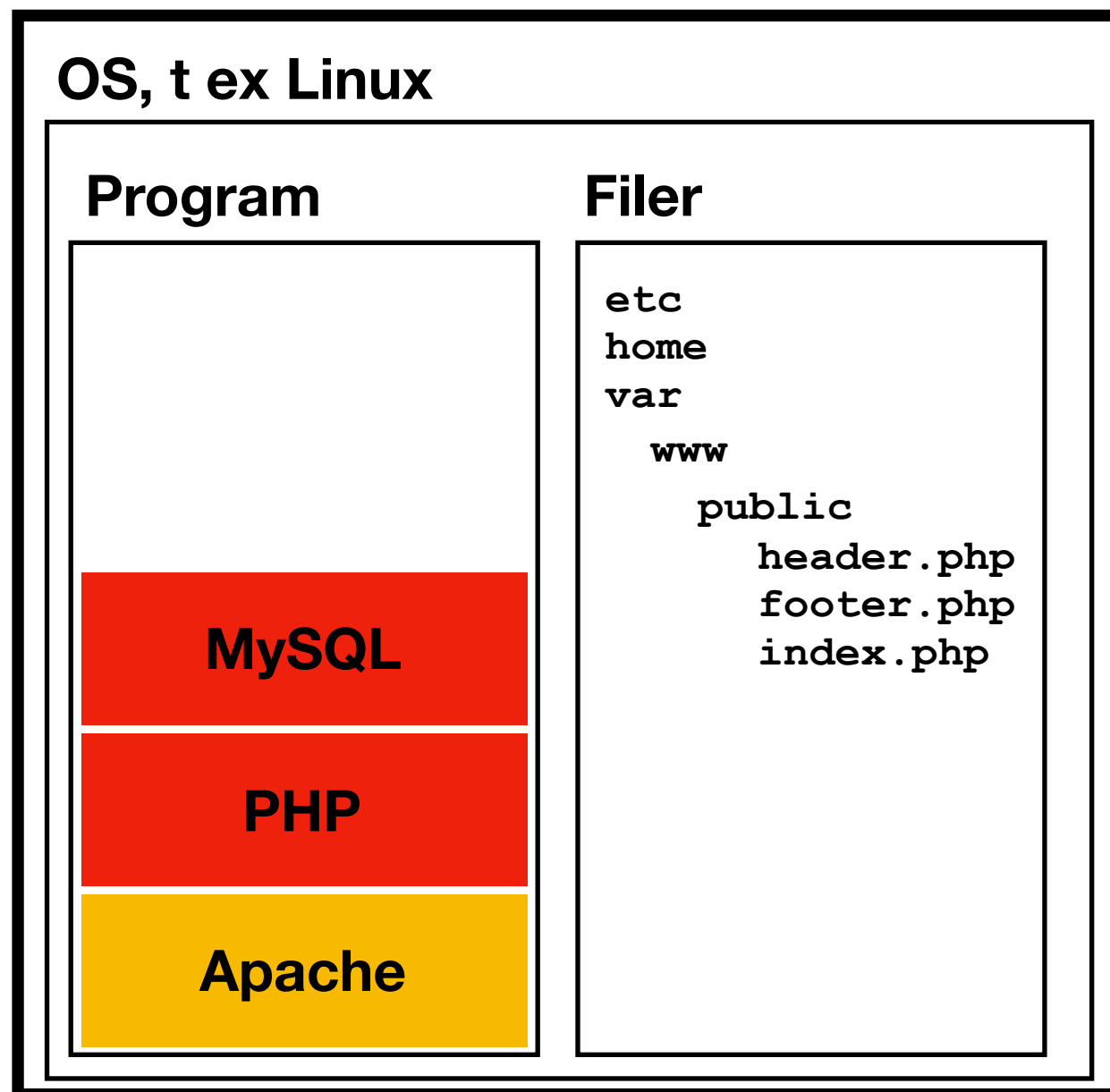
# Lokal utvecklingsmiljö

- En tredje variant: En annan virtuell dator
  - Docker - <https://www.docker.com/>
    - Typ som VirtualBox men med annan teknik.

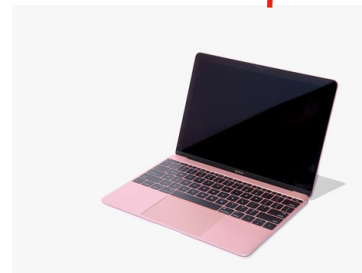
# Att köra php-filer

Dator

Internet

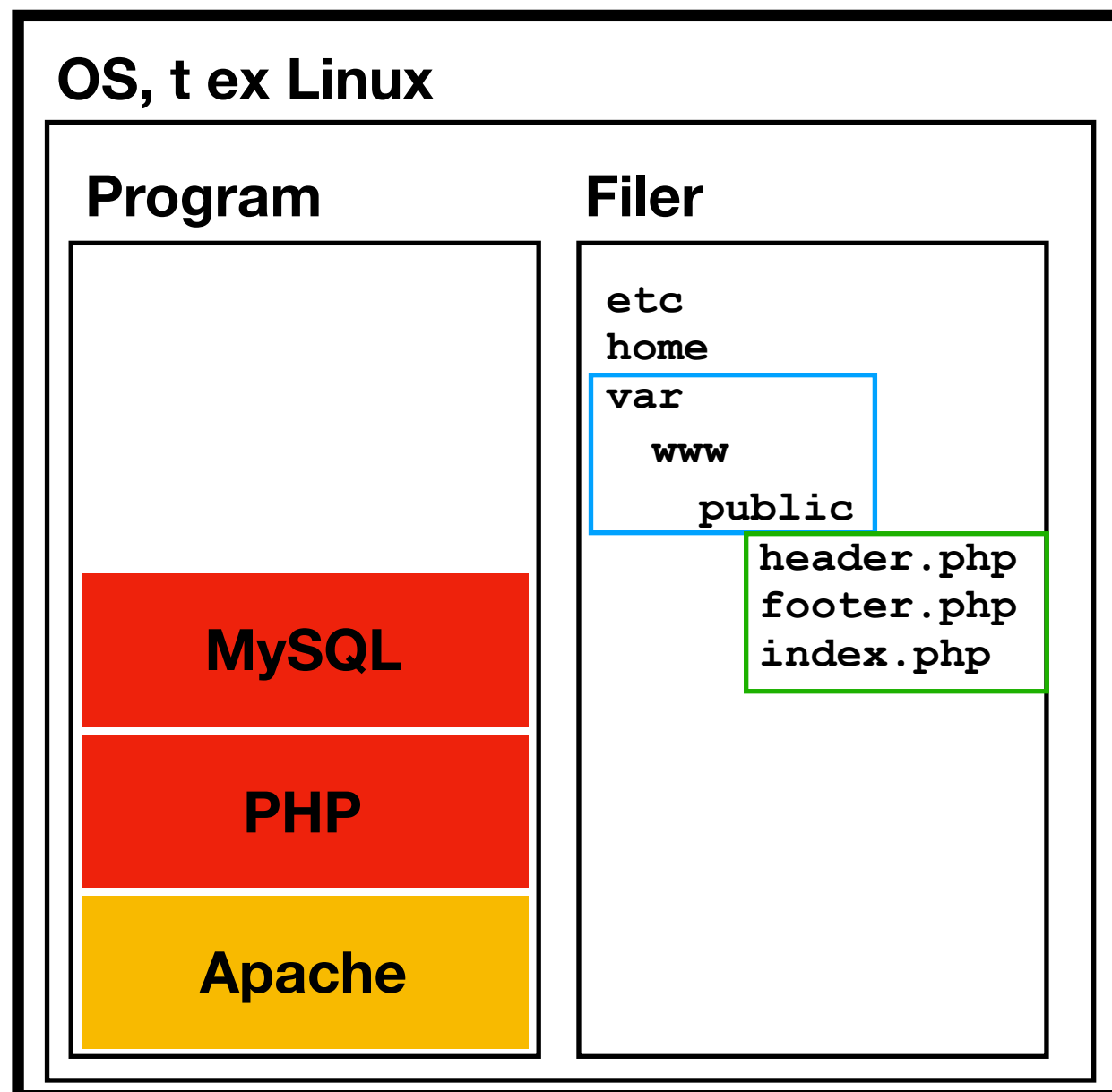


← → ↻ [example.com/](http://example.com/)



# Att köra php-filer

## Dator



## Apache kollar konfigurationen

```
<VirtualHost *:80>  
    DocumentRoot "/var/www/public/"  
    ServerName example.com  
    DirectoryIndex index.html index.php  
</VirtualHost>
```

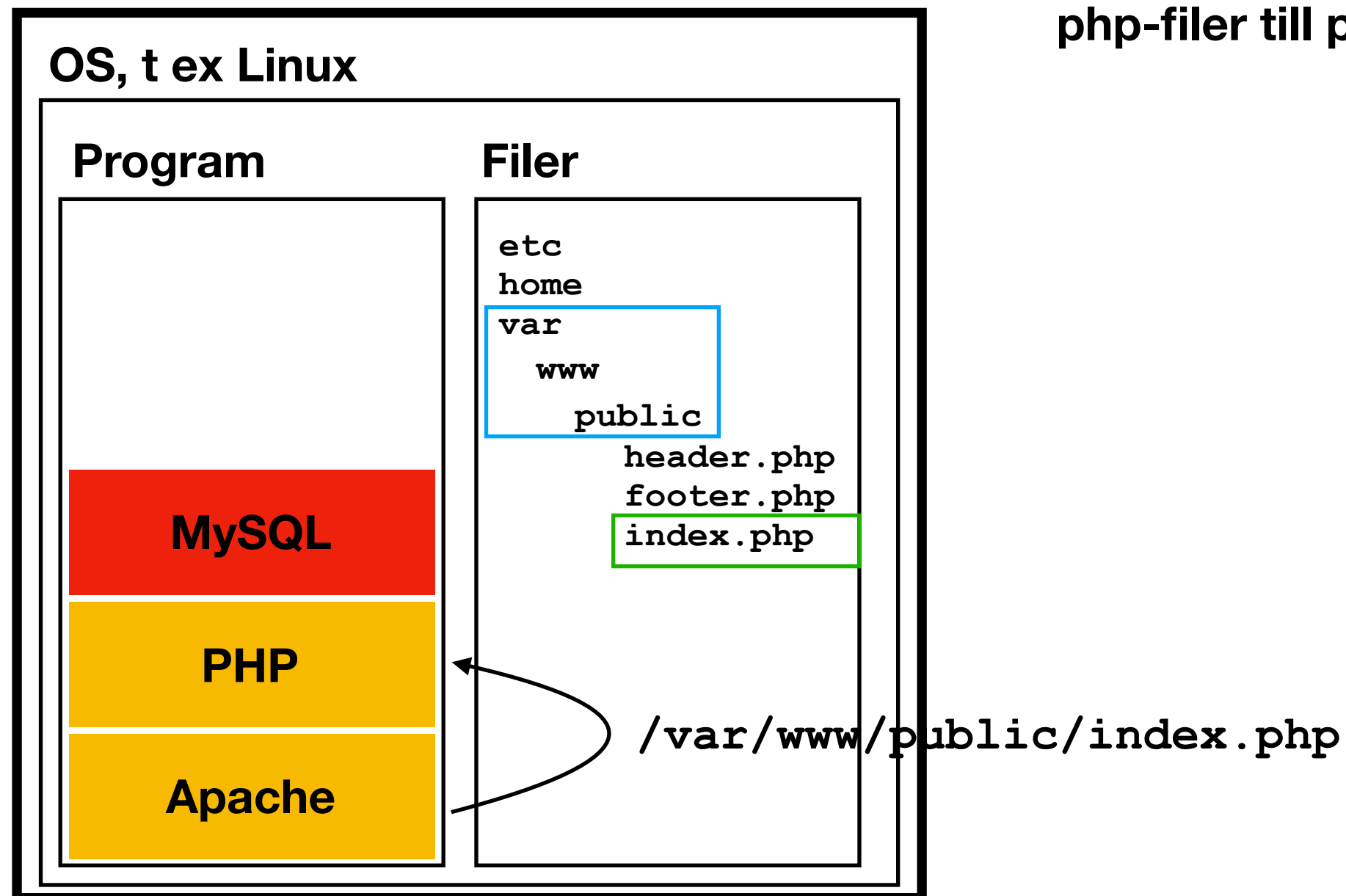
- Den här regeln ska gälla om besökaren går in på domänen example.com
- Filerna för den här webbplatsen ligger i mappen /var/www/public
- DirectoryIndex visar vilken fil som automatiskt ska visas.
  - Finns det någon index.html?
  - Finns det någon index.php?



# Att köra php-filer

Dator

Apache är konfigurerat för att skicka php-filer till php-applikationen.



# Att köra php-filer

## PHP

### index.php

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Min shop</title>
  <meta name="viewport" content="width=device-width,
initial-scale=1">
  <link rel="stylesheet" type="text/css" media="screen"
href="styles/main.css">
  <link rel="stylesheet" type="text/css" media="screen"
href="styles/styles/style01.css">
</head>
<body>
<?php
include 'functions.php';

hello();

?>
</body>
</html>
```



```
function hello() {
    echo "Hello world";
}

hello();
```

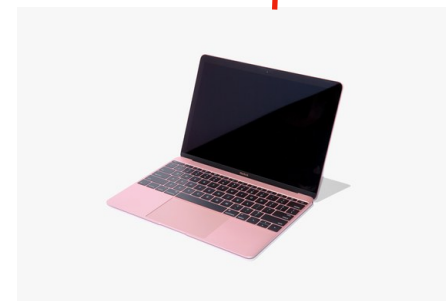
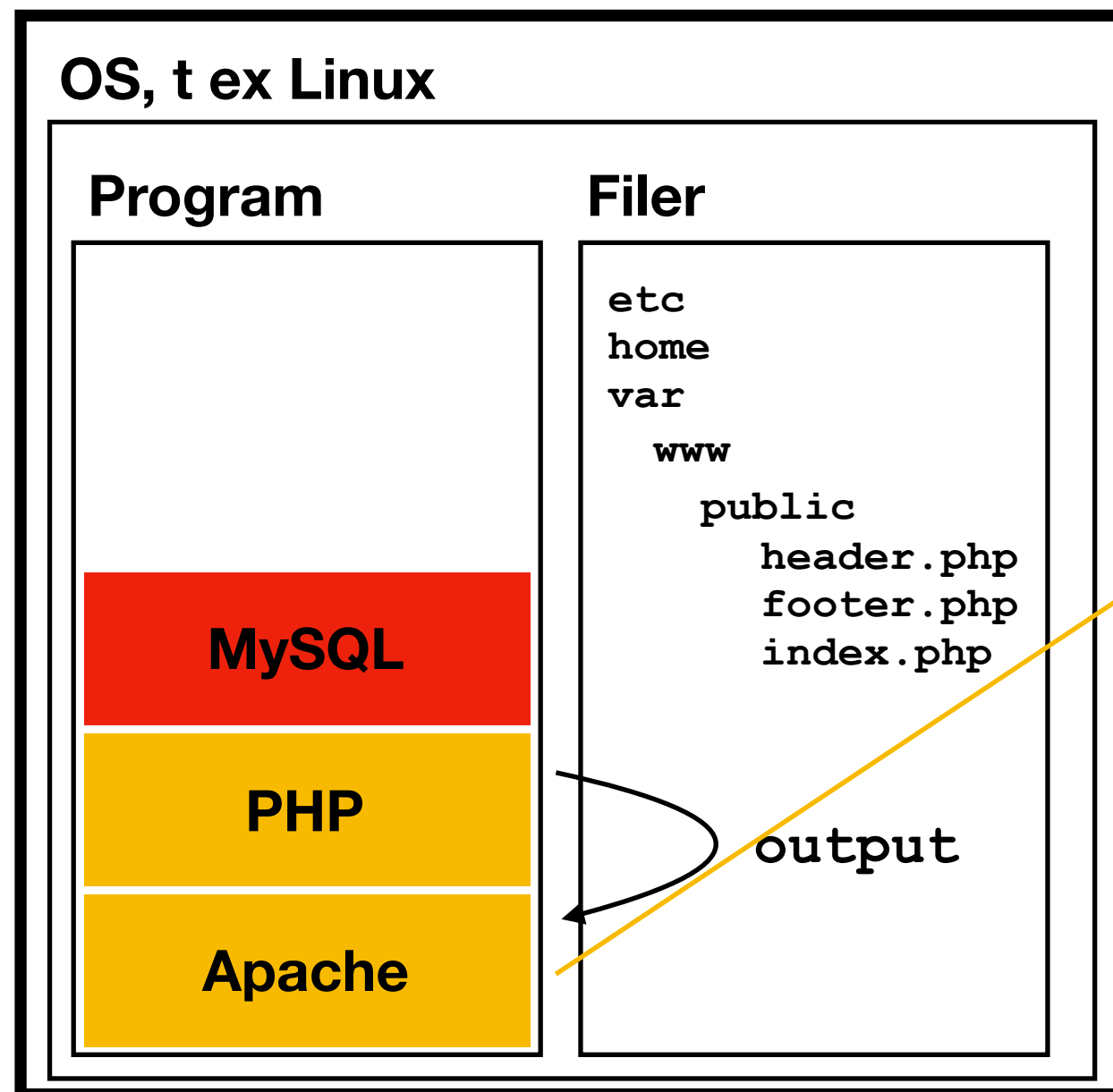
### Output till apache

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Min shop</title>
  <meta name="viewport" content="width=device-width,
initial-scale=1">
  <link rel="stylesheet" type="text/css" media="screen"
href="styles/main.css">
  <link rel="stylesheet" type="text/css" media="screen"
href="styles/styles/style01.css">
</head>
<body>
Hello world
</body>
</html>
```

# Att köra php-filer

Dator

Internet

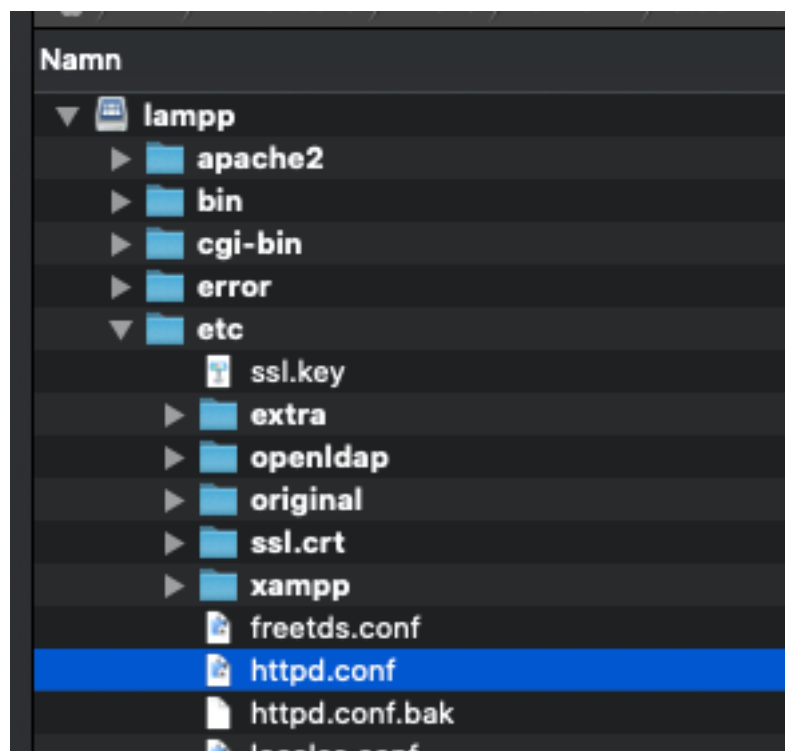


# Apache

- Apache är en av världens mest använda webbservrar.
- Andra webbservrar inkluderar nginx (min personliga favorit), LiteSpeed, IIS, GWS mm.
- En webbrowser tar emot requests och returnerar resurser, t ex filer med html, bilder, css-filer osv.
- Node har en inbyggd webbrowser. Det har även PHP, men man använder den sällan.

# Apache - config

- Konfigureringen av Apache (och även nginx, PHP mm) görs i speciella text-filer.
- För apache heter filen ofta apache2.conf, httpd.conf eller något liknande.



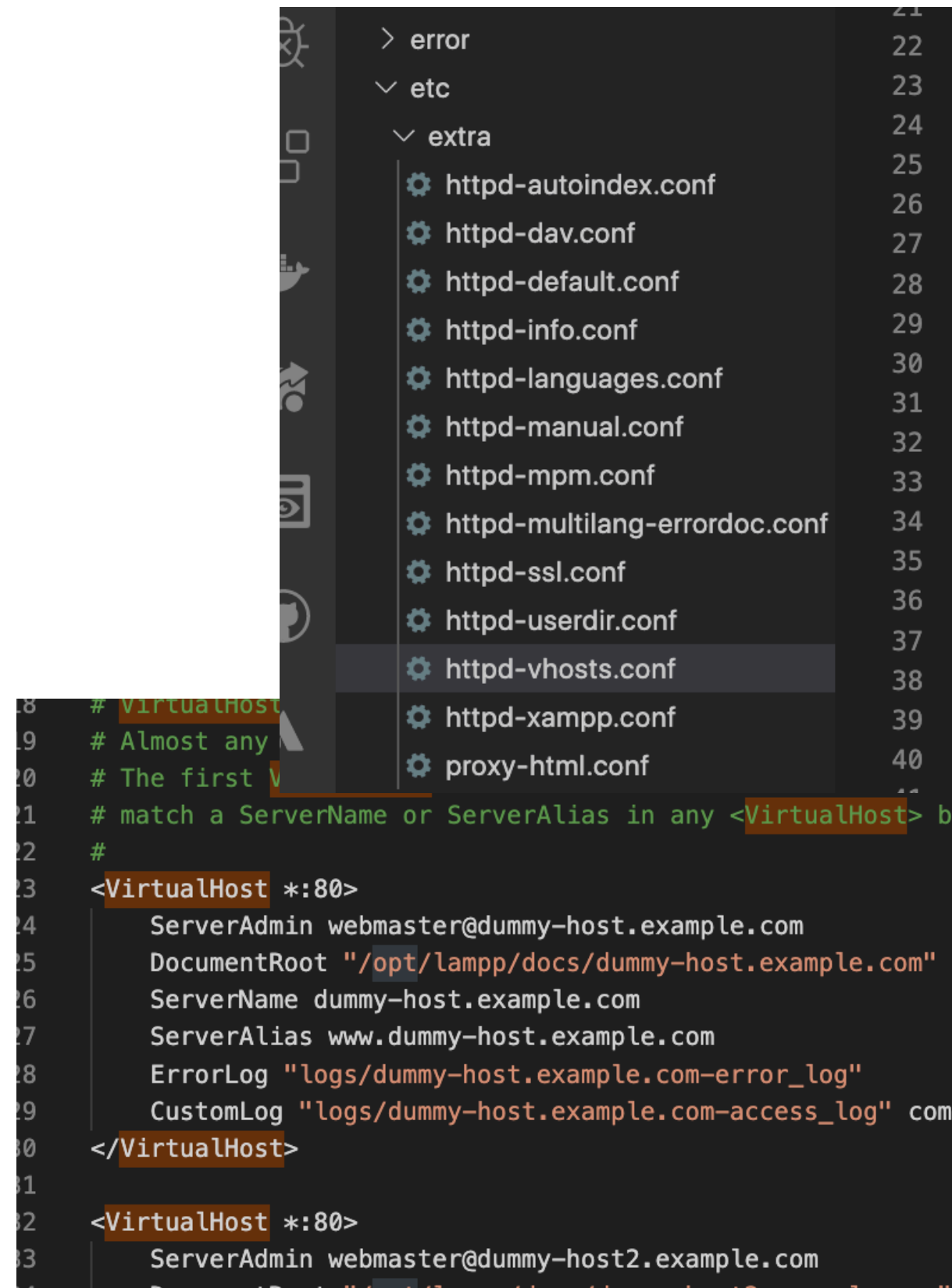
# Apache - config

- I Windows har ni troligen bara den Apache som ingår i XAMPP och därmed troligen bara en config-fil. (I C:\xampp\apache eller liknande.)
- I Mac/Linux finns det troligen flera Apache-installationer och då måste man först och främst veta vad binärfilen heter. (Oftast `httpd`.)
- Sedan tar man reda på vilken server systemet använder med `which` och vilken config-fil den använder med parametern `-V`.

```
> ~ which httpd
/usr/sbin/httpd
> ~ /usr/sbin/httpd -V
Server version: Apache/2.4.41 (Unix)
Server built:   Oct 17 2019 18:04:28
Server's Module Magic Number: 20120211:88
Server loaded:  APR 1.5.2, APR-UTIL 1.5.4
Compiled using: APR 1.5.2, APR-UTIL 1.5.4
Architecture:   64-bit
Server MPM:      prefork
                  threaded:  no
                  forked:    yes (variable process count)
Server compiled with....
-D APR_HAS_SENDFILE
-D APR_HAS_MMAP
-D APR_HAVE_IPV6 (IPv4-mapped addresses enabled)
-D APR_USE_FLOCK_SERIALIZE
-D APR_USE_PTHREAD_SERIALIZE
-D SINGLE_LISTEN_UNSERIALIZED_ACCEPT
-D APR_HAS_OTHER_CHILD
-D AP_HAVE_RELIABLE_PIPED_LOGS
-D DYNAMIC_MODULE_LIMIT=256
-D HTTPD_ROOT="/usr"
-D SUEXEC_BIN="/usr/bin/suexec"
-D DEFAULT_PIDLOG="/private/var/run/httpd.pid"
-D DEFAULT_SCOREBOARD="logs/apache_runtime_status"
-D DEFAULT_ERRORLOG="logs/error_log"
-D AP_TYPES_CONFIG_FILE="/private/etc/apache2/mime.types"
-D SERVER_CONFIG_FILE="/private/etc/apache2/httpd.conf"
```

# Apache - virtual hosts

- I Apache kan man sätta upp flera sites på samma server. Det enklaste sättet är att använda en VirtualHost.
- Varje VirtualHost-taggen innehåller konfiguration för en site.



```
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
# VirtualHost  
# Almost any  
# The first V  
# match a ServerName or ServerAlias in any <VirtualHost> b  
#  
23 <VirtualHost *:80>  
24     ServerAdmin webmaster@dummy-host.example.com  
25     DocumentRoot "/opt/lampp/docs/dummy-host.example.com"  
26     ServerName dummy-host.example.com  
27     ServerAlias www.dummy-host.example.com  
28     ErrorLog "logs/dummy-host.example.com-error_log"  
29     CustomLog "logs/dummy-host.example.com-access_log" com  
30 </VirtualHost>  
31  
32 <VirtualHost *:80>  
33     ServerAdmin webmaster@dummy-host2.example.com
```

# Sätta upp projekt

- Kommentera bort den befintliga VirtualHost:en och skapa en ny för ett projekt med valfritt namn. Följ instruktionerna här:
- <https://programmer.help/blogs/xampp-configuration-of-domain-name-based-virtual-host-under-macos.html>  
(Det står att den är för Mac men instruktionerna bör fungera för Windows också.)