

Backendprogrammering 1

NODE.JS

Utbildare: Mikael Olsson

mikael.olsson@emmio.se
076-174 90 43

NACKADEMIN

Kursplan

Internets uppbyggnad

- Decentraliserat
- Kunna koppla upp via olika vägar
- IP-adress
- DNS
- Ping/Traceroute
- http / https
- ftp / ft�
- ssh

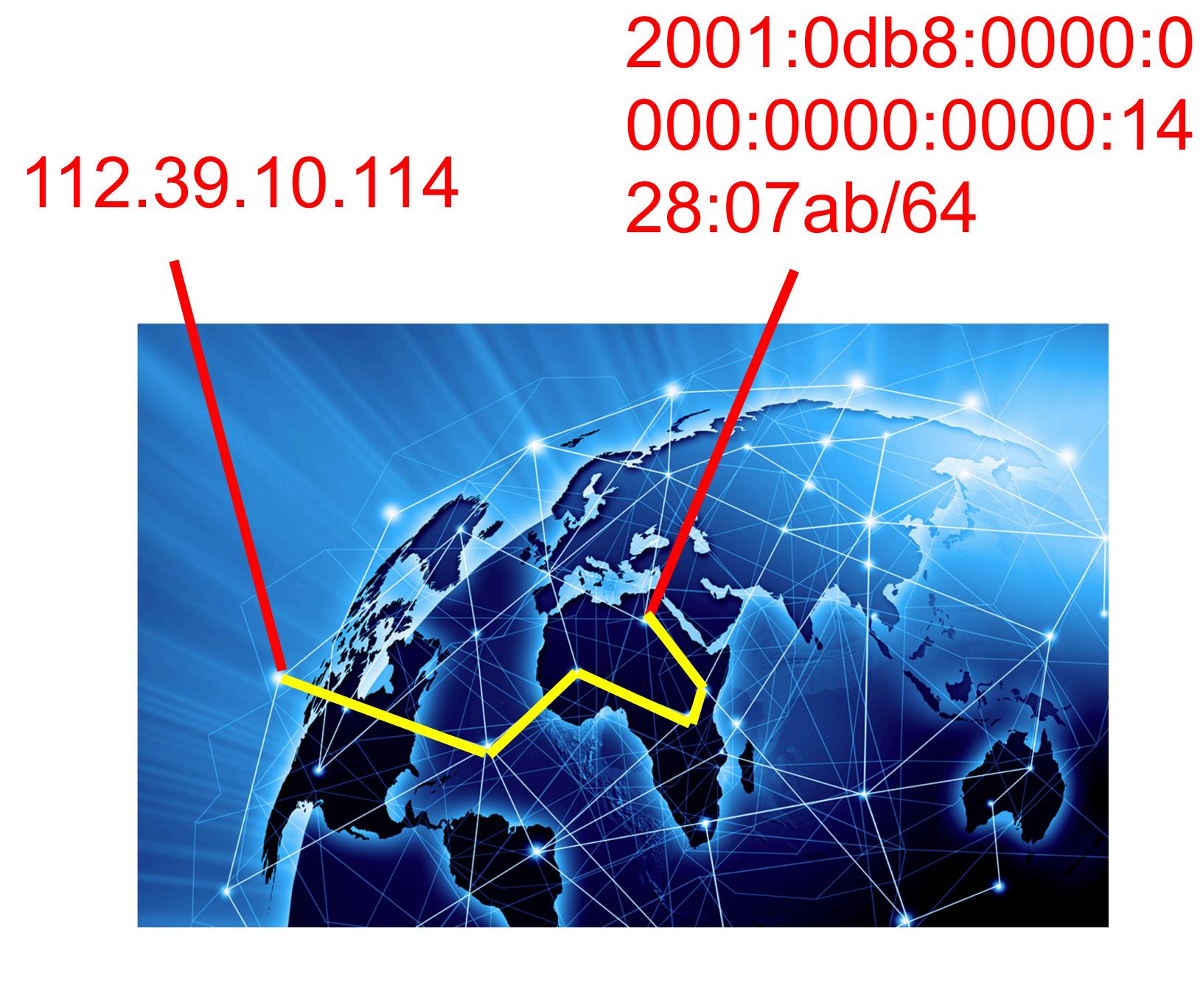


Internet - decentralisert

- 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:xxxx:
xxxx:xxxx (/nn)

Internet - DNS

- Domain Name System
- Adressbok för IP-adresser

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

Mickes
dator

www.aftonbladet.se?

Yes, den har
104.20.55.70

www.aftonbladet.se

104.20.55.70

Jag har din
sida,
kommer här!



DNS-
server

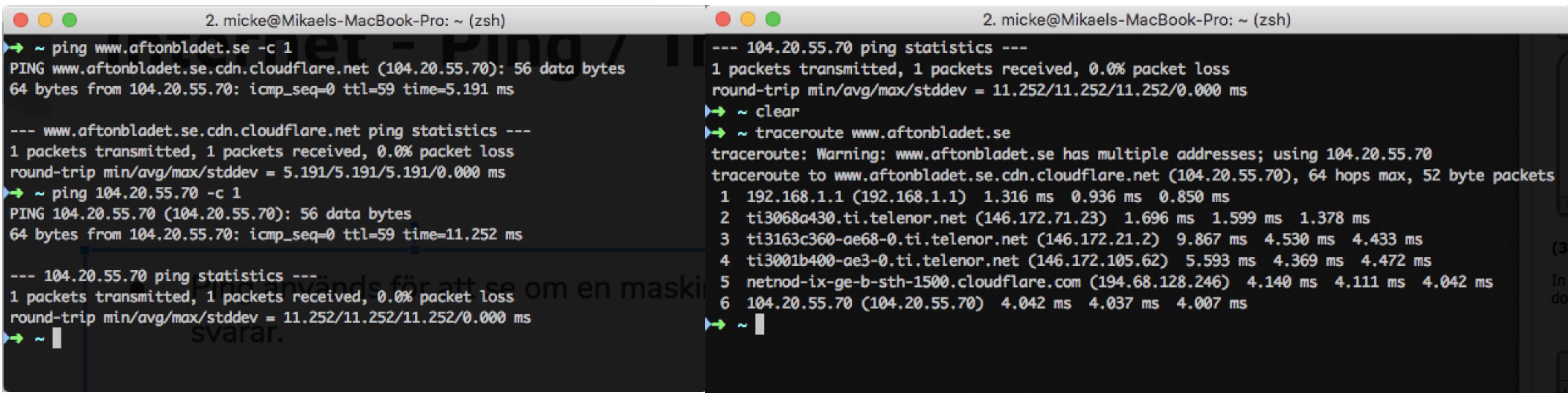
Nice! Sparar
det ett tag.

DNS-
server

Ingen aning,
frågar vidare.

Internet - Ping / Traceroute

- Ping används för att se om en maskin svarar.
- Vad innebär det om IP-adressen svarar men inte domänen?
- Traceroute används för att se vilka maskiner man passerar på vägen.
- Bra för att ta reda på var på vägen det strular.



The image shows two side-by-side terminal windows on a Mac OS X desktop. Both windows have a title bar '2. micke@Mikael-MacBook-Pro: ~ (zsh)' and a red, yellow, and green window control icon.

Terminal Window 1 (Left):

```
ping www.aftonbladet.se -c 1
PING www.aftonbladet.se.cdn.cloudflare.net (104.20.55.70): 56 data bytes
64 bytes from 104.20.55.70: icmp_seq=0 ttl=59 time=5.191 ms

--- www.aftonbladet.se.cdn.cloudflare.net ping statistics ---
1 packets transmitted, 1 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 5.191/5.191/5.191/0.000 ms

ping 104.20.55.70 -c 1
PING 104.20.55.70 (104.20.55.70): 56 data bytes
64 bytes from 104.20.55.70: icmp_seq=0 ttl=59 time=11.252 ms

--- 104.20.55.70 ping statistics ---
1 packets transmitted, 1 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 11.252/11.252/11.252/0.000 ms
```

Terminal Window 2 (Right):

```
--- 104.20.55.70 ping statistics ---
1 packets transmitted, 1 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 11.252/11.252/11.252/0.000 ms
clear
traceroute www.aftonbladet.se
traceroute: Warning: www.aftonbladet.se has multiple addresses; using 104.20.55.70
traceroute to www.aftonbladet.se.cdn.cloudflare.net (104.20.55.70), 64 hops max, 52 byte packets
1  192.168.1.1 (192.168.1.1)  1.316 ms  0.936 ms  0.850 ms
2  ti3068a430.ti.telenor.net (146.172.71.23)  1.696 ms  1.599 ms  1.378 ms
3  ti3163c360-ae68-0.ti.telenor.net (146.172.21.2)  9.867 ms  4.530 ms  4.433 ms
4  ti3001b400-ae3-0.ti.telenor.net (146.172.105.62)  5.593 ms  4.369 ms  4.472 ms
5  netnod-ix-ge-b-sth-1500.cloudflare.com (194.68.128.246)  4.140 ms  4.111 ms  4.042 ms
6  104.20.55.70 (104.20.55.70)  4.042 ms  4.037 ms  4.007 ms
```

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

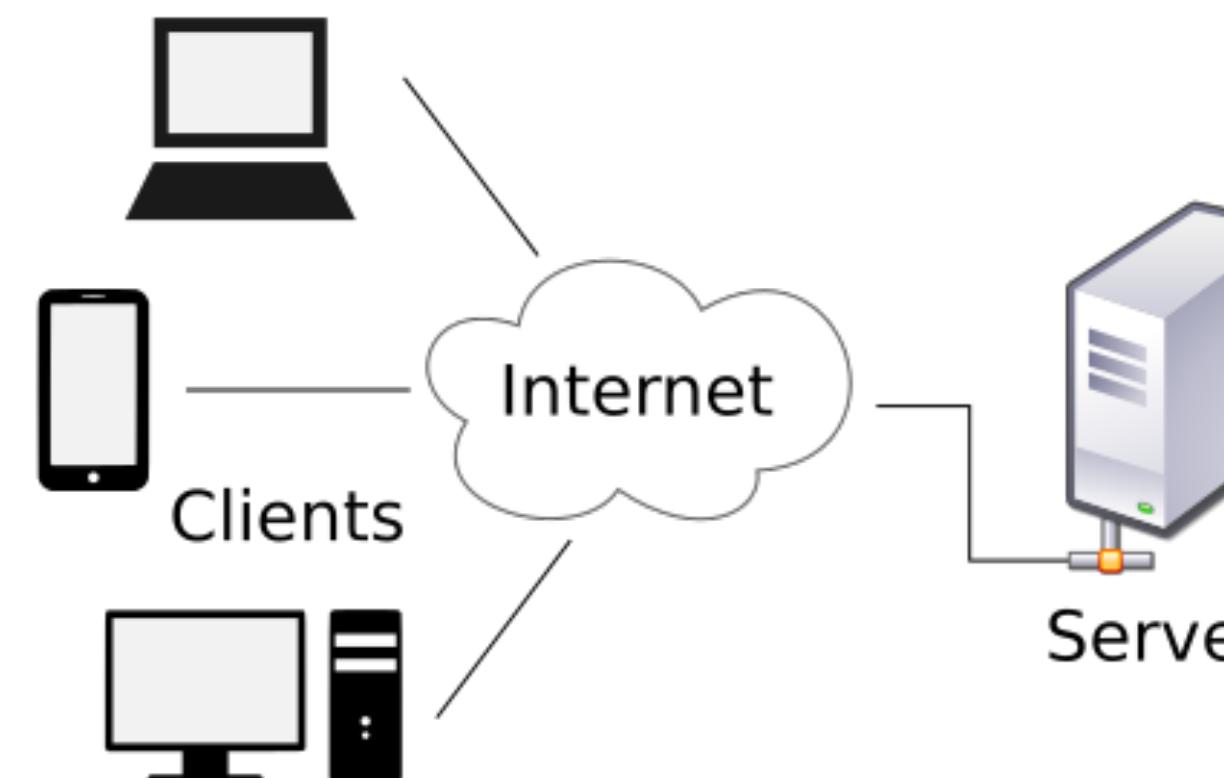
Internet - SSH

- Secure SHell
- Används för att logga in på säkert sätt mot andra maskiner
- Terminalverktyg

Internet - ftp / ftps / sftp

- File Transfer Protocol
- Protokoll för filöverföring
- FTPS = FTP over SSL
- SFTP = SSH FTP

Klient/server-modellen



Server utför något slags arbete, delar resurser

Klient är en beställare.

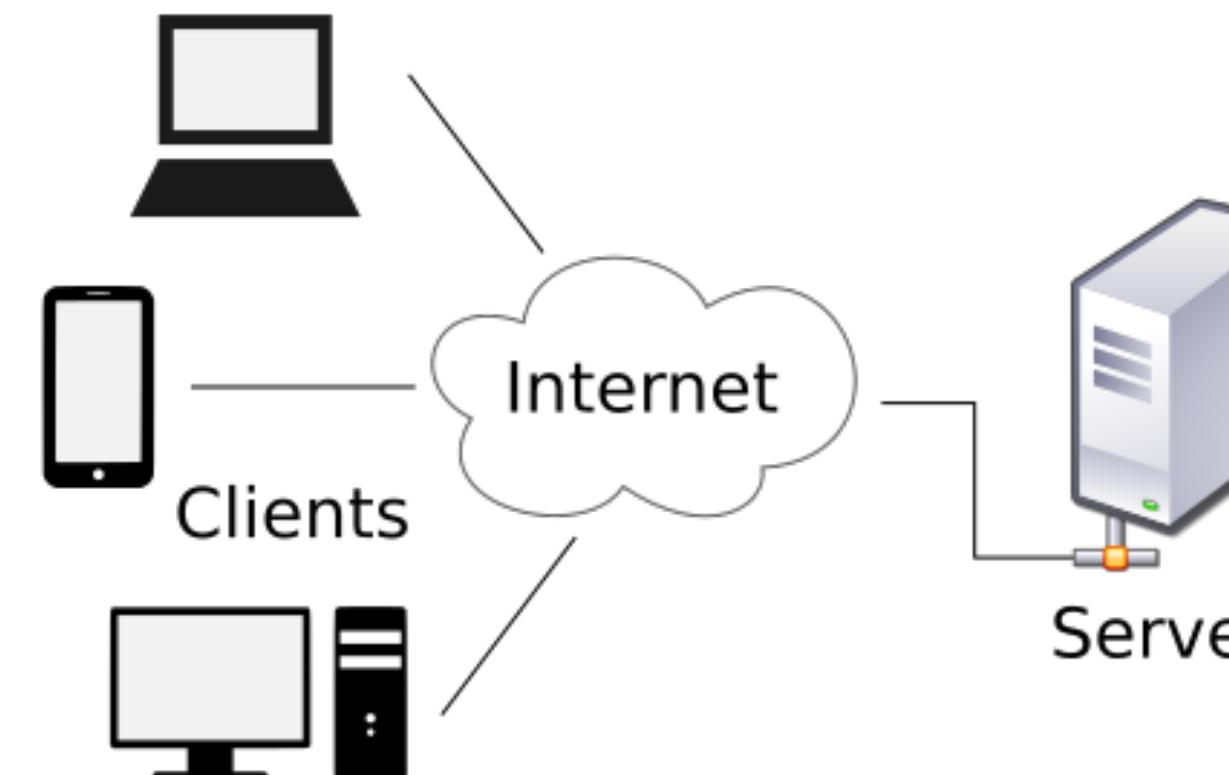
Ofta över ett nät, men inte alltid.

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

Klienten initierar kommunikation, servern inväntar den.

Ex: mail, nätverksskrivare, www.

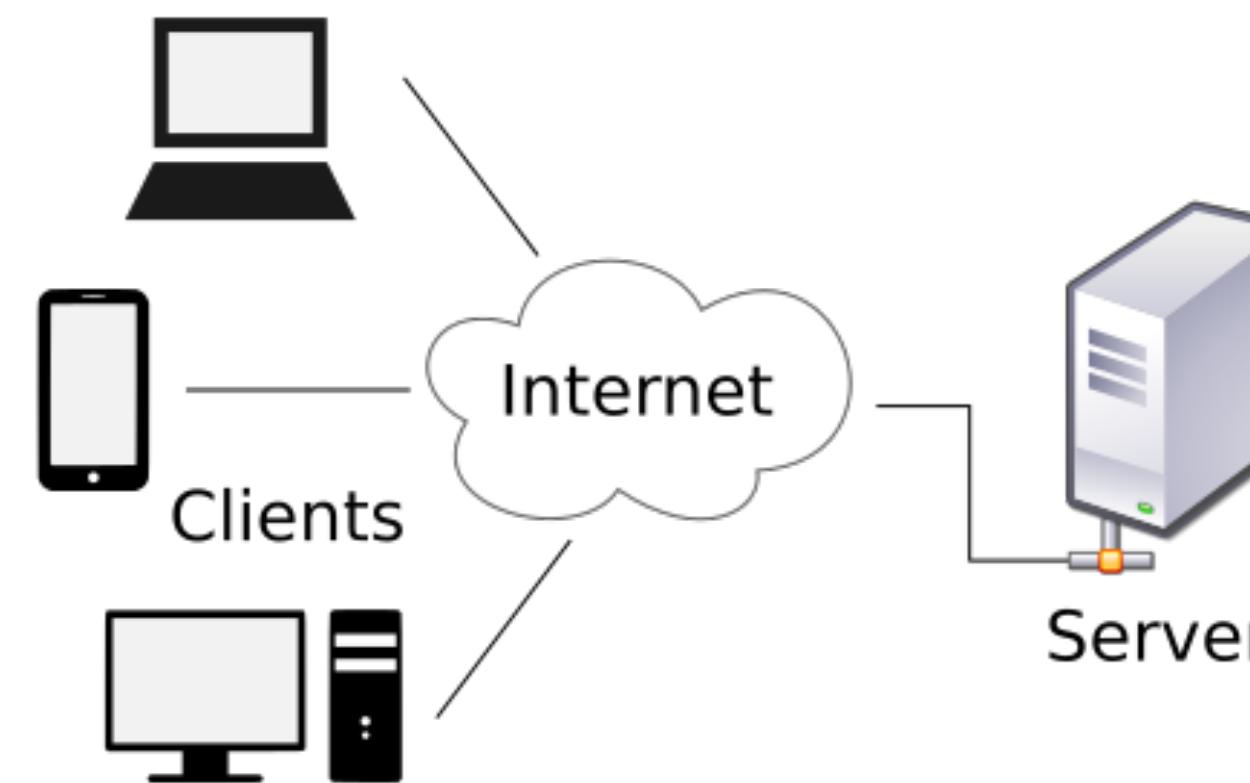
Klient/server-modellen



- 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 serverprogrammet 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 - webserver

www.aftonbladet.se

104.20.55.70



Webbserver

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

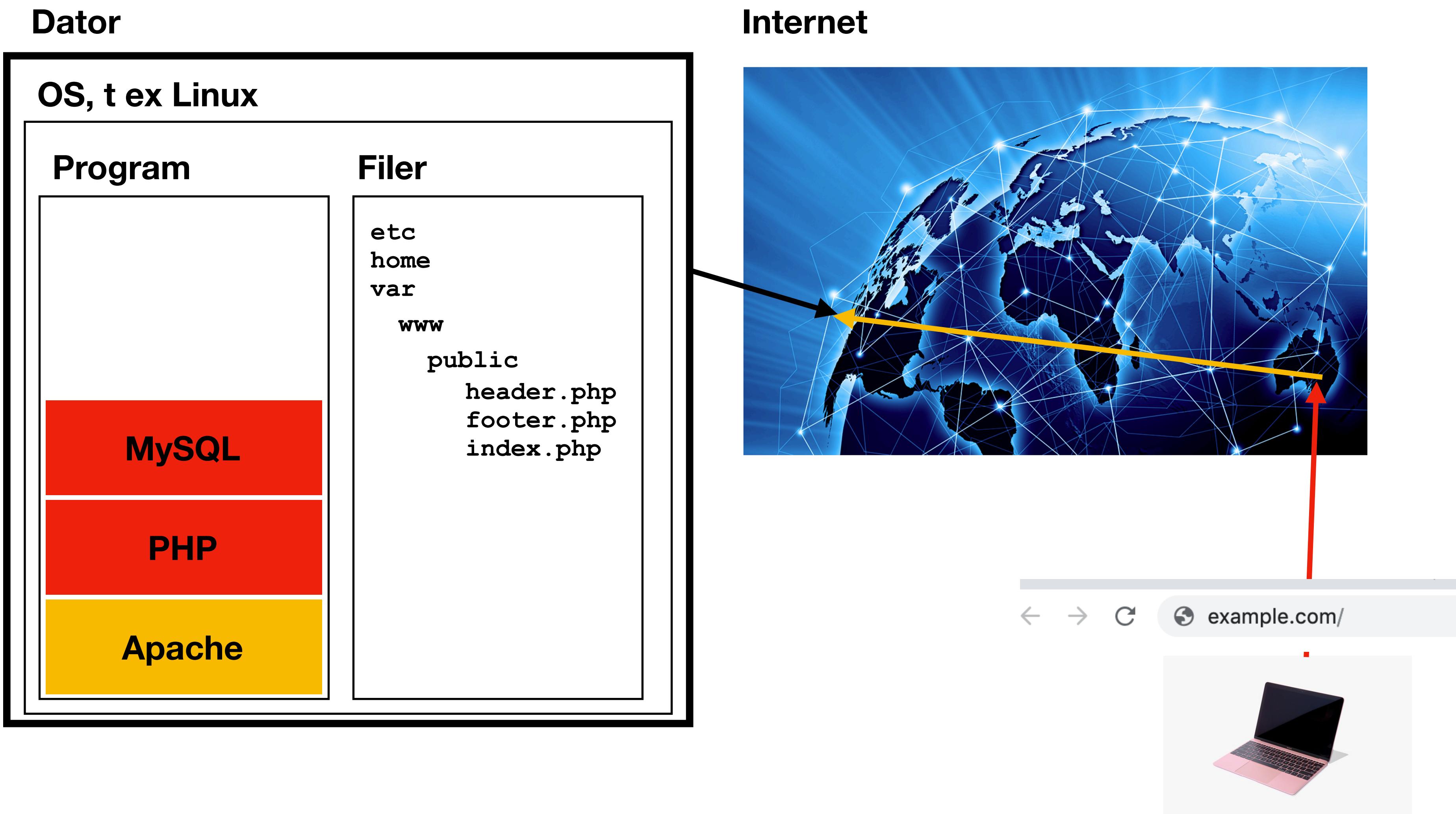
Databasserver

- Hanterar databas.
- Kan oftast kommunicera med webserver.

Lite om hur ett request fungerar

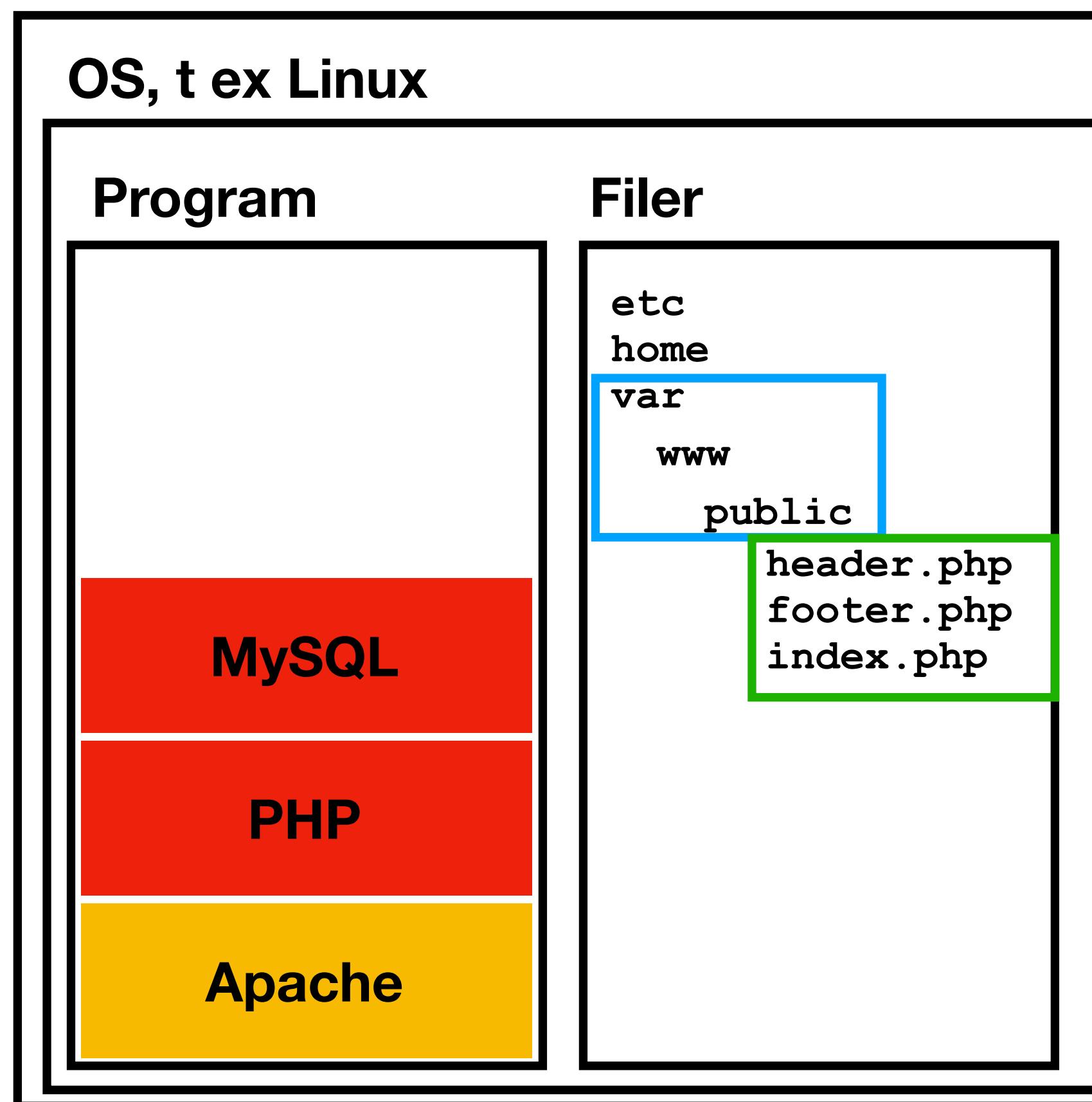
I detta exempel använder vi PHP, men principen är densamma för Node.

Att köra php-filer



Att köra php-filer

Dator

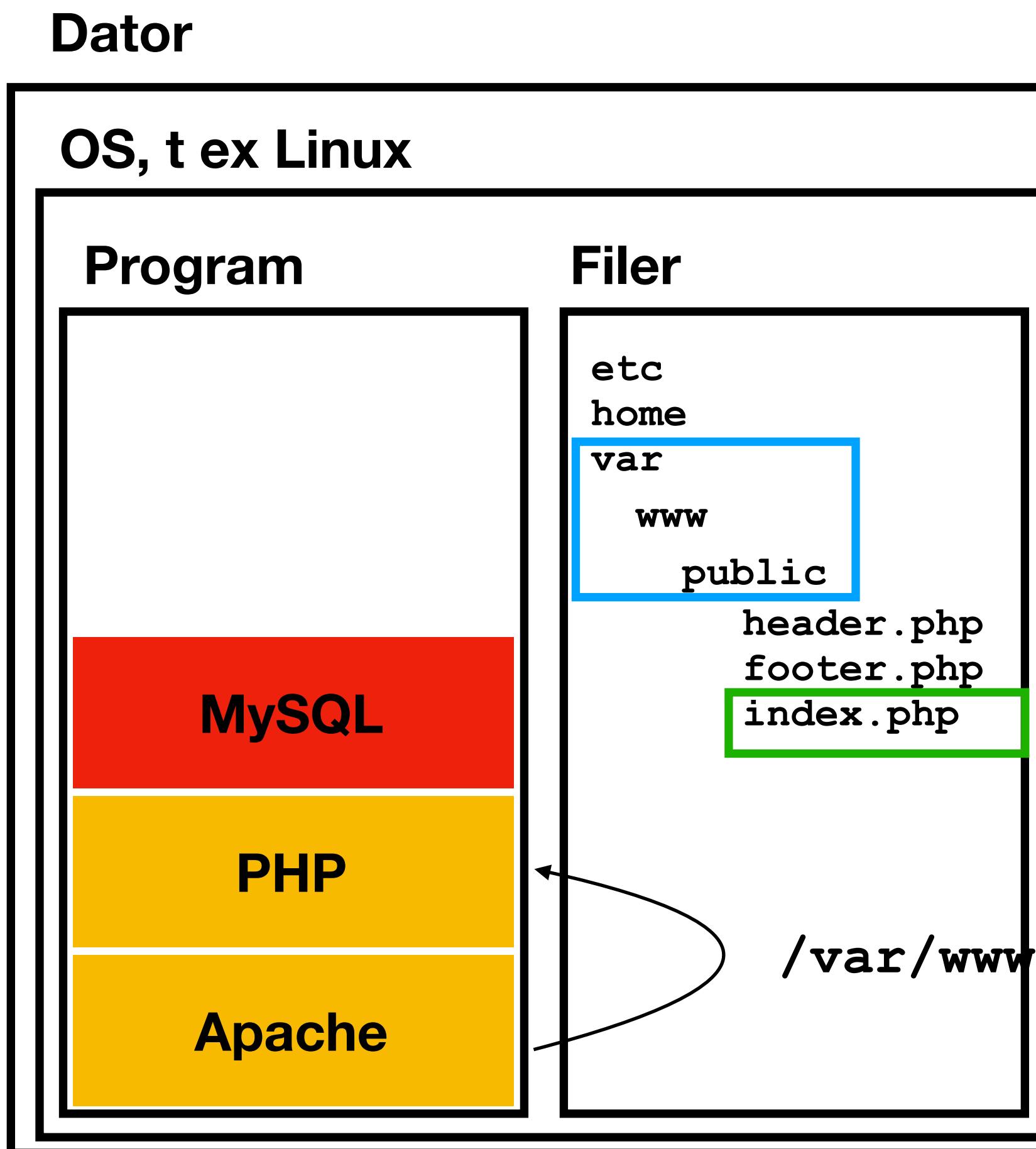


Apache kollar konfigurationen

```
<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



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

Att köra php-filer

PHP

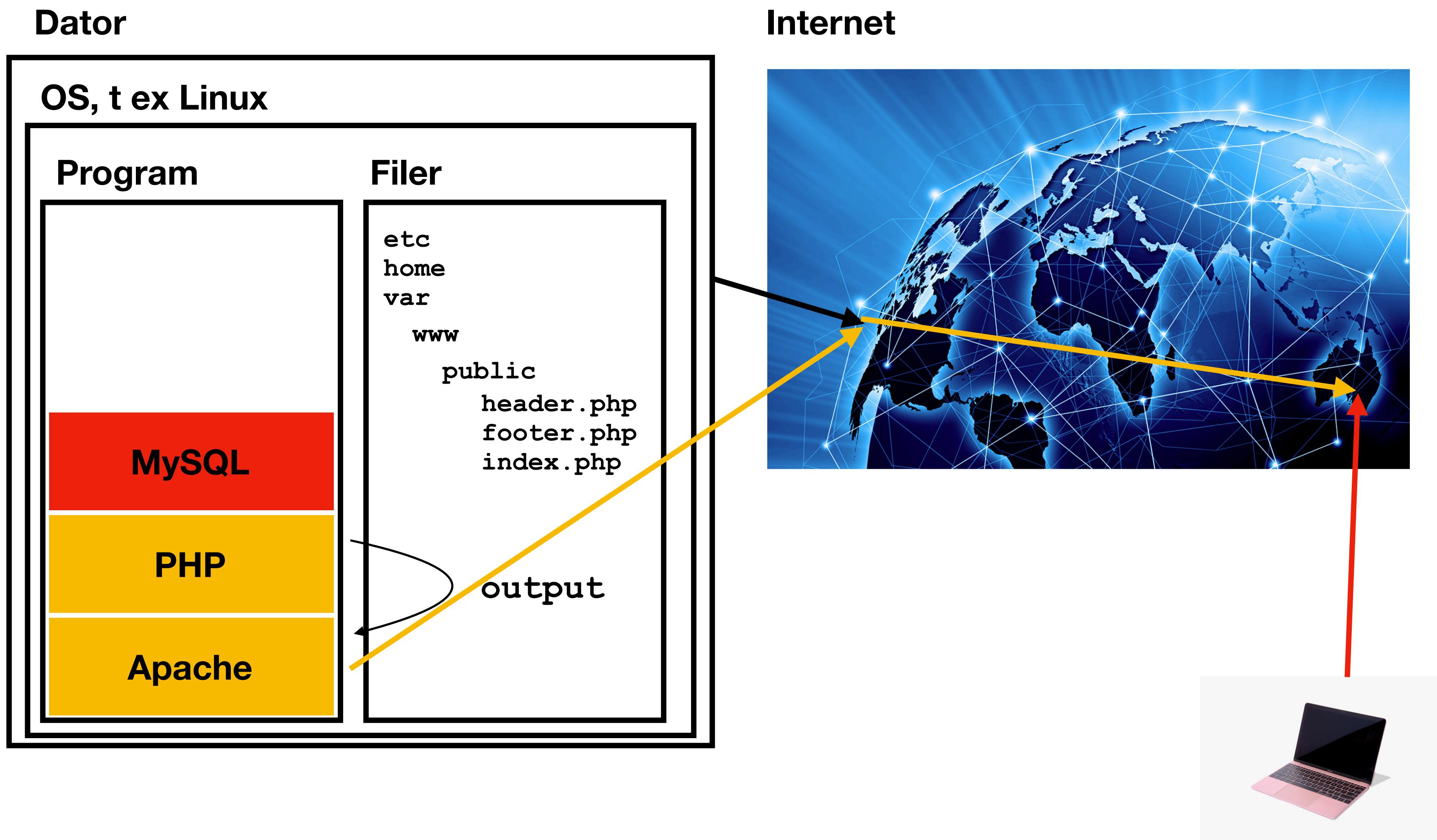
index.php

```
<!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>
```

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



NodeJS

- Open-source, cross-platform JavaScript run-time-miljö som exekverar JavaScript utanför browsern.
- Låter utvecklare använda JS för att skriva Command Line-verktyg och för skript på server-sidan för att producera dynamiska webbsidor. (Alltså precis som PHP.)
- Event-driven arkitektur.
- Hanterar I/O asynkront.

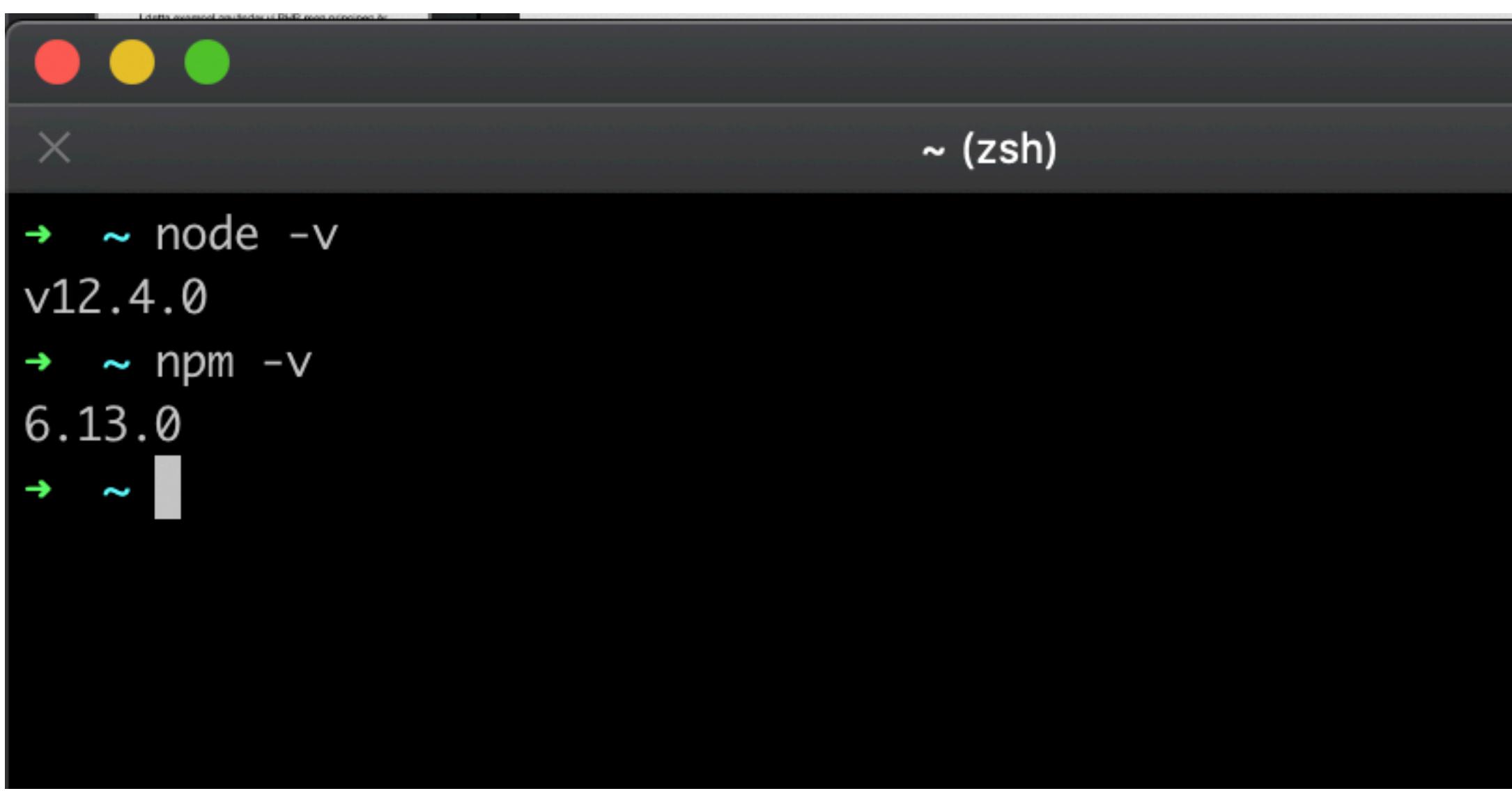
NodeJS

Asynkront

```
const request = require("request");
request(
  "http://swapi.co/api/starships/10/",
  function(err, response, body) {
    console.log("Sen det här");
    console.log(JSON.parse(body));
  }
);
console.log("Det här kommer skrivas ut först");
```

Installera Node

- <https://nodejs.org/en/>
- Testa genom att köra i terminalen.
- Ni borde redan ha node / npm sedan JS-kursen.



A screenshot of a macOS terminal window titled '~ (zsh)'. The window shows the following command-line session:

```
→ ~ node -v
v12.4.0
→ ~ npm -v
6.13.0
→ ~ |
```

NodeJS

npm

- npm är en pakethanterare för NodeJS som gör det lätt att ladda hem funktionalitet som andra redan byggt.
- Det är också enkelt att ladda upp sina egna paket.
- Just nu finns det uppskattningsvis 1,5 miljoner paket som man kan använda sig av.

NodeJS package.json

- Vi kan initiera ett nytt projekt genom att skriva npm init.
- Startar en guide som hjälper oss skapa en package.json.

NodeJS package.json

- Package.json innehåller information om vårt projekt, dependencies osv.
- Kan vara olika för olika miljöer.

```
1  {
2    "name": "realtime-chat",
3    "scripts": {
4      "start": "node index"
5    },
6    "dependencies": {
7      "express": "latest",
8      "socket.io": "^2.1.1"
9    }
10 }
11 
```

Argv

- Argv ger oss en array med sökvägen till node, filen och alla parametrar.
- https://nodejs.org/docs/latest/api/process.html#process_argv
- *Uppgift:* Använd denna kunskap för att göra ett program som tar två tal och ett räknesätt som parametrar och skriver ut resultatet.

```
// print process.argv
process.argv.forEach((val, index) => {
  console.log(`#${index}: ${val}`);
});
```

```
$ node process-args.js one two=three four
0: /usr/local/bin/node
1: /Users/mjr/work/node/process-args.js
2: one
3: two=three
4: four
```

Filsystem

- Vi har inte haft möjlighet att jobba med filer på klientsidan ("vanliga js"), men det har vi i node.

We will need the fs module, so first require it, like this:

```
1 | var fs = require('fs');
```

`__dirname`

`__dirname` is a special variable that always points to the current working directory.

If we `console.log __dirname` it will give us a string showing us our current directory.

Read a file with `fs.readFile()`.

We can read the file using `fs.readFile()`:

```
1 | fs.readFile(path, function (error,data) {  
2 |   // data contains the file contents  
3 |   // error is null if no error occurred  
4 | });
```

Filsystem

- Exercise 1 - get the current directory
 - Create a little program that outputs the current working directory.
- Exercise 2 - Read a file
 - Make a file containing some content. Write a program that can read a file from the current directory and output it to the console.

Göra requests

- Från Node kan man göra requests till andra system (t ex API:er) och hämta/lämna information.
- <https://nodejs.dev/learn/making-http-requests-with-nodejs#perform-a-get-request>
- *Uppgift:* Hämta ett par skämt från <https://icanhazdadjoke.com/api> och skriv ut dem.
 - Klura ut hur ni får ut svaret som JSON, vad ni behöver ändra i exemplet för att få till det och hur man ska göra det. (Allting finns på sidan i olika exempel.)

Skapa en enkel webserver

- Både node och php har inbyggda webbservrar som kan lyssna på requests.
- Innehåller inte alla funktioner som t ex Apache och Nginx innehåller, men gott nog för att testa lite.

Skapa filen app.js

```
const http = require('http');

const hostname = '127.0.0.1';
const port = 3000;

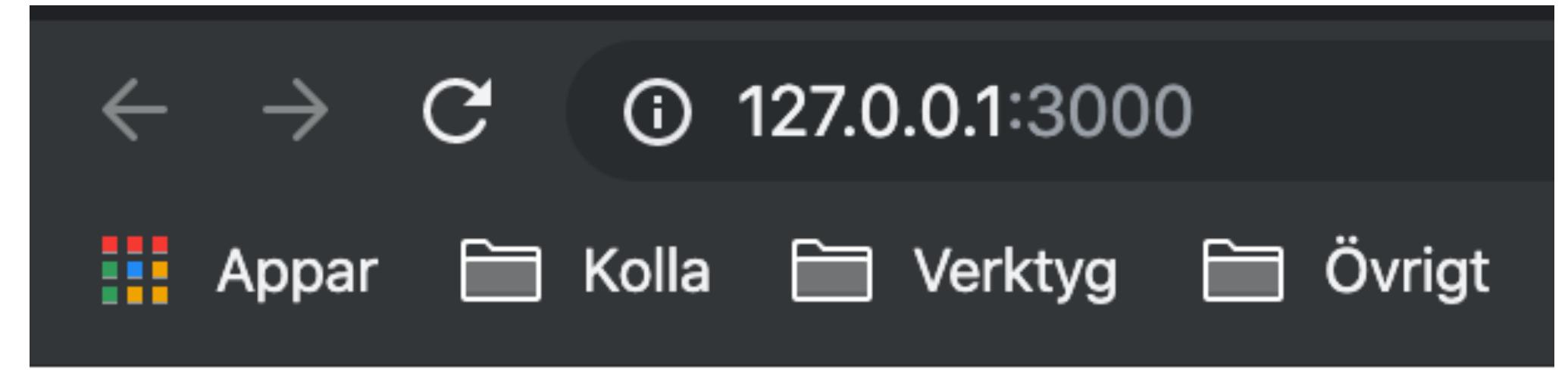
const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World\n');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

Kör filen med
node app.js

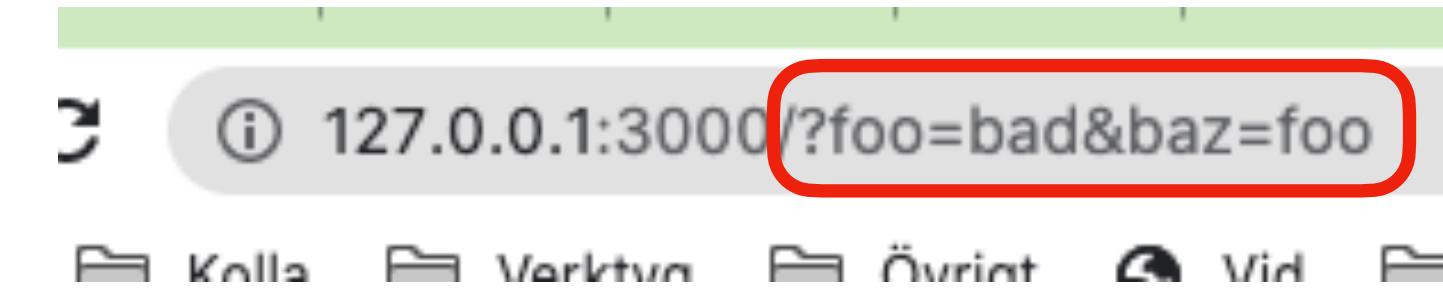
PROBLEMS 14 OUTPUT DEBUG CONSOLE TERMINAL

```
→ hello_world node app.js
Server running at http://127.0.0.1:3000/
```



Hello World

Querystring



- Kommer ni ihåg vad en querystring är? Allt efter frågetecknet i URL:en.
- <https://nodejs.org/en/knowledge/HTTP/clients/how-to-access-query-string-parameters/>
- *Uppgift:* Utgå från er tidigare applikation och skapa en applikation som skapar en webbserver och tar emot en querystring som innehåller två tal och ett räknesätt och både skriver ut talen och summan ($2 + 3 = 4$) i loggen och i browsern.

Övningar

- Fler övningar? Testa nodeschool! Den bygger på ett utdaterat ramverk, men de flesta av övningarna ska fungera!
- <https://nodeschool.io/>
- `npm install -g learnyounode`